

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA
ESCUELA DE INGENIERIA EN COMPUTACIÓN



**DISEÑO Y DESARROLLO DE UNA APLICACIÓN WEB PARA EL MANEJO DE
INFORMACIÓN CORRESPONDIENTE AL PROCESO DE REHABILITACIÓN DE
PACIENTES CON DISCAPACIDADES FÍSICAS DE FUNTER.**

TRABAJO DE GRADUACIÓN
PREPARADO PARA LA FACULTAD DE INGENIERÍA

PARA OPTAR AL GRADO DE
INGENIERO EN CIENCIAS DE LA COMPUTACIÓN

PRESENTADO POR:

JUAN JOSÉ RIVAS ÁNGEL
BORIS ALEXANDER MARTÍNEZ DÍAZ
GIOVANNI ALEXANDER RAMÍREZ MADRID

CIUDADELA DON BOSCO

SEPTIEMBRE, 2002.

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA
ESCUELA DE INGENIERIA EN COMPUTACIÓN



RECTOR
ING. FEDERICO MIGUEL HUGUET RIVERA

SECRETARIO GENERAL
LIC. MARIO OLMOS

DECANO DE LA FACULTAD DE INGENIERIA
ING. CARLOS GUILLERMO BRAN

ASESOR DEL TRABAJO DE GRADUACIÓN
ING. CARLOS ROSALES

JURADO EVALUADOR DEL TRABAJO DE GRADUACIÓN
ING. MELVIN CARIAS
ING. LORNEE QUINTANILLA

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA
ESCUELA DE INGENIERIA EN COMPUTACIÓN



**DISEÑO Y DESARROLLO DE UNA APLICACIÓN WEB PARA EL MANEJO DE
INFORMACIÓN CORRESPONDIENTE AL PROCESO DE REHABILITACIÓN DE
PACIENTES CON DISCAPACIDADES FÍSICAS DE FUNTER.**

JURADO EVALUADOR DEL TRABAJO DE GRADUACIÓN

ING. MELVIN CARIAS
JURADO

ING. LORNEE QUINTANILLA
JURADO

ING. CARLOS ROSALES
ASESOR

AGRADECIMIENTOS

Agradezco a Dios todopoderoso por el don de la vida, por todas sus bondades y bendiciones y por la oportunidad que me ha dado de culminar esta carrera.

Agradezco a mis padres Francisco y Cristina por su amor y por darme tanto en esta vida, por sus palabras de apoyo y comprensión y por sus sabios consejos. A mis hermanos Francisco y Ana Iris por estar siempre pendientes en el desarrollo de mi trabajo de graduación animandome a cada momento.

Agradezco a mi esposa Karla y a mi hijito Enrique que han sabido comprender los momentos que no he podido estar con ellos por el tiempo que le dedique a este trabajo de graduación.

Agradezco a mis suegros Jorge y Guadalupe por apoyarme en mis estudios, además porque me ayudaron en los cuidados de mi hijito cuando yo no podía atenderlo.

Agradezco a mis compañeros y amigos Giovanni Ramírez y Juan José Rivas, por compartir conmigo todo este tiempo de trabajo, en el cual juntos crecimos no solo en lo profesional sino que también en lo humano; también agradezco a mis amigos: Meylin, Paty, Xenia y Denver, con quienes compartimos muchos momentos agradables como estudiantes de la universidad.

Agradezco a mis compañeros de trabajo que siempre estuvieron pendientes del desarrollo de mi trabajo de graduación y me ayudaron con algún consejo cuando se los solicitaba; además agradezco a mi jefe ingeniero Julio Fong, por otorgarme todos los permisos que necesité, cuando por motivos de mi trabajo de graduación necesitaba ausentarme de mi trabajo.

Boris Alexander Martínez Díaz.

AGRADECIMIENTOS

Agradezco a Dios todopoderoso por regalarme la vida, por darme la oportunidad de terminar mi carrera y por derramar sobre mí sus bendiciones y la sabiduría necesaria para superarme cada día.

Agradezco a mis padres Juan José y Judith por su amor, su cariño y comprensión porque me han ayudado y brindado su apoyo durante toda mi vida, gracias a sus enseñanzas y consejos que ahora son la luz en mi camino, porque cuando necesite una palabra de aliento siempre la encontré en ellos.

Agradezco a mis hermanitas Mirta y Judith María por su amor y comprensión, por apoyarme en cada momento, por haber estado siempre ahí cuando yo las necesité; también agradezco a Karla por haberme apoyado tanto, por su amor incondicional y por su comprensión en todo momento.

Agradezco a mis amigos Alex y Giovanni por tantas noches de desvelo y trabajo juntos, porque nuestra amistad se ha fortalecido y porque juntos hemos crecido en lo humano y en lo profesional, gracias por todo su apoyo y trabajo.

Agradezco a mis amigos Meylin, Xenia, Paty y Denver por su apoyo incondicional, por haber sido una pieza fundamental en mi desarrollo personal y profesional.

Agradezco a los Lobos Laicos por compartir tantas experiencias de vida juntos y por el apoyo incondicional que me brindaron; también agradezco a los Hermanos Maristas y a los profesores del Liceo Salvadoreño que me ayudaron tanto en mi formación cristiana y en mi formación académica, en especial a los profesores Priscila, Hidanne de Melado y al profesor Rivas, gracias por todo su apoyo.

Juan José Rivas Ángel.

AGRADECIMIENTOS

Agradezco en primer lugar a Dios todopoderoso por el don de la vida, por todas las bendiciones que día a día derrama sobre mí y por la oportunidad que me ha dado de culminar esta carrera universitaria.

Agradezco a mis padres René y Aida por su amor, por su comprensión y por darme tanto en esta vida, no solo en lo material sino también en lo espiritual y en lo humano, con lo cual he caminado durante todo este tiempo. A mis hermanos René y Walter por estar siempre pendientes de mí, por su cariño y sus palabras de aliento.

Agradezco a Roxana y Carmen, mis mejores amigas, por su amor y apoyo incondicional brindado en todo momento.

Agradezco a mis compañeros y amigos Juan José y Boris Alexánder, por compartir conmigo todo este tiempo de trabajo y desvelo, en el cual juntos crecimos no solo en lo académico sino que también en lo humano; también agradezco a mis amigas y amigos: Meylin, Paty, Xenia y Denver, con quienes compartimos muchos momentos inolvidables como estudiantes de la universidad.

Agradezco a mis hermanos cooperadores salesianos y de Iglesia Joven, por sus palabras de ánimo y fortaleza brindadas en los momentos difíciles; así como también a mis compañeros del FISDL por todos los aportes y la experiencia que me han transmitido en todo el tiempo que llevo compartiendo con ellos el trabajo cotidiano.

Finalmente a los ingenieros Carlos Rosales, Melvin Carías y Lornee Quintanilla, por todo el conocimiento, tiempo y esfuerzo compartido.

Giovanni Alexnder Ramrez Madrid.

AGRADECIMIENTOS

Agradecemos a la familia Martínez Díaz, por acogernos en su hogar durante todo el tiempo en el que desarrollamos nuestro trabajo de graduación, además por las noches de desvelo que compartieron con nosotros, así como también por todas las palabras de aliento que nos brindaron en los momentos más difíciles.

Agradecemos al ingeniero Carlos Rosales, nuestro asesor, por todos los consejos que nos brindó durante el desarrollo de nuestro trabajo de graduación, además le agradecemos por toda su experiencia que compartió con nosotros, por su paciencia, esfuerzo y por el buen sentido del humor con el cual siempre nos animó a seguir adelante a pesar de las adversidades del camino.

Agradecemos al personal de la Fundación Teletón Pro-Rehabilitación (FUNTER) que nos atendió y dedicó su tiempo en el proceso de investigación de nuestro proyecto.

Agradecemos al personal de la Universidad Don Bosco que nos apoyó siempre profesional y cordialmente en todo el proceso de nuestra carrera universitaria.

Finalmente, agradecemos a los ingenieros Melvin Carías y Lornee Quintanilla, por todo el conocimiento, tiempo y esfuerzo compartido con nosotros durante el proceso de nuestro trabajo de graduación.

Boris Alexander Martínez Díaz

Juan José Rivas Ángel

Giovanni Alexánder Ramírez Madrid

INDICE

INTRODUCCIÓN	Página I
--------------	-------------

CAPITULO I

PLANTEAMIENTO GENERAL PROYECTO

1. Antecedentes.	1
1.1. Identificación y Naturaleza de FUNTER.	2
1.2. Misión de FUNTER.	2
1.3. Visión de FUNTER.	3
1.4. Objetivos de FUNTER.	3
1.5. Caracterización de la Población Objetivo de FUNTER.	4
1.6. Organización Institucional.	5
1.7. Experiencia Institucional de FUNTER.	6
1.8. Metodología de FUNTER.	7
2. Justificación.	9
3. Proyección Social.	10
4. Definición del Tema.	11
5. Objetivos.	12
5.1. Objetivo General.	12
5.2. Objetivos Específicos.	12
6. Alcances.	13
7. Limitaciones.	14

CAPITULO II

INVESTIGACIÓN Y ANÁLISIS DE LA INFORMACIÓN

1. Metodología y Técnicas Utilizadas	15
1.1. Metodología RUP.	15
1.2. Casos de Uso.	18
1.3. Diagramas de Flujo de Datos.	23
2. Marco Teórico	25

2.1. Internet.	25
2.2. Intranet.	27
2.3. Tecnologías	28
2.3.1. Arquitecturas de Aplicaciones.	28
2.3.2. Windows 2000 Server.	33
2.3.3. Internet Information Server 5.0.	41
2.3.4. SQL Server 2000.	43
2.3.5. Active Server Pages.	46
3. Situación Actual.	47

CAPITULO III

DESCRIPCIÓN DE LA APLICACIÓN

1. Descripción general de la Aplicación.	49
2. Arquitectura de la Aplicación.	51
2.1. Tecnología utilizada en la Aplicación.	52
3. Estándares de la Aplicación y de la Base de Datos.	53
4. Diseño de la Base de Datos.	57
4.1. Modelo Conceptual de Datos.	57
4.1.1. Modelo Conceptual: General del SIAPD.	58
4.1.2. Sub-Modelo Concep.: Pacientes del SIAPD.	59
4.2. Modelo Físico de Datos.	60
4.2.1. Modelo Físico: General del SIAPD.	61
4.2.2. Sub-Modelo Físico: Pacientes del SIAPD.	62
4.3. Diccionario de Datos	63
4.3.1. Listado de Objetos de la Base de Datos.	63
4.3.1.1. Listado de Tablas	63
4.3.1.1. Listado de Referencias	65
4.3.1.1. Listado de Triggers	70
4.3.1.1. Listado de Procedimientos Almac.	71
4.3.2. Detalle de las Tablas de la Base de Datos.	72
5. Descripción de los Módulos de la Aplicación	73
5.1. Módulos Públicos de la Aplicación.	73
5.1.1. Módulo de Información.	73
5.1.2. Módulo de Solicitud de Servicios.	74
5.2. Módulos Privados de la Aplicación	75
5.2.1. Módulo SIAPD.	75
5.2.1. Módulo Pacientes.	78

CAPITULO IV

SEGURIDAD DE LA APLICACIÓN

1. Conceptos Generales	79
2. Esquema de Control de Accesos.	80
3. Seguridad a nivel de la Aplicación.	81
7. Seguridad a nivel de la Base de Datos.	82

CONCLUSIONES	83
FUENTES DE INFORMACIÓN	85
GLOSARIO	87
ANEXOS	
I. Plantilla para describir los Casos de Uso.	90
II. Diagramas de Flujos de Datos de la Aplicación.	91
III. Casos de Uso de la Aplicación.	101
IV. Detalle de las Tablas de la Base de Datos.	151
V. Script Almacenados de la Base de Datos.	226

INTRODUCCIÓN

Para que una sociedad pueda desarrollarse apropiadamente se necesita, entre otros elementos, que su población tenga las cualidades físicas e intelectuales adecuadas para impulsar dicho desarrollo; actualmente en El Salvador aproximadamente el 13% de la población no posee las cualidades físicas apropiadas para dichos fines, debido a discapacidades que son consecuencia de diferentes fenómenos, esto resta al país de un gran potencial humano que pueda contribuir con su desarrollo.

Por tal motivo, es necesario impulsar en el país acciones integrales en favor de esta población con discapacidades físicas, para que así estas personas puedan reinsertarse en la sociedad, pudiendo de esta manera contribuir con su esfuerzo en el desarrollo de la misma.

Actualmente, la Fundación Teletón Pro-Rehabilitación (FUNTER) es una institución sin fines de lucro que se dedica a la rehabilitación integral de los salvadoreños que padecen de alguna discapacidad física y/o sensorial, para que luego estos se puedan incorporar plenamente en la sociedad. Es así como hasta la fecha la Institución ha atendido a más de 591,250 personas¹ con diversas discapacidades físicas, mediante servicios médicos, psicológicos, fisioterapéuticos y sociales apropiados para una efectiva rehabilitación integral de dichas personas.

La creciente demanda de los servicios que presta FUNTER dificulta el control de información correspondiente a dichos servicios, es por este motivo que se concibe la idea de desarrollar una aplicación que le permita a la Institución agilizar la atención a sus pacientes, mediante una herramienta versátil que facilite el manejo de información correspondiente a los procesos de rehabilitación de dichos pacientes.

¹ Según datos de FUNTER, registrados hasta el año 2000.

CAPITULO I

PLANTEAMIENTO GENERAL DEL PROYECTO

1. ANTECEDENTES.

En El Salvador cerca del 13% de la población sufre de algún tipo de discapacidad física, sensorial o mental, lo cual significa que en el país existen aproximadamente 832 mil personas² con discapacidades físicas a causa diferentes factores, como por ejemplo enfermedades, accidentes, el pasado conflicto armado, entre otros; estas personas se ven imposibilitadas para realizar óptimamente sus actividades personales y laborales, debido a estas discapacidades y a la carencia en la atención especializada que deben recibir para poder así reinsertarse y desarrollarse adecuadamente en la sociedad.

Actualmente, FUNTER es una Institución que se dedica a la rehabilitación integral de las personas con discapacidades físicas, a través de la atención a dichas personas mediante servicios clínicos especializados, así como también mediante programas orientados a la capacitación laboral e inserción a puestos de trabajo remunerados para dichas personas.

Hay que recalcar que los servicios que presta esta institución en ocasiones se ven estancados debido a la gran demanda de los mismos y también a la falta de recursos especializados para poder brindar la atención debida a los pacientes en cuanto al manejo de la información correspondiente a sus procesos de rehabilitación.

² Según datos de FUNTER, registrados hasta el año 2000.

1.1. Identificación y Naturaleza de FUNTER.

La Fundación Teletón Pro-Rehabilitación FUNTER, obtiene su personería jurídica el 15 de enero de 1987, por Decreto Ejecutivo número 294 y a través de estos 15 años de trabajo, ha logrado posicionarse como una institución ejecutora experimentada en el desarrollo de Programas y Proyectos orientados a dar cumplimiento a su deber ser, es decir a brindar servicios de Rehabilitación Integral de las personas con discapacidades, a través de la atención de servicios clínicos especializados, así como atender a esta misma población, con programas dedicados a la capacitación laboral e inserción a puestos de trabajo remunerados.

Por otra parte, la Institución atiende aspectos de política, en la elaboración de la Ley de Equiparación de oportunidades y sobre la ejecución de un proyecto a nivel Regional sobre la Accesibilidad: Eliminación de Barreras Arquitectónicas, para la población con discapacidades.

1.2. Misión de FUNTER.

La Fundación Teletón pro Rehabilitación FUNTER, marca su misión de la siguiente manera: “Procurar la Rehabilitación Integral de las personas con discapacidades para su incorporación plena a la sociedad”.

Su Misión, ha sido el baluarte en el desarrollo institucional de FUNTER y durante 15 años, sus servicios han estado dirigidos a beneficiar a la población más vulnerable de nuestro país, a través de la atención a las personas con algún tipo de discapacidad, perteneciente a los estratos más pobres

1.3. Visión de FUNTER.

La visión de FUNTER se enfoca en las siguientes palabras: “Ser la institución líder de la rehabilitación integral en El Salvador, por medio de la ejecución de programas, coordinación, integración y apoyo a las entidades que prestan servicios de rehabilitación”.

1.4. Objetivos de FUNTER.

Los Objetivos principales de la Fundación Teletón Pro-Rehabilitación son los siguientes:

- ◆ Fortalecer la red de atención y brindar servicios profesionales de rehabilitación funcional y vocacional con eficiencia, calidad y en forma oportuna a las personas con discapacidades neuro-músculo-esqueléticas, así como afecciones del oído y/o problemas del habla.
- ◆ Contribuir a procurar una mejor calidad de vida a las personas con discapacidad y de su grupo familiar a través de su educación formal o informal para su incorporación en el campo laboral y/o productivo.
- ◆ Consolidar el Fondo Patrimonial y ampliar la red de apoyo técnico-financiero para asegurar la permanencia y cobertura nacional de los servicios que brinda FUNTER a los diferentes niveles de atención.
- ◆ Facilitar al acceso a tecnologías de punta que mejoran la calidad de los servicios de la Institución.
- ◆ Promover medidas, estrategias, mecanismos y la educación constante encaminada a la prevención, detección y tratamiento de factores discapacitantes, la participación plena y la igualdad de oportunidades en la población discapacitada.
- ◆ Promover programas de sensibilización a la comunidad sobre necesidades y derechos de las personas con discapacidad.

- ◆ Contribuir a garantizar la dotación de servicios de rehabilitación integral para todos los salvadoreños estableciendo diferentes planes de acceso a los mismos, asegurando su oportunidad y eficiencia.

1.5. Caracterización de la Población Objetivo de FUNTER.

u

La Organización Mundial de la Salud, OMS/OPS, señala que todos los países del mundo poseen un 10% de su población en general con discapacidades. De acuerdo a datos proporcionados por las Naciones Unidas, más de 500 millones de personas sufren a nivel mundial algún tipo de discapacidad. Este dato ha sido tomado dentro de la norma de un comportamiento típico.

En el caso de El Salvador, tendríamos que considerar la incidencia de carácter excepcional de la situación del enfrentamiento bélico, que se sufrió por más de una década y los desastres naturales que han afectado a nuestro país en los últimos años. En el país, cerca del 13% de la población, sufre de algún tipo de discapacidad física, sensorial o mental, lo cual significa que existe aproximadamente 832 mil personas con discapacidades, quienes se ven imposibilitadas para realizar a plenitud sus actividades personales y laborales, por la carencia en las atenciones especializadas que deben recibir.

Se tiene conciencia sobre este problema en suma delicado, el cual constituye un reto importantísimo, ya que se deben crear las condiciones y el ambiente propicio, para que estas personas puedan integrarse a la vida familiar comunitaria y laboral, por medio del acceso directo a servicios eficientes de rehabilitación, donde se les brinde atención de alta calidad, que les posibilite un mejor desarrollo de sus potencialidades residuales, vida.

Es por eso que FUNTER, está realizando esfuerzos por potenciar la atención a personas con discapacidad física (neurológica, muscular y/o esquelética además de sensoriales: audición y lenguaje) dentro del Área de Rehabilitación Funcional; además, ha atendido con mucho éxito, aspectos de capacitación vocacional e inserción laboral, en el Área de Rehabilitación Profesional y en menor grado, aunque

no menos importante, educación, capacitación y sensibilización de la sociedad hacia las necesidades y derechos de las personas con discapacidad, que corresponde al Área de Rehabilitación Social.

La institución ofrece básicamente, Rehabilitación Funcional, con atención médica fisiatra, Atención Psicológica, Terapia Física, Terapia de Lenguaje y dotación de prótesis y órtesis. No se debe olvidar que la atención en rehabilitación es un proceso, por lo tanto al paciente no se le recupera y se le olvida, sino que debe llevarse una labor de seguimiento, de tal forma, que la persona rehabilitada en el área funcional siga después accedendo al área subsiguiente: La Rehabilitación Vocacional y de esta manera, pueda superar cada etapa de su proceso; esto no implica un paternalismo de la institución, sino un derecho del ser humano en la búsqueda de la equiparación de las oportunidades, derecho ineludible dentro del Desarrollo Humano Sostenible.

1.6. Organización Institucional.

Estructuralmente FUNTER cuenta con tres niveles organizacionales para su accionar, los cuales son: Nivel Decisorio, de Apoyo y Ejecutor.

El nivel Decisorio está formado por la Asamblea General, la Junta Directiva y la Dirección Ejecutiva.

El nivel de Apoyo lo integran la unidad de Planificación, Relaciones Públicas, el Comité Técnico y la unidad de Informática.

EL nivel ejecutor está conformado por tres gerencias: Asistencia Integral a Pacientes, Recaudación de Fondos y Administrativa Financiera.

Esta estructura esta compuesta por aproximadamente 50 personas, entre las cuales se cuenta con profesionales especializados en el área de rehabilitación: médicos, fisiatras, terapistas, técnicos protesistas y sicólogos.

1.7. Experiencia Institucional de FUNTER.

El trabajo que la Fundación realiza en pro de las personas con discapacidad, es presentado brevemente a continuación:

- ◆ ***Centro de Rehabilitación Integral (CRI):***

Fue creado en abril de 1989 y es el responsable de proporcionar servicios médicos, psicológicos, sociales, fisioterapéuticos, de terapia ocupacional, terapia de lenguaje y estimulación temprana, a pacientes con discapacidades físicas, facilitando su integración social y productiva. A la fecha, se han brindado 591,254 atenciones diversas a personas con alguna discapacidad.

- ◆ ***Fábrica de Prótesis y Ortesis (FA PRO):***

Inaugurada en mayo de 1988, esta área es la encargada de la fabricación de prótesis de miembros inferiores y superiores, así como aparatos ortopédicos y protésicos. La FAPRO se encuentra equipada con maquinaria idónea y cuenta, además, con personal técnico debidamente capacitado. A la fecha fabricado más de 6,164 prótesis y 6,041 aparatos ortopédicos.

- ◆ ***Programa Protejamos el oído de nuestros niños (PONI):***

Gracias a la colaboración de la Public Welfare Foundation de Washington y del Sr. Chun Boum Park, ex embajador de la República de Corea, en 1989, dio inicio el Programa PONI, el cual desarrolla campañas de prevención y detección de sordera en niños en edad escolar. Hasta la fecha, se han realizado exámenes auditivos a 330,754 niños y niñas del área urbana y rural.

- ◆ ***Rehabilitación Vocacional:***

En enero de 1991, da inicio el Programa de Rehabilitación Vocacional, cuyo objetivo es brindar servicios de orientación vocacional y lograr la inserción laboral de personas con discapacidad en edad productiva. Este esfuerzo obedece al apoyo de gobiernos amigos e instancias que apoyan la

rehabilitación laboral de personas con discapacidad, habiéndose logrado capacitar a 273 personas con alguna discapacidad en diversas actividades productivas y se han colocado en puestos de trabajo remunerados a un total de 711 personas, a lo largo del territorio nacional.

◆ **Otros Servicios:**

Además FUNTER cuenta con servicios como lo es la dotación de aditamentos para la marcha (sillas de ruedas, bastones, muletas, etc.) y, una completa información Bibliográfica con temas relacionados a la rehabilitación y Discapacidad en general en la cual se incluyen Manuales de Accesibilidad: Eliminación de Barreras Arquitectónicas, con los cuales se garantizan un adecuado soporte técnico para cualquier iniciativa en estas áreas. Como un aporte adicional a la población con discapacidades, la Fundación cuenta actualmente con un Centro de Capacitación en Informática “Sin Barreras”, en el cual se capacitan a personas con discapacidad en el uso de paquetes informáticos, de manera que se mejoren sus hojas de vida y se les amplíen las oportunidades en otras áreas laborales o complementen las existentes.

1.8. Metodología de FUNTER.

La Fundación provee de servicios de rehabilitación a la población salvadoreña que asiste al Instituto FUNTER, aunque conocedores de la falta de estos servicios en el interior del país, se apoya a 17 Centros de atención primaria, distribuidos en los catorce departamentos, de manera que a partir de un diagnóstico especializado el personal de estos Centros puede dar seguimiento al cumplimiento del tratamiento.

Por otra parte el Programa Protejamos el Oído de Nuestros Niños (PONI), se traslada por todo el territorio nacional, brindando atenciones enfocadas a la prevención y detección temprana de problemas auditivos, principalmente en la población infantil, contando para ello con personal y equipo especializado en el área.

Además el Programa de Rehabilitación Vocacional, recorre el territorio nacional en la búsqueda de personas con discapacidad en edad productiva, para apoyar su inserción laboral, realizando previamente una evaluación física, para determinar las habilidades residuales de estas personas y una evaluación psicológica, de manera que se pueda garantizar que están aptos para desarrollar una actividad productiva y en algunos casos, canalizar la capacitación que estos necesiten antes de ser incorporados a su vida laboral.

2. JUSTIFICACIÓN.

Una aplicación web confiable que permita al personal de FUNTER manipular adecuadamente la información correspondiente a los procesos de rehabilitación de personas con discapacidades físicas que se realizan en la Institución permitiría hacer más eficientes y eficaces dichos procesos, pudiendo así lograr que más personas tengan acceso a ellos, y por lo tanto fomentar así la capacidad de reinserción en la sociedad de dichas personas y su buen desenvolvimiento en ella.

Debido a que esta aplicación contiene elementos que pueden ser accedidos por la población discapacitada del país que no son pacientes de FUNTER, a través del Internet, se facilitará a estas personas el acceso a información de su interés así como la solicitud de diversos servicios a la Institución, por lo tanto esto permite agilizar el proceso de selección de nuevos pacientes para que puedan acceder a dichos servicios, así como también se mejora el tiempo de respuesta para algunos de los servicios que se presta en FUNTER.

Además una aplicación de este tipo colaboraría en gran medida con la interacción y comunicación entre FUNTER y sus diferentes pacientes, permitiendo así acelerar algunos de los procesos de la Institución que actualmente toman mucho tiempo.

3. PROYECCIÓN SOCIAL.

Por realizarse el proyecto de tesis para FUNTER, la cual es una entidad sin fines de lucro y que presta diversos servicios a la comunidad salvadoreña con problemas de discapacidades físicas, sobre todo a aquellos con escasos recursos económicos; se considera que dicho proyecto tiene una amplia proyección social, ya que con este se estará colaborando a que la Institución tenga un medio que le permita hacer más eficientes sus procesos, para así poder ampliar y mejorar sus servicios a la población con discapacidades físicas de El Salvador, permitiendo de esta forma que más personas puedan posteriormente reinsertarse en la sociedad pudiendo de esta manera beneficiarse a nivel personal y beneficiar también a la sociedad salvadoreña en general.

4. DEFINICIÓN DEL TEMA.

Diseño y desarrollo de una aplicación web para el manejo de información correspondiente a los procesos de rehabilitación de pacientes con discapacidades físicas, que son atendidos por la Fundación Teletón Pro-Rehabilitación (FUNTER); que además permita a los usuarios que no son miembros de la Institución acceder a la información concerniente a los servicios que se desarrollan en la misma, así como la solicitud en línea de dichos servicios.

5. OBJETIVOS.

5.1. OBJETIVO GENERAL.

Diseñar y desarrollar una aplicación web para el manejo de información correspondiente a los procesos de rehabilitación de pacientes con discapacidades físicas de FUNTER, la cual ofrezca una gama de recursos y servicios al personal de la Institución, sus pacientes y a la población en general.

5.2. OBJETIVOS ESPECÍFICOS.

1. Proveer a FUNTER de un mecanismo eficaz, eficiente y confiable para el almacenamiento y manejo de la información correspondiente a sus pacientes y sus respectivos procesos de rehabilitación.
2. Permitir a cualquier interesado solicitar los servicios de rehabilitación que presta la Institución a través de la aplicación web.
3. Facilitar, por medio de la aplicación web, la interacción entre los pacientes con discapacidades físicas y el personal de la Institución.
4. Brindar a través de la aplicación web, los enlaces (links) a información general concerniente a los servicios que presta FUNTER así como también a información relacionada con el acceso a dichos servicios.

6. ALCANCES.

1. Se diseñará y desarrollará una base de datos que permita almacenar y manipular eficientemente toda la información acerca de los procesos de rehabilitación de los pacientes de la Institución.
2. Se desarrollará una aplicación web que permitirá a sus usuarios lo siguiente:
 - 2.1. Consultar información general acerca de los servicios que FUNTER presta a las personas con discapacidades físicas, así como también información acerca de cómo acceder a dichos servicios.
 - 2.2. Llenar formularios en línea, como solicitud de consultas médicas y terapias, para poder así aplicar a los servicios que presta la Institución.
 - 2.3. Acceso, solo para usuarios específicos, a un módulo administrativo que permitirá el mantenimiento de la Información relacionada con los procesos de rehabilitación de los pacientes con discapacidades físicas que son atendidos en la Institución.
 - 2.4. Acceso a la información relacionada con el historial de rehabilitación de los pacientes de la Institución.
3. La aplicación contará con dos tipos de acceso: público y privado. El acceso público estará dirigido a cualquier persona que navegue en Internet, mientras que el acceso privado estará destinado a los pacientes de FUNTER así como al personal de la Institución.
4. Elaboración de manual técnico y manual de usuario, para facilitar la comprensión de la estructura, funcionalidad y forma de uso de la aplicación web.

7. LIMITACIONES.

1. El proyecto abarca la parte de diseño y desarrollo de la aplicación web, mientras que la implementación de la aplicación será realizada por la Institución, si así lo decide.
2. El contenido de la aplicación será definido en cierta medida por los requerimientos de la Institución, a través de los representantes de las diferentes unidades involucradas en el proyecto, las recomendaciones y observaciones del equipo que desarrollará el proyecto.
3. En cuanto a la seguridad de la aplicación, el proyecto a realizar considerará la consistencia de los datos y la seguridad en cuanto al acceso a la Base de Datos, más sin embargo la seguridad en cuanto al ataque a la aplicación por parte de “hackers” será responsabilidad directa de la Institución.
4. La aplicación web será diseñada y desarrollada considerando los recursos mínimos requeridos de hardware y software que el usuario final puede tener, ya que este no debe adquirir recursos particulares para poder utilizar la aplicación.
5. Las páginas web informativas de la aplicación serán diseñadas y elaboradas por la Institución, sin embargo la aplicación contendrá los enlaces (links) necesarios para poder acceder a estas páginas.
6. La aplicación permitirá a usuarios externos de la Institución llenar formularios en línea, como solicitud de consultas médicas y terapias, sin embargo la verificación de los datos ingresados por dichos usuarios será responsabilidad de FUNTER.

CAPITULO II

INVESTIGACIÓN Y ANÁLISIS DE LA INFORMACIÓN

1. METODOLOGÍAS Y TÉCNICAS UTILIZADAS.

1.1. Metodología RUP.

Para el desarrollo de este proyecto, se utilizó la metodología denominada Proceso Unificador de Rational (RUP, Rational Unified Proces), la cual es una metodología que sirve para levantar información, administrar requerimientos, controlar los cambios, reutilizar componentes, hacer pruebas sistemáticas, administrar los recursos, entre otros; todo esto con el fin de gestionar de una forma eficiente los proyectos informáticos en el menor tiempo posible y de la manera mas efectiva.

A continuación se describe esta metodología, se utilizarán algunos términos propios de la misma, los cuales serán explicados, además se dará un vistazo a las mejores practicas de software que recomienda la metodología para tener proyectos exitosos.

El proceso unificador de Rational (RUP).

El Proceso Unificador de Rational es un *proceso de ingeniería de software*, que provee una metodología de cómo asignar tareas y responsabilidades en un desarrollo organizacional. La meta es asegurar un producto de software de alta calidad con un programa y presupuesto predecible.

El RUP es un *producto de procesos*, que es desarrollado y mantenido por Rational Software e integrado al conjunto de herramientas de desarrollo de Software. Tiene un *proceso de infraestructuras*, que pueden ser adaptadas y extendidas al conjunto de

necesidades que adopta una organización. Además captura muchas de las mejores practicas en el desarrollo de software modernos en una forma que es adecuada para un gran número de proyectos y organizaciones. Las seis mejores prácticas que se proponen son:

- ◆ Desarrollo de software interactivamente.
- ◆ Manejo de requerimientos.
- ◆ Uso de arquitectura basada en componentes.
- ◆ Modelación visual del software.
- ◆ Verificación de la calidad del software
- ◆ Control de cambio del software.

Fases y traslapes del enfoque Rational.

Antes que nada diremos que en un proyecto una *Interacción* es una secuencia de actividades distintas con un plan definido y resultados evaluados con un criterio específico.

Desde la perspectiva de un administrador de proyectos, se necesita un método para medir el progreso, con el que se pueda asegurar que no se está errado y que no se va a la deriva de interacción con interacción y que todas ellas convergen hacia un producto determinado.

Desde la perspectiva del administrador, se debe también definir puntos en el tiempo para operativamente garantizar las funciones básicas a realizar en el desarrollo del proyecto. Los *traslapes* proveen estos puntos, con los cuales se puede decidir proceder, abortar o cambiar el curso de las interacciones.

Finalmente, se debe particionar y organizar la secuencia de interacciones para acordar objetivos en términos cortos.

El progreso de un proyecto debe ser medido en términos de número de casos de uso³ completos, funcionalidades completadas, casos de prueba pasados, cumplimiento de requerimientos satisfechos y riesgos eliminados.

El proceso interactivo es organizado en *fases*, como se muestra en la figura 1. Pero a diferencia de los pasos en el tradicional *enfoque en cascada*, las fases aquí no están en la secuencia tradicional de requerimientos analizados, diseño, codificación, integración y prueba, y además cada fase es completada por un mayor traslape (milestone).

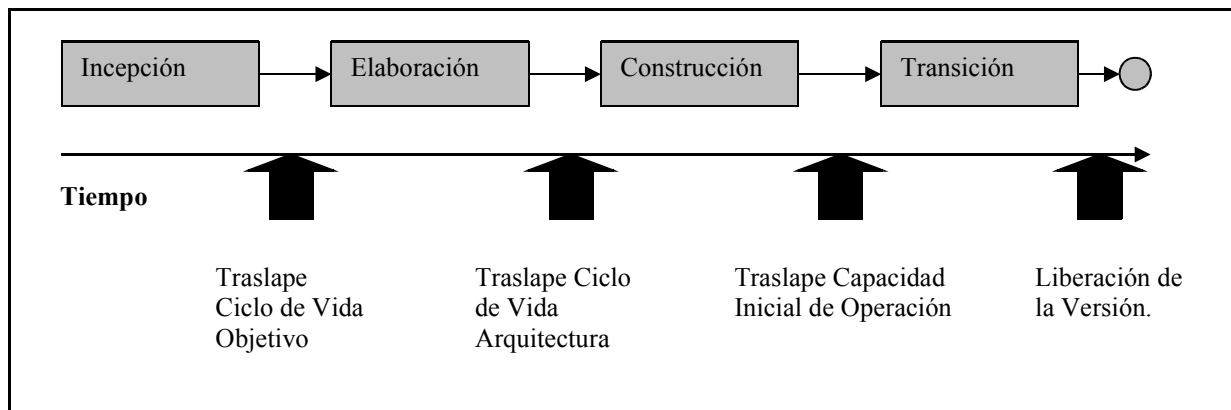


Figura 1. Las cuatro fases y traslapes del proceso interactivo

Las cuatro fases del enfoque Rational son:

◆ *Incepción.*

En esta fase se especifica la visión del producto final, las reglas del negocio y definición del alcance del proyecto. La fase de incepción es concluida por el traslape ciclo de vida objetivo (LCO).

³ Los Casos de Uso de describirán en la sección de Técnicas a Utilizar para la recolección de información y manejo de requerimientos.

◆ *Elaboración.*

Fase en la que se planean las actividades necesarias y recursos requeridos; especificando las características y desarrollo de la arquitectura. La fase de elaboración es concluida por el traslape ciclo de vida de arquitectura (LCA).

◆ *Construcción.*

Consiste en construir el producto y envolver la visión, la arquitectura y el plan hasta el producto, – la completa visión – esta lista para distribuir a los usuarios comunes. Las fase de construcción es concluida por el traslape Operación Inicial de Capacitación (IOC).

◆ *Transición.*

Transferir los productos a los usuarios, con lo cual se incluye, construcción, distribución, entrenamiento, soporte y mantenimiento de los productos hasta que el usuario este satisfecho. Esta fase es concluida por la realización del producto traslape, con el cual concluye el ciclo.

1.2. Casos de Uso.

Para la recolección de información y el manejo de requerimientos en este proyecto se utilizó la técnica de *Casos de Uso*, la cual se describe a continuación:

Una buena parte del Proceso Unificador de Rational - RUP se enfoca en modelar. Los modelos ayudan a entender y encontrar el problema y la solución. Elegir el modelo y elegir la técnica usada para expresarlos tiene un impacto significativo en el método pensado acerca del problema y probar para encontrar la solución.

Hay muchos métodos para modelar problemas y expresar los requerimientos y restricciones en el sistema a ser desarrollado. Pero se debe tener presente que después de esta etapa se puede tener formulado un modelo de la solución; si el

modelo del problema esta alejado del modelo de la solución, un magnifico esfuerzo podría ser costoso para trasladar el problema de una forma entendible por el usuario a una forma entendible por el diseñador y el constructor. Esto significa que hay muchas mas oportunidades para confusiones, y esto hace que sea más difícil validar la solución con respecto al estado del problema. Por otra parte, el proceso envuelve muchos modelos que deben mantenerse consistentes.

Para modelar el problema, el RUP recomienda la técnica de modelación *Casos de Uso*. Los casos de uso proveen un significado para expresar un problema en un método que es entendible por un amplio rango de stakeholders: usuarios, desarrolladores y arquitectos.

Definiciones

El RUP define dos conceptos claves: casos de uso y actores:

- ◆ *Casos de Uso*: es una secuencia de acciones en un sistema funcional, que produce resultados observables de valor a un actor en particular.

- ◆ *Actor*: es alguien o algo fuera del sistema que interactúa con el sistema.

Por lo tanto, tenemos el sistema bajo consideración, el cual gira alrededor de los actores (personas u otros sistemas) que interactúan con este, y tenemos casos de uso que definen esas interacciones.

De las definiciones anteriores hay que considerar muchos puntos clave:

- ◆ *Acciones*: es un procedimiento o algoritmo computacional que es invocado cuando el actor provee una señal al sistema o cuando el sistema tiene un tiempo de evento. Una acción puede implicar señal de transmisión para invocar un actor u

otros actores. Una acción es atómica, esto significa que esta funcionando completamente o no es su totalidad.

◆ *Secuencia de Acciones:* la secuencia referida en la definición es un flujo de eventos específicos a través del sistema. Muchos flujos de eventos son posibles y muchos de ellos pueden ser muy parecidos. Para hacer un modelo de caso entendible, el grupo de flujo de eventos parecidos deben de estar dentro de un solo caso de uso.

◆ *Las funciones del sistema:* Esto significa que se está conciente con que el sistema hace en orden de funciones y la secuencia de acciones. El caso de uso ayuda tanto a definir una firma límite alrededor del sistema y que esta sea como la última palabra. Es esta forma se ayuda al límite de alcances del sistema.

◆ *Un resultado observable de valores:* La secuencia de acciones debe producir algo que tenga valor a un actor del sistema. Un actor podría no tener funciones críticas en el caso de uso como podría ser archivar algo útil. Enfocarse en valores útiles proveídos por un actor, por lo tanto hay que asegurarse que el caso de uso tenga relevancia y esté en un nivel que puede ser comprendido por el usuario.

◆ *Un actor en particular:* Enfocar esfuerzos en un actor en particular tanto para aislar el valor proveído por un grupo específico de usuarios del sistema, asegúrese de que el sistema hace lo que ellos necesitan.

La descripción de un Caso de Uso define que ocurre en el sistema cuando el caso de uso está funcionando. La funcionalidad del sistema está definida por un conjunto de diferentes casos de uso, cada cual representa un específico flujo de eventos. Para describir los casos de uso se utilizan plantillas o formularios como el mostrado en el Anexo I.

Modelo de caso de uso.

El modelo de caso de uso es un modelo de las funciones entendidas del sistema y su ambiente y esta sirve como contrato entre el usuario y el desarrollador. Este comprende un conjunto de todos los casos de uso para el sistema, junto con el conjunto de actores así que toda la funcionalidad del sistema es cubierta.

Casos de uso en el proceso de desarrollo.

El RUP es un acercamiento al manejo de los casos de uso. Esto significa que los casos de uso definidos por un sistema son la base para el desarrollo completo del proceso (ver figura 2).

Los casos de uso es el resultado de los requerimientos del workflow. En esta actividad con el cliente, el caso de uso es usado para capturar lo que el sistema debe de hacer desde el punto de vista del usuario. Así, el caso de uso actúa como el lenguaje común entre el usuario y el desarrollador del sistema.

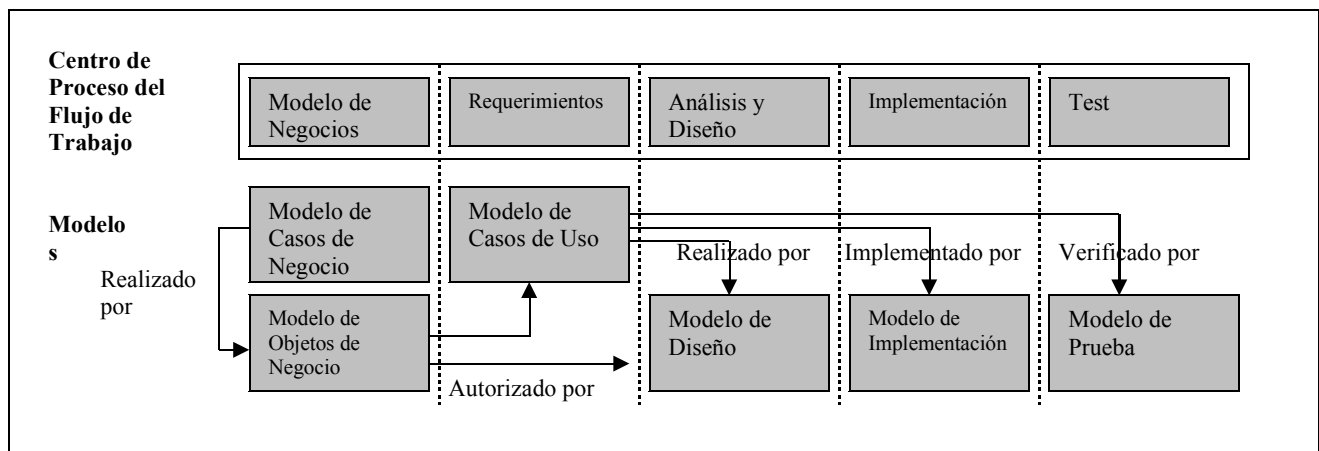


Figura 2. Flujo de Casos de Uso a través de varios modelos

En el análisis y diseño, el caso de uso es el puente que une requerimientos y actividades de diseño. Hay servicios tan básicos para la realización de los casos de uso, como describir que casos de uso son funcionales en términos de interacción de objetos en el diseño del modelo. Los objetos y las clases son más comúnmente encontradas a medida que se camina a través de los casos de uso. Esto técnicamente asegura que todas las conductas de los requisitos están representadas en el diseño del sistema.

Durante la implementación, el modelo de diseño es la especificación del modelo de implementación. Porque los casos de uso son la base del modelo de diseño, hay implementación en términos de diseño de clases. La realización de casos de uso en el modelo de diseño es usada para entender la dinámica del sistema y determinar a donde optimizar para mejor funcionalidad.

Durante las pruebas los casos de uso constituyen la base para identificar casos de prueba y procesos de prueba. En otras palabras, cada caso de uso es clasificado para verificar el sistema.

Otras actividades son también agregadas al caso de uso. Por ejemplo, porque el caso de uso especifica como un actor (un usuario) interactúa con el sistema, el caso de uso provee mucha de la estructura y contenido del manual de usuario.

En la administración de proyectos, los casos de uso y escenarios son usados para definir el contenido de interacciones. Las estimaciones de esfuerzos pueden ser derivadas de la descripción de los casos de uso por técnicas como función de análisis de puntos. En el desarrollo, los paquetes de casos de uso pueden servir para planear una fase de desarrollo o para definir varios sistemas.

1.3. Diagramas de Flujo de Datos.

Los Diagramas de Flujo de Datos (DFD) representan gráficamente los procesos y flujos de datos en un sistema de negocios. En su estado original los DFD muestran el panorama más amplio posible de entradas, procesos y salidas de un sistema de información.

Los Diagramas de Flujo de Datos, tienen como finalidad poder dar a entender el flujo de la información entre diferentes entidades, detallando los cambios que esta información pueda tener en los diferentes procesos descritos en los mismos; en este tipo de diagrama se visualizan los responsables (entidades) y su rol en el sistema de información.

Ventajas de los DFD.

El enfoque de flujo de datos tiene cuatro ventajas principales sobre la explicación narrativa de la forma en que se mueven los datos a través del sistema, estas ventajas son:

- ◆ Libertad para realizar en forma muy temprana la implementación técnica del sistema.

- ◆ Una mayor comprensión de las interrelaciones de los sistemas y subsistemas.

- ◆ Comunicación del conocimiento del sistema actual a los usuarios por medio de diagramas.

- ◆ Análisis de un sistema propuesto para determinar si han sido definidos los datos y procesos necesarios.

Convenciones utilizadas en los DFD.

Se utilizan cuatro símbolos básicos para diagramar el movimiento de los datos en los diagramas de flujos de datos, estos símbolos y su significado se muestran a continuación:

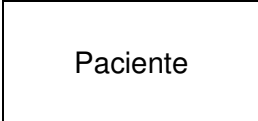
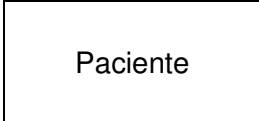

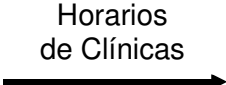
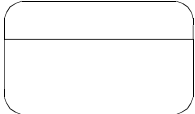
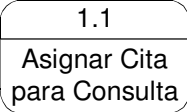


Símbolos	Significado	Ejemplo
	Entidad	
	Flujo de Datos	
	Proceso	
	Almacén de Datos	

Tabla 1. Convenciones Utilizadas en los DFD

2. MARCO TEÓRICO.

2.1. Internet.

La Internet es una red de computadoras conectada a cientos de redes de computadoras en todo el mundo. En un inicio no se sabía la magnitud de este invento, sin embargo, es hoy en día el medio de comunicación más avanzado que existe.

El número total de usuarios de la Internet asciende hoy en día a varios millones, y su crecimiento es exponencial. Este alto nivel de conectividad ha creado un grado de comunicación, colaboración, acceso a la información e intercambio de recursos sin precedentes en la historia de la humanidad.

La Internet como si sólo no pertenece a uno o varios grupos, es decir, nadie está a cargo de la Internet. Hay organizaciones que desarrollan aspectos técnicos de esta red, pero ningún cuerpo del gobierno o alguna organización está en mando.

Todas las computadoras en la Internet comunican una con otra, usando el Protocolo de Control de Transferencia / Protocolo de Internet, sus siglas TCP/ IP. Las computadoras en la Internet usan arquitectura cliente/ servidor, esto significa que la máquina del servidor remoto proporciona archivos y servicios a la máquina del cliente local del usuario.

Un usuario de Internet tiene acceso a una gran variedad de servicios: Correo electrónico, transferencia de archivos, inmensos recursos de información, número de miembros del grupo del interés, colaboración interactivo, multimedia, radiodifusión en tiempo real, oportunidades de compra y venta, y muchas cosas más.

Principales Servicios de Internet:

World Wide Web (WWW):

Es un sistema de información que integra todos los recursos accesibles por Internet en páginas de información y se basa en un protocolo llamado HTTP, que permite hacer ligas de un documento, publicado en Internet, a otro, haciendo la búsqueda de información bastante amigable.

El WWW es un sistema de información multimedia que permite la creación de documentos hipertexto, de libros electrónicos con aplicaciones como revistas informativas, documentos de referencia y consulta, formación o soporte técnico, etc.; es decir, desarrollar aplicaciones interactivas con soporte de entrada de datos en formularios. Esto permite desarrollar aplicaciones de tipo interactivo sobre cualquier navegador (browser).

Correo Electrónico:

Es de los servicios más utilizados en la Internet. Permite transmitir texto y archivos de una manera sencilla, como un correo normal, sólo que más rápido que un telegrama (suponiendo que la red esté funcionando de manera adecuada). El correo electrónico permite el intercambio rápido de mensajes entre personas o grupos de personas del sistema de comunicación.

Telnet:

Este es otro de los primeros servicios de Internet. Sirve para establecer una sesión en una computadora remota, prácticamente como si estuviera en una red local.

Telnet es el protocolo o servicio que permite iniciar sesiones de trabajo en máquinas remotas conectadas a la Internet. El Telnet permite el teletrabajo, ya que permite, a través de un terminal lejano al puesto de trabajo, acceder a toda la información, recursos informáticos y aplicaciones de la máquina situada en el centro de trabajo, tal y como si se estuviera frente a ella. El Telnet permite también acceder a bancos de datos emulando un terminal que estuviera directamente enlazado al ordenador central. Obviamente para ingresar al sistema remoto se debe conocer un nombre de

usuario y su respectivo password. En algunos sistemas hay cuentas para huéspedes o usuarios anónimos con las que se puede contar con parte de los recursos de la computadora remota. Cada sistema tiene sus propias aplicaciones que permiten usar ciertos recursos.

File Transfer Protocol (FTP):

Uno de los protocolos más útiles es el FTP, Protocolo de Transferencia de Archivos, que sirve para transmitir archivos de todo tipo a través de la Red. Generalmente hay servidores llamados de FTP anónimo, de donde se puede “bajar” (transferir a una computadora) archivos públicos. En este tipo de sistemas, el nombre de usuario que se utiliza es anónimo y la contraseña es la dirección del correo electrónico. Hay otros sistemas en los que es necesario un nombre de usuario y su respectiva contraseña.

2.2. Intranet.

Se llama Intranet a las redes tipo Internet pero que son de uso interno, por ejemplo, una red corporativa de una empresa que utilizara protocolo TCP/IP y servicios similares como World Wide Web.

Una Intranet puede resolver muchos problemas, como por ejemplo, el problema de la distribución de información, problemas comunicación en el ambiente en donde se desarrollará la Intranet, seguridad en el manejo de información, etc.

Se puede publicar fácilmente cualquier documento de una Intranet en la Internet, ya que una Intranet no es mas que una Internet en pequeño. Además se puede dar acceso a Internet a los usuarios de una Intranet. Las ventajas de una intranet son muchas, siendo además una herramienta esencial de trabajo para medianas y grandes empresas. El futuro en administración y gestión interna de empresas está aquí, en la Intranet.

2.3. Tecnologías.

A continuación se describen las principales tecnologías utilizadas para la elaboración del presente proyecto:

2.3.1. Arquitectura de las Aplicaciones.

Hoy en día, las aplicaciones computacionales están estructuradas en porciones de elementos que son ejecutados en una o varias computadoras conectadas entre si, a la forma en que estos elementos se encuentran distribuidos se le conoce como *Arquitectura de Aplicaciones* y a dichos elementos distribuidos se les conoce como *Capas*.

Las capas son un concepto lógico que describen como las aplicaciones pueden ser segmentadas en *Servicios*, específicamente tres tipos de servicios: servicios de usuario, servicios de reglas del negocio y servicios de datos. Los tres tipos de capas pueden ser generalmente descritos como capa de usuario (primera capa), capa de reglas del negocio (segunda capa) y capa de datos (tercera capa). El concepto de capas se enfatiza en la segmentación lógica de los servicios y no en el número de computadoras en las que son implementados dichos servicios.

Arquitectura de Una Capa (Single-Tier)

Las aplicaciones basadas en esta arquitectura se caracterizan por tener los tres tipos de servicios (usuario, reglas del negocio y datos) residiendo en una sola computadora, es decir que estas aplicaciones poseen programas que se ejecutan en la computadora del usuario y estos pueden también comunicarse con una base de datos siempre y cuando se encuentre en la misma computadora.

Normalmente cada instalación de una aplicación de esta arquitectura es usada solamente por una persona.

Entre las principales desventajas que presentan este tipo de aplicaciones es que son resistentes al cambio, no utilizan una tecnología abierta y su arquitectura es costosa e inflexible.

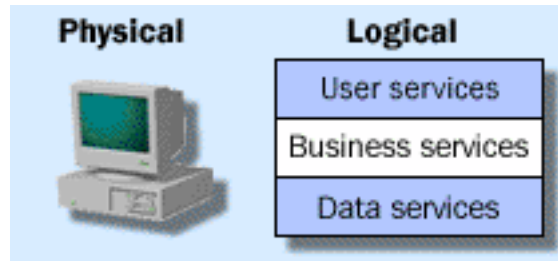


Figura 3. Esquema de Arquitectura de una capa.

Arquitectura de Dos Capas (Two-Tier)

Las aplicaciones que se basan en este tipo de arquitectura son conocidas también como aplicaciones cliente-servidor. En estas aplicaciones los servicios de usuario y de reglas del negocio residen en una máquina conocida como cliente mientras que la base de datos se encuentra en otra conocida como servidor; en este tipo de aplicaciones el peso del cálculo recae en el servidor, mientras que el cliente hace la parte menos pesada, y por eso los clientes suelen ser máquinas menos potentes que los servidores. Uno de los grandes problemas en este tipo de aplicación es el mantenimiento de las aplicaciones, ya que las modificaciones hechas en una máquina cliente han de ser trasladada a todos los clientes.

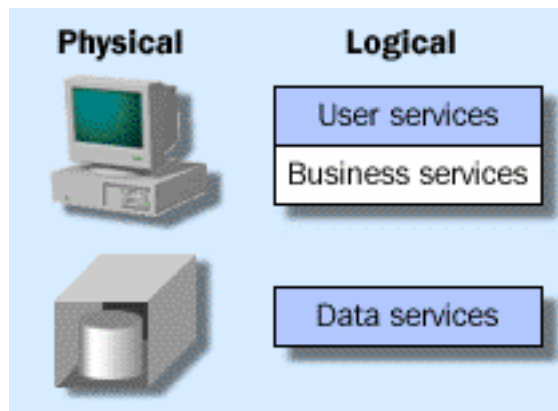


Figura 4. Esquema de Arquitectura de dos capas.

Arquitectura de Tres Capas (Three-Tier).

Para solucionar los diversos problemas de las aplicaciones cliente – servidor se ha desarrollado el concepto de arquitecturas de tres capas: interfaz de usuario, reglas de negocios y datos, una de las muchas ventajas que presenta este tipo de arquitectura es la capacidad de extraer la capa de reglas de negocios de la capa de usuarios y la de datos.

La primera capa de este tipo de arquitectura (capa de usuario) consiste en la interfaz de presentación, es decir todo lo concerniente a la parte que visualizará el usuario de la aplicación y el medio a través del cual interactuará con los datos y elementos de la aplicación; la capa intermedia (capa de reglas del negocio) es el código que el usuario invoca para recuperar los datos deseados, mientras que la tercera capa consiste en los datos que gestiona la aplicación. La capa de presentación recibe los datos y los formatea para mostrarlos adecuadamente. Esta división entre la capa de presentación y la de reglas del negocio permite una gran flexibilidad a la hora de construir aplicaciones, ya que se pueden tener múltiples interfaces sin cambiar la lógica de la aplicación.

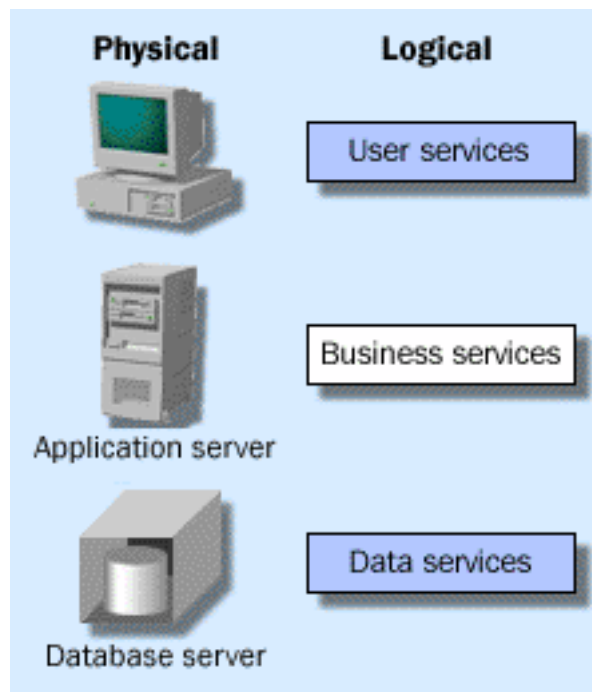


Figura 5. Esquema de Arquitectura de tres capas.

Arquitectura de Aplicaciones Web.

La arquitectura de las aplicaciones Web suelen presentar un esquema de tres capas: La primera capa es de presentación al usuario, que incluye no sólo el navegador, sino también el servidor web que es el responsable de dar a los datos un formato adecuado. La segunda capa está referida habitualmente a algún tipo de programa o *script*. Finalmente, la tercera capa proporciona a la segunda los datos necesarios para su ejecución.

Una aplicación Web típica recogerá datos del usuario (primera capa), los enviará al servidor, que ejecutará un programa (segunda capa) y cuyo resultado (tercera capa) será formateado y presentado al usuario en el navegador (primera capa otra vez).

En una arquitectura cliente-servidor la aplicación envía solicitudes al servidor de datos a través de una red privada, generalmente algún tipo de LAN o WAN, dicho servidor procesa estas solicitudes y regresa la información solicitada. Es la aplicación la que interpreta y despliega totalmente estos datos al usuario. Mientras que en una arquitectura de WEB, una página HTML es la interfaz con el usuario dentro del *web browser*, esta página fue preparada y enviada al usuario por el servidor de Internet; cuando el usuario llena un formulario y lo envía, la información llega al Servidor vía un método de HTML, entonces el servidor puede utilizar diversos componentes para procesar los datos de la Base de Datos y regresar al usuario una página de resultados. Nótese que en el esquema de la arquitectura Web, no existe procesamiento lógico en la parte cliente, el *browser* simplemente despliega las páginas HTML y transporta los datos hacia el/del servidor de internet.

Otra diferencia clave en este escenario de Web es que la comunicación entre el *browser* y el servidor se lleva a cabo sobre el protocolo HTTP, donde no existe un conexión persistente entre clientes y servidor. Esto conlleva varias ventajas, especialmente en términos de escalabilidad, pero introduce también nuevos obstáculos como la necesidad de que la aplicación controle el estado usuario/sesión durante la navegación entre las diferentes páginas.

Las características claves de la arquitectura de una aplicación de Web son:

- ◆ El ambiente de la aplicación es abierto, público (aunque las Intranets puede ser privadas) y distribuido.
- ◆ Todas las operaciones son ejecutadas dentro de un contenedor externo: el *browser*.
- ◆ La arquitectura es inherentemente distribuida y multicapas. La mayor parte de la lógica de negocio es implementada por el *back-end* y donde los componentes de negocio del servidor se comunican con el RDBMS.
- ◆ Las conexiones entre cliente y servidor son intermitentes y HTTP es el protocolo de transporte utilizado para pasar datos entre el servidor y el *browser*.

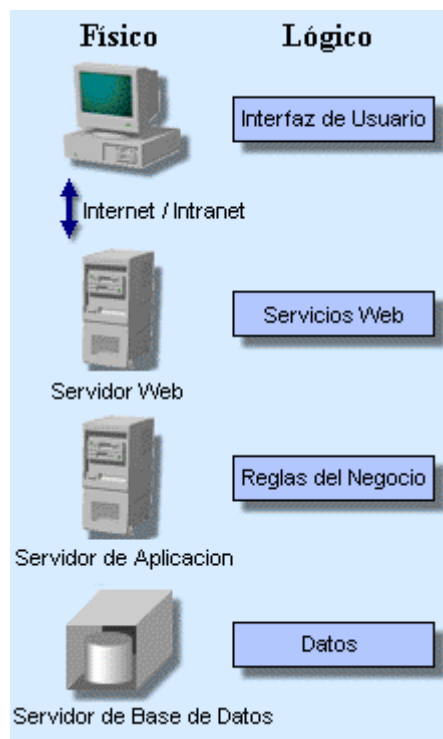


Figura 6. Esquema de Arquitectura de aplicaciones web.

2.3.2. Windows 2000 Server.

Windows 2000 Server es un innovador sistema operativo, construido con base en tecnología NT, que está diseñado para cubrir necesidades en todo tipo de empresas, desde organizaciones pequeñas y centralizadas, hasta las empresas distribuidas más grandes. Basado en los puntos principales de Windows NT Server 4.0, Windows 2000 Server establece un nuevo estándar acerca de lo bien que se puede integrar un sistema operativo con servicios de directorios, aplicaciones Web, redes, archivos e impresión (todos basados en estándares), así como con la administración y confiabilidad eficaz de un extremo a otro, para proporcionar la mejor base que permita integrar a una empresa en Internet.

La familia Windows 2000 Server se basa en los principios fundamentales de Windows NT y se compone de los productos siguientes:

- ◆ *Windows 2000 Server*, es la versión siguiente de Windows NT Server 4.0. Se trata del sistema operativo de red para múltiples propósitos destinado a todo tipo de empresas.
- ◆ *Windows 2000 Advanced Server*, es la versión siguiente de Windows NT Server, Enterprise Edition. Es el sistema operativo para aplicaciones de línea de empresa y comercio electrónico.
- ◆ *Windows 2000 Datacenter Server*, es el miembro más reciente de la familia Windows Server. Es el sistema operativo para las soluciones empresariales que demandan el nivel más alto de escalabilidad.

Características:

A continuación se presentan las principales características innovadoras de Windows 2000 Server:

- ◆ *Servicios de Internet Information Server 5.0 (IIS)*: Los servicios Web integrados permiten a los usuarios alojar y administrar fácilmente sitios Web para compartir información, crear aplicaciones empresariales basadas en Web y ampliar al Web los servicios de archivos, impresión, medios y comunicaciones.

- ◆ *Entorno de programación de Páginas Active Server (ASP):* Se pueden crear aplicaciones que permitan al servidor Web intercambiar datos con formato XML con Microsoft Internet Explorer y cualquier servidor capaz de analizar XML.
- ◆ *Windows DNA 2000:* La arquitectura de Aplicaciones distribuidas de red de Windows (Windows DNA 2000), el modelo de desarrollo de aplicaciones para la plataforma Windows, permite crear soluciones seguras, confiables y altamente escalables que faciliten la integración de las aplicaciones y los sistemas heterogéneos.
- ◆ *Modelo de objetos componentes + (COM+):* COM+ se basa en las características y los servicios integrados de COM, lo que permite fácilmente a los programadores crear y utilizar componentes de software en cualquier lenguaje y con cualquier herramienta.
- ◆ *Servicios de transacciones:* Se trata de un sistema de procesamiento de transacciones basado en componentes para desarrollar, instalar y administrar aplicaciones de servidor, intranets, Internet y empresas robustas, escalables y de alto rendimiento.
- ◆ *Plataforma multimedia:* Mediante los Servicios de Windows Media integrados, se puede configurar y administrar contenido de medios digitales de alta calidad a través de Internet y las intranets.
- ◆ *Aplicaciones habilitadas para directorios:* Los programadores pueden usar un número de interfaces estándar para escribir aplicaciones que utilicen información almacenada en Active Directory acerca de los usuarios, otras aplicaciones y dispositivos. Esto permite disponer de aplicaciones dinámicas y ventajosas que son más fáciles de desarrollar y administrar.
- ◆ *Carpetas Web:* Las carpetas Web aportan la riqueza de Windows al Web mediante el uso de Creación y versión de documentos Web (WebDAV) para permitir arrastrar y soltar publicación en Web.
- ◆ *Rendimiento de ASP mejorado:* Un procesamiento de Páginas Active Server (ASP) más escalable, control de flujo de ASP mejorado y ruta rápida de ASP para los archivos ASP sin secuencias de comandos, permiten un procesamiento de páginas Web más rápido.

- ◆ *Alojamiento de sitios múltiples:* Los servicios de Internet Information Server (IIS5.0) permiten alojar más sitios Web por servidor con un gran rendimiento.
- ◆ *Limitación de CPU de IIS:* Limita la cantidad de tiempo que una aplicación o sitio Web puede utilizar la CPU para asegurar que el tiempo del procesador está disponible para otros sitios Web o aplicaciones que no son de Web, lo que supone además una mejora del rendimiento.
- ◆ *Máximo rendimiento y utilización del ancho de banda:* La compatibilidad de Windows 2000 Server con redes de hasta 1 GB, proporciona procesamiento al máximo rendimiento en redes de alto rendimiento. De esta manera, se aumenta el rendimiento sin necesidad de aumentar el ancho de banda de la red.
- ◆ *Autenticación Kerberos:* La compatibilidad total con el protocolo Kerberos versión 5, proporciona rápidos inicios de sesión únicos a los recursos de Windows, así como otros entornos que admiten este protocolo.
- ◆ *Sistema de cifrado de archivos:* Permite cifrar los datos del disco duro para aumentar la seguridad. Estos datos permanecen cifrados incluso si se realizan copias de seguridad o se archivan.
- ◆ *Comunicaciones de red seguras:* Se trata de comunicaciones cifradas de un extremo a otro a través de la red de la compañía mediante el estándar IPsec. Ideal para proteger las comunicaciones internas más importantes de intrusiones accidentales o intencionadas. Active Directory proporciona un control central de directiva de uso para permitir su distribución.
- ◆ *Traducción de direcciones de red (NAT):* Los servicios de traducción de direcciones de red extensibles refuerzan la instalación de la red, la seguridad de red mejorada y la escalabilidad aumentada de una dirección de red de grupo disponible.
- ◆ *Servicio de enrutamiento y acceso remoto:* Conecta trabajadores remotos, trabajadores a distancia y sucursales de la red corporativa a través de acceso telefónico, línea alquilada y vínculos a Internet.
- ◆ *Telefonía IP:* La compatibilidad con paquetes de datos telefónicos, integración de sitios Web en Internet y los servicios complementarios de software de cliente, proporciona una plataforma para unas mejores comunicaciones y comercio en

Internet. Integrada ahora con Calidad de servicio para permitirle controlar la forma de uso de este servicio en la red.

- ◆ *Alto rendimiento en redes rápidas:* Mediante compatibilidad con el Modo de transferencia asincrónica (ATM) y las redes de 1 Gigabit, Windows 2000 Server proporcionará un alto rendimiento incluso en las redes más rápidas.
- ◆ *Protección contra escritura del modo núcleo:* Ayuda a evitar que código errante pueda interferir en el funcionamiento del sistema.
- ◆ *Protección de archivos de Windows:* Evita que las instalaciones de programas nuevos sustituyan archivos de sistema importantes.
- ◆ *Certificación de controladores:* Permite identificar los controladores de dispositivos que han pasado las pruebas realizadas en las instalaciones de control de calidad del hardware de Windows y advierte a los usuarios en caso de que vayan a instalar un controlador sin certificar.
- ◆ *Funciones de red:* Windows 2000 Server utiliza dispositivos de red compatibles con las tecnologías de red más recientes, incluidas Plug and Play, DSL, VPN, enrutamiento, NAT, DHCP, enrutadores y conmutadores de calidad de servicios, dispositivos de red compatibles con directorios, IPsec, SSL y Modo de transferencia asincrónica.
- ◆ *Dispositivos periféricos:* Windows 2000 Server funciona con los periféricos más recientes como impresoras USB, adaptadores de red, *mouse* (ratones) y teclados. Ofrece compatibilidad avanzada con controladores de impresora, así como con dispositivos digitales e infrarrojos, PCMCIA y 1394.
- ◆ *Servicios de directorio:* Entre las nuevas características de servicios de directorio destacan: Protocolo compacto de acceso a directorios (LDAP); uso de nombres estándar; basado en DNS, el servicio de localizador usado en Internet y en la mayoría de las intranets; Interfaz de servicio de Active Directory (ADSI); esquema extensible; catálogo global; replicación de múltiples maestros y compatibilidad con versiones anteriores.
- ◆ *Seguridad:* Entre las nuevas características de seguridad destacan: Administrador de configuración de seguridad, autenticación Kerberos, PPTP/L2TP,

Servidor de certificados de clave pública, infraestructura de tarjeta inteligente y protocolo de seguridad IP.

◆ *Interfaces de administración:* Entre las nuevas características de interfaces de administración destacan: Microsoft Management Console; Herramienta de migración de servicios de directorio de Microsoft (complemento de MMC); Administración de equipos (complemento de MMC); Administración de discos (complemento de MMC); Administración del servicio de sistemas (complemento de MMC); Administrador de dispositivos y Asistente para hardware (complemento de MMC); Directiva de grupo; Windows Scripting Host (WSH); y Programador de tareas.

◆ *Instalación de equipos:* Mediante el uso de tecnología de inicio remoto (PXE) basada en estándares, un equipo puede conectarse automáticamente a un servidor Windows 2000 Server e instalar Windows 2000 Professional. El servicio de instalación remota del sistema operativo se puede utilizar para configurar un nuevo equipo, actualizar un equipo a Windows 2000 Server o volver a formatear e instalar el sistema operativo en un equipo existente.

◆ *Instrumental:* El Instrumental de administración de Windows proporciona compatibilidad totalmente integrada con el sistema operativo para la administración uniforme de aplicaciones y empresas basada en el Modelo común de información (CIM) adoptado por Desktop Management Task Force (DMTF) como parte de la iniciativa Administración de empresas basada en Web (WBEM, Web-Based Enterprise Management). El Instrumental de administración de Windows simplifica la instrumentación de los controladores y las aplicaciones, lo que supone un mayor control y ofrece el potencial a un costo de propiedad más bajo.

◆ *Servicios de Terminal Server:* Entre las nuevas características de servicios de Terminal Server destacan: compatibilidad con varios clientes; compatibilidad con desconexiones móviles; compatibilidad con inicios de sesión múltiples; compatibilidad con sistemas de archivos distribuidos; administración de Servicios de Terminal Server y administración de control remoto; configuración de Servicios de Terminal Server; integración con el Monitor de rendimiento de Windows 2000 Server; tiempo de espera de inactividad configurable; varios niveles de cifrado.

- ◆ *Herramientas de instalación y de solución de problemas:* Entre las nuevas características de instalación y solución de problemas destacan: Recuperación del sistema avanzada (ASR); Administrador de componentes opcionales; instalación en modo a prueba de errores; duplicación de disco; Asistente para incompatibilidad de controladores; Asistente para configuración de servidor; instalación desatendida; Consola de comandos de reparación; actualización automática de Service Packs; compatibilidad con discos dinámicos.
- ◆ *Administración del servicio de archivos:* El complemento MMC de administración del sistema de archivos permite a los usuarios crear recursos compartidos y administrar las sesiones y conexiones de equipos locales o remotos. Sustituye la funcionalidad que anteriormente se encontraba en el Panel de control del sistema. Además de sus capacidades remotas, permite también al usuario crear recursos compartidos para cualquiera de los servicios de archivos instalables ofrecidos por Microsoft: servicios de impresión y archivos tanto para Macintosh como para NetWare.
- ◆ *Servicios de impresión:* Entre las nuevas características de servidor de impresión destacan: integración con Active Directory; Protocolo de impresión en Internet (IPP); interfaz de usuario simplificada y mejorada; impresión de alta disponibilidad.
- ◆ *Administración de publicación en Web:* Entre las mejoras realizadas en la administración de publicación en Web destacan: instalación y actualización integradas, información sobre procesos, limitación de la CPU, dominios de usuarios múltiples, Asistente para certificados, Asistente para permisos, panel de tareas MMC y secuencias de comandos mejoradas de administración de línea de comandos.
- ◆ *Servicios de Windows Media Server:* Entre los nuevos servicios de Windows Media Server destacan: distribución a través de protocolos HTTP, autenticación del proxy de distribución, simplificación y cambios de secuencia, reenvío de UDP, capacidad de limitar las conexiones de cliente por punto de publicación, capacidad para limitar el ancho de banda total por punto de publicación, salida a formato de archivos W3C, análisis e informes basados en analistas, modelo de autorización

conectable, ejemplo de autorización a Commerce Server y notificación de sucesos, modelo de autenticación conectable, seguridad por punto de publicación, etc.

- ◆ *Sistema de archivos:* Entre las mejoras realizadas en el nuevo sistema de archivos destacan: mejoras en NTFS, sistema de archivos de cifrado, seguimiento de vínculos distribuidos, cuotas de disco y archivos dispersos.

- ◆ *Redes y comunicaciones:* Entre las nuevas características de redes y comunicaciones de Windows 2000 Professional Y Windows 2000 Server se incluye compatibilidad con: Seguridad IP (IPSec); Protocolo de túnel de nivel 2 (L2TP); Protocolo de túnel punto a punto (PPTP); IPX/SPX; AppleTalk y las nuevas mejoras TCP/IP; y H.323.

- ◆ *Autenticación, cifrado y seguridad de plataformas de red:* Las nuevas características de autenticación, cifrado y seguridad de plataformas de red incluyen: CHAP, MS-CHAP, PAP; Protocolo de autenticación extensible (EAP); compatibilidad con tarjetas inteligentes (EAP-TLS); servicios de cifrado RC4; cliente RADIUS, RFC 2138; servidor RADIUS; filtrado de paquetes IP y filtrado de paquetes IPX.

- ◆ *Servicios de puerta de enlace y protocolos de enrutamiento:* Entre los nuevos servicios de puerta de enlace y protocolos de enrutamiento destacan: traductor de direcciones de red (NAT); IGMP versión 2; RIP v2 (y v1) para IP; OSPF; proxy DNS; agente de retransmisión DHCP; RIP y SAP para IPX; y enrutamiento estático.

- ◆ *Servicios de componentes para aplicaciones distribuidas:* El modelo de objetos componentes (COM+) de Windows 2000 Server simplifica notablemente la creación y el uso de los componentes de software. COM+ proporciona tiempo de ejecución y servicios que pueden utilizarse desde cualquier lenguaje o herramienta de programación y permite una amplia interoperabilidad entre componentes, independientemente de cómo se implementaron.

- ◆ *Servicios distribuidos de aplicaciones Web:* Entre las nuevas características de servicios distribuidos de aplicaciones Web destacan: páginas Active Server; limitación de CPU e información sobre procesos.

- ◆ *Alta interoperabilidad con equipos cliente:* Ofrece compatibilidad con los sistemas operativos Windows NT Workstation, Windows 9x, Windows 3.x, Unix y

Macintosh. La compatibilidad de Appleshare con TCP/IP mejora el uso compartido de recursos para el sistema operativo Macintosh.

- ◆ *Servicio de autenticación de Internet:* La autorización, administración de cuentas y autenticación RADIUS permiten administrar de manera centralizada las directivas de acceso a la red y mejoran la administración de cuentas y el diseño de la capacidad del costo de red. Ahora incluye administración de directivas mejorada que le permite controlar quiénes pueden tener acceso a qué tipos de información, cuándo y cómo.

- ◆ *Programador de tareas:* Permite programar la secuencia de comandos, programa o documento que se desee para que pueda invocarse en cualquier momento o intervalo, ya sea durante el inicio del sistema, un inicio de sesión de usuario o un proceso de inactividad del sistema.

- ◆ *DNS dinámico:* El servicio DNS (Sistema de nombres de dominio) basado en estándares de Internet e integrado en Active Directory, simplifica el proceso de nomenclatura y localización de objetos mediante protocolos de Internet, y mejora la escalabilidad, el rendimiento y la interoperabilidad. Los sistemas que reciben direcciones desde un servidor de Protocolo de configuración dinámica de host (DHCP), se registran automáticamente en DNS. Las opciones de replicación con sistemas DNS heredados y a través de Active Directory, pueden simplificar y reforzar la infraestructura de replicación de nombres.

2.3.3. Internet Information Server (IIS) 5.0.

Internet Information Server es un grupo de servidores de Internet (incluyendo uno web o un servidor Hypertext Transfer Protocol y un servidor con Protocolo de Transferencia de Archivos - FTP) con capacidades adicionales para el Sistema

Operativo Windows 2000 Server. Con IIS, Microsoft incluye un set de programas para construir y administrar sitios web, motores de búsqueda, y soporte para hacer aplicaciones basadas en web que acceden a bases de datos.

Las empresas buscan tecnologías de Web basadas en estándares para, de una manera más eficiente, intercambiar información con clientes, socios y empleados de todo el mundo. Además, intentan implementar soluciones que aprovechen al máximo las inversiones existentes en conocimientos, formación y tecnología. Microsoft Internet Information Server (IIS) está diseñado para cubrir estas necesidades a un amplio rango de usuarios, desde grupos de trabajo y departamentos de una intranet corporativa hasta proveedores de servicios Internet que alojan sitios Web que reciben millones de visitas diarias. IIS 5.0 revoluciona las capacidades de Web del sistema operativo Microsoft Windows 2000 Server, proporcionando la forma más fácil de compartir información, crear y distribuir aplicaciones de negocio, y alojar y administrar sitios.

Las características innovadoras de publicación de Web, herramientas personalizables y tecnologías de nuevos asistentes, hacen de Windows 2000 Server con IIS 5.0 la forma más fácil de publicar y compartir información, de forma segura, a través de intranets corporativas y a través de Internet.

Las herramientas personalizables de administración, opciones flexibles de administración y herramientas de análisis hacen de Windows 2000 Server con IIS 5.0 el servidor de Web más fácil de administrar.

Características:

A continuación se presentan las principales características del Internet Information Server:

- ◆ *Protección contra errores graves:* La protección contra errores graves permite a los clientes ejecutar varias aplicaciones en el servidor Web con fiabilidad. Si una aplicación produce un error, el servidor Web y las demás aplicaciones continuarán ejecutándose y la aplicación que falló se reinicia con la siguiente petición.
- ◆ *Active Server Pages con transacciones:* IIS 5.0 presenta Transactional Active Server Pages, que permiten a las aplicaciones con secuencias de comandos y componentes realizar múltiples acciones, aceptando todas las acciones simultáneamente, o sin aceptar ninguna. Esto es decisivo para las aplicaciones de bases de datos.
- ◆ *Depuración de secuencias de comandos:* IIS 5.0 ofrece una solución integrada para depurar secuencias de comandos y aplicaciones de Active Server Pages, lo que facilita la programación de aplicaciones ASP.
- ◆ *Cola de mensajes integrada:* Microsoft Message Queue Server (MSMQ) ofrece una forma fácil para que las aplicaciones envíen y reciban mensajes a través de una red con fiabilidad, incluso en el caso en que parte de la aplicación, o la red, no estén disponibles.
- ◆ *Compatibilidad con Java:* IIS 5.0 incluye un conjunto de clases de Java para crear componentes del lado del servidor y Java 1.1 Virtual Machine de Microsoft. Si utiliza Active Server Pages y Microsoft Transaction Server, puede crear aplicaciones Java y ejecutarlas fácilmente en entornos de servidor.
- ◆ *Múltiples sitios Web:* Se admiten múltiples sitios Web en una única dirección IP, operadores de sitios Web, ajuste de ancho de banda por cada sitio Web y administración mejorada y personalizable, que permite a las organizaciones alojar múltiples sitios de intranets de departamentos o múltiples sitios públicos de Internet.
- ◆ *Estándares de Internet:* Se incorporan los estándares existentes y los más recientes de la industria, incluidos: HTTP 1.1 para un mayor rendimiento de Internet, administrando y emitiendo certificados digitales X.509 y enviando mensajes a servidores estándar de noticias NNTP y de correo SMTP.
- ◆ *Compatibilidad con administración automatizada:* Con los nuevos objetos de administración de IIS puede escribir secuencias de comandos y ejecutarlas desde la línea de comandos, automatizando diferentes tareas de administración.

- ◆ *Herramientas flexibles de administración:* Las herramientas flexibles de administración del servidor Web IIS 5.0 incluyen una consola integrada de administración basada en Windows, administración mejorada basada en el explorador, secuencias de comandos ejecutables desde la línea de comandos, así como interfaces programables para crear herramientas de administración personalizadas.
- ◆ *Seguridad:* Se incluye un servidor de certificados que está estrechamente integrado con el modelo de seguridad de Windows 2000 Server, lo que permite a las organizaciones emitir y administrar certificados digitales X.509 estándar de Internet.
- ◆ *Motor de Búsqueda:* Incluye capacidades de búsqueda integradas, que permiten a los usuarios crear formularios de búsqueda personalizados con Active Server Pages, objetos de datos ActiveX y consultas SQL para buscar información en el servidor Web.

2.3.4. SQL Server 2000.

Microsoft SQL Server 2000 Enterprise Edition es una oferta completa de base de datos y análisis para poder pasar rápidamente a la próxima generación de soluciones escalables de comercio electrónico, empresariales y de almacenamiento de datos.

Microsoft SQL Server 2000 es totalmente preparado para Web, ya que permite consultar, analizar y manipular datos a través del Web. Se puede utilizar el lenguaje de marcado extensible (XML) en SQL Server 2000 para intercambiar datos entre sistemas débilmente conectados. Además se puede tener acceso a los datos de manera fácil y segura desde un explorador, a través de servidores de seguridad, y se puede realizar búsquedas rápidas de texto en documentos con formato. Permite también analizar y vincular cubos de Procesamiento analítico en línea (OLAP), incluso a través del Web.

Microsoft SQL Server 2000 es muy escalable y confiable, ya que permite que el usuario crezca sin límites con las características mejoradas de escalabilidad y

confiabilidad. Permite repartir la carga de trabajo de la base de datos para poder escalar las aplicaciones.

Características:

A continuación se presentan las principales características innovadoras del Microsoft SQL Server 2000:

- ◆ *Compatibilidad con los estándares de Internet:* Permite integrar los sistemas existentes sin necesidad de programación, gracias a la completa compatibilidad con estándares de W3C, incluyendo XML, Xpath, XSL y HTTP.
- ◆ *Vistas XML:* Acceso a datos relacionales mediante técnicas XML asignando fácilmente elementos y atributos XML al esquema relacional.
- ◆ *Acceso mediante direcciones URL y http:* Facilita el acceso a datos de manera flexible desde el Web mediante una dirección URL. Utilice el Lenguaje de consulta estructurado (SQL), plantillas XML o XPath en las direcciones URL para hacer consultas.
- ◆ *SELECT ... FOR XML:* Obtiene código XML de las consultas SQL y controla fácilmente la forma del árbol XML devuelto mediante tres opciones de formato.
- ◆ *Extracción de datos integrada:* Permite Analizar datos relacionales y OLAP recopilados, incluyendo las páginas visitadas por los usuarios e historiales de compra para reconocer tendencias y hacer predicciones.
- ◆ *Accesos http a cubos:* Facilita que los cubos estén disponibles para las aplicaciones de análisis basadas en Web a través de servidores de seguridad mediante un sistema de escucha HTTP integrado en SQL Server 2000 Enterprise Edition.
- ◆ *Compatibilidad con sistemas de multiprocesamiento simétrico (SMP):* Es posible escalar bases de datos de SQL Server 2000 a sistemas SMP con hasta 32 procesadores.
- ◆ *Almacenamiento de dimensiones ROLAP:* Se pueden realizar análisis en dimensiones enormes (más de 10000 miembros) almacenando esas dimensiones de manera relacional.

- ◆ *Cubos de particiones distribuidas:* se puede utilizar fácilmente el Asistente para particiones con el fin de separar un cubo lógico en varias particiones físicas con total transparencia para conseguir un almacenamiento de datos flexible y un mejor rendimiento de las consultas.
- ◆ *Acciones OLAP:* Permite que los resultados de análisis inicien procesos automáticamente en aplicaciones Web y empresariales.
- ◆ *Editor de cubos virtuales:* Permite explorar, examinar y modificar cubos virtuales de manera gráfica.
- ◆ *Nuevos tipos de datos:* Se puede almacenar y hacer referencia a datos de manera flexible con los tipos de datos bigint, sql_variant y de tabla.
- ◆ *Búsqueda de texto:* Permite la búsqueda de texto en documentos con formato (por ejemplo, Word, Excel y HTML) en el Web y en intranets.
- ◆ *English Query:* Permite a todos los usuarios tener acceso a los datos mediante consultas en lenguaje natural (sólo está disponible en inglés).
- ◆ *Seguridad:* Permite Proteger los datos con una mayor seguridad predeterminada en la instalación. Incluye compatibilidad con conexiones Secure Sockets Layer (SSL) y Kerberos.

2.3.5. Active Server Pages.

Active Server Pages (ASP), es una tecnología propietaria de Microsoft. Se trata básicamente de un lenguaje de tratamiento de textos (scripts), basado en VBScript (Visual basic Script). Se utiliza casi exclusivamente en los servidores Web de Microsoft (Internet Information Server y Personal Web Server). Los scripts ASP se ejecutan en el servidor y puede utilizarse conjuntamente con HTML y Javascript para realizar tareas interactivas y en tiempo real con el cliente.

Con ASP se pueden realizar fácilmente páginas de consulta de bases de datos, funciones sencillas como obtener la fecha y la hora actual del sistema servidor, cálculos matemáticos simples, etc.

Las páginas ASP comienzan a ejecutarse cuando un usuario solicita un archivo .asp al servidor Web a través del explorador. El servidor web llama a ASP, que lee el archivo solicitado, ejecuta las secuencias de comandos que encuentre y envía los resultados al explorador del cliente.

Puesto que las secuencias de comandos se ejecutan en el servidor y no en el cliente, es el servidor el que hace todo el trabajo necesario para generar las paginas que se envían al explorador. Las secuencias de comandos quedan ocultas a los usuarios, estos solo reciben el resultado de la ejecución en formato HTML.

Desaparece por tanto el problema de si el cliente puede o no ejecutar sentencias de comandos, el servidor Web solo envía el resultado en código HTML standard interpretable por cualquier explorador.

Los archivos .asp son archivos de texto normales, no es necesario ningún editor especial para crearlos, puede usarse cualquier editor que genere código ASCII. Un archivo .asp puede contener texto, código HTML, código ASP o cualquier combinación de estos. Si no contiene código ASP se comporta como un archivo .html normal.

3. SITUACION ACTUAL.

Actualmente el manejo de la información de los pacientes de FUNTER se realiza en forma manual: se registra en un archivo que contiene el expediente de los pacientes y su historial médico y de rehabilitación. FUNTER no cuenta con un sistema automatizado que permita controlar de mejor manera dicha información, por lo que se hace difícil el manejo de la misma para el personal que labora en la Institución, esta situación repercute en los pacientes de la Institución ya que, en ocasiones, la atención de estos puede ser lenta debido a los problemas ocasionados por el manejo inadecuado de la información.

Por otro lado se tiene la problemática que existen muchas personas con discapacidades físicas que viven en el interior del país a las cuales se les dificulta tener acceso a los servicios que presta FUNTER, por carecer de medios que les faciliten conocer acerca de estos y de la forma en como solicitarlos, ya que tienen que viajar hasta la capital para poder tener una información completa de ellos o para poder solicitar dichos servicios.

Actualmente FUNTER cuenta con la infraestructura necesaria para poder implementar un sistema que atienda la mayoría de estas necesidades y así poder mejorar los tiempos de respuesta para poder atender a un mayor número de personas con discapacidades físicas.

Entre el equipo con el cual cuenta FUNTER se tiene:

Una red de área local la cual posee 31 computadoras conectadas por medio de un servidor central el cual administra las transacciones en la red.

Las características de las computadoras que conforman la red de área local de FUNTER se describen a continuación:

Características de las Computadoras de FUNTER	
Servidor	
Hardware	Software
HP Brio 600	Windows 2000 Server
Procesador Pentium III a 600 MHz	Sql Server 7.0
3 Discos Duros de 18 GB c/u.	
256 MB RAM	
LOTE 1: 12 Computadoras	
Hardware	Software
HP Brio 600	Windows 2000
Procesador Pentium III a 500 MHz	Office 2000
Disco Duro de 8 GB	
64 MB RAM	
LOTE 2: 19 Computadoras	
Hardware	Software
HP Brio 600	Windows 98
Procesador AMD K6 a 450 MhZ	Office 2000
Discos duros de 5, 10 o 15 GB	
64 MB RAM	

Tabla 2: Características de las computadoras que actualmente posee FUNTER

CAPITULO III

DESCRIPCIÓN DE LA APLICACIÓN PROPUESTA

1. DESCRIPCIÓN GENERAL DE LA APLICACIÓN.

La solución propuesta a la problemática planteada anteriormente consiste en el desarrollo de una aplicación web para el manejo de información correspondiente a los procesos de rehabilitación de pacientes con discapacidades físicas, que son atendidos por FUNTER; que además permita a los usuarios que no son miembros de la Institución acceder a la información concerniente a los servicios que se desarrollan en la misma, así como la solicitud en línea de dichos servicios.

La solución propuesta abarca los siguientes elementos:

- ◆ Base de Datos eficiente y confiable para manipular adecuadamente la información correspondiente a los procesos de rehabilitación de los diversos pacientes de FUNTER.

- ◆ Módulo administrativo que permita la creación, modificación y eliminación de la información contenida en la base de datos.

- ◆ Enlaces (links) a páginas web estáticas con información general acerca de los diversos servicios que presta la Institución, así como también información acerca de cómo acceder a dichos servicios.

- ◆ Módulo de solicitud de servicios en línea, tanto para pacientes de FUNTER como para las personas que no son pacientes de la Institución.

- ◆ Módulo de consultas específicas relacionadas con el historial de rehabilitación de los pacientes de la Institución.

Procesos a ser considerados en la Aplicación:

Para describir los diversos procesos a ser considerados por la aplicación propuesta se hace uso de la metodología de Diagramas de Flujos de Datos (DFD) descrita en el capítulo II; en estos diagramas se observan los diferentes flujos de información correspondiente a los pacientes de FUNTER y a todas las entidades implicadas en la aplicación, así como también una serie de procesos propuestos que modificarán dicha información para poder así satisfacer las necesidades de los usuarios. Los diversos DFD se encuentran en el Anexo II de este documento.

Para poder recolectar la información necesaria para crear los Diagramas de Flujos de Datos con los respectivos procesos propuestos se establecieron varias reuniones con los diferentes miembros de FUNTER involucrados con la rehabilitación de pacientes, en estas reuniones se utilizó la técnica de Casos de Uso, descrita en el capítulo II, para poder recolectar dicha información y para poder establecer requerimientos. Los diversos Casos de Uso generados de la reuniones con el personal de FUNTER se encuentran en el Anexo III de este documento.

2. ARQUITECTURA DE LA APLICACIÓN.

Por ser una aplicación web que accesa una base de datos, se ha utilizado una arquitectura constituida por tres capas:

- ◆ Capa 1, capa de Usuario:
Esta capa consiste en la interfaz de presentación, es decir todo lo concerniente a la parte que visualizará el usuario de la aplicación y el medio a través del cual interactuará con los datos y elementos de la aplicación.
- ◆ Capa 2, capa de Reglas del Negocio:
Esta capa es la responsable de atender todos los requerimientos hechos por los usuarios a través de los navegadores, aquí se encuentra el código que el usuario invoca para recuperar los datos deseados.
- ◆ Capa 3, capa de Datos:
Esta capa se encuentran todos los datos que gestiona la aplicación.

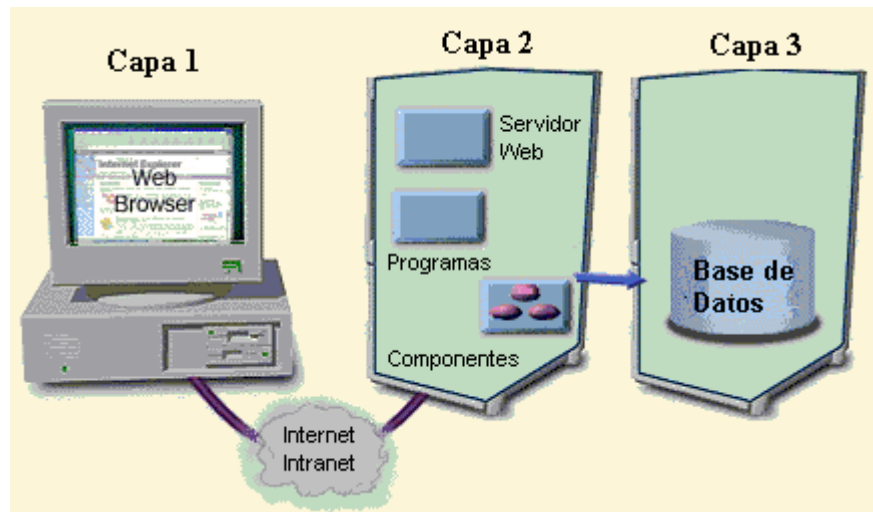


Figura 7. Esquema de Arquitectura de la Aplicación.

2.1. Tecnología utilizada en la Aplicación.

La tecnología y los diversos productos que se utilizaron para la construcción de la aplicación se detallan a continuación:

- ◆ **Capa 1, capa de Usuario:**
Cualquier tipo de navegador, de preferencia Microsoft Internet Explorer 5.0 ó superior, ó Netscape Navigator 6.0 ó superior.
- ◆ **Capa 2, capa de Reglas del Negocio:**
Servidor web Internet Information Server 5.0, Servidor de componentes COM+, corriendo sobre sistema operativo Windows 2000 Server.
- ◆ **Capa 3, capa de Datos:**
Manejador de Bases de Datos Microsoft SQL Server 2000.

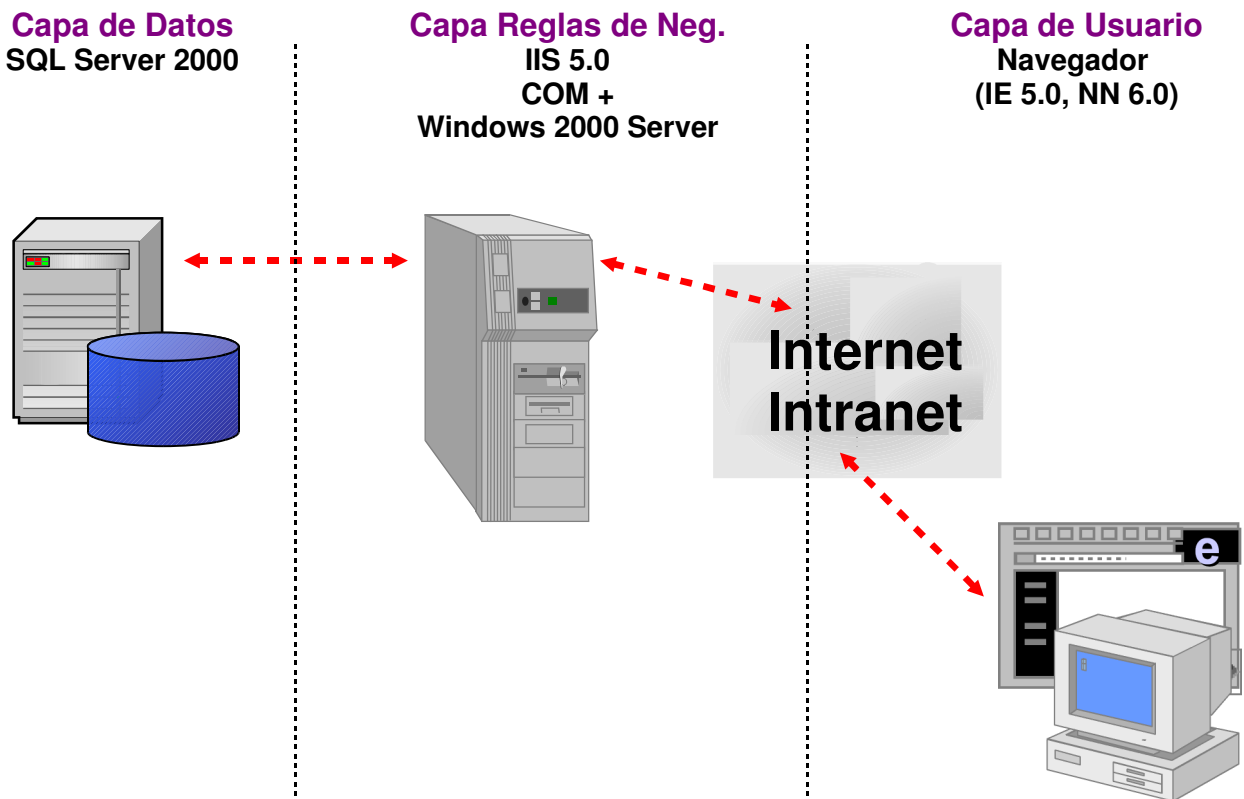


Figura 8. Tecnología utilizada en la Aplicación.

3. ESTÁNDARES DE LA APLICACIÓN Y DE LA BASE DE DATOS.

Para el diseño y desarrollo de la aplicación propuesta se han definido una serie de estándares, con la finalidad de normalizar el contenido de los diversos elementos que la conformarán, estos estándares han sido agrupados en estándares de la aplicación y estándares de la base de datos, los cuales se describen a continuación:

Estándares de la Aplicación.

A continuación se presentan los estándares utilizados en el desarrollo del la Aplicación Web del Sistema de Atención a Personas con Discapacidades:

- ◆ Los nombres de las páginas que conforman las opciones del sistema estarán constituidos por cuartetos de letras y/o números, separados por guiones bajos, que describan al objeto, hasta un máximo de cuatro cuartetos por nombre de objeto.
- ◆ Todos los nombres de las páginas llevarán un prefijo, el cual indicará el tipo de función que desempeñara la página, por ejemplo mantenimiento, reporte, catalogo, etc.
- ◆ Ejemplo de nombres de páginas:
 - mant_clin.asp, es el identificador de la ASP de mantenimiento de clínica.
 - mant_clin_resp.asp, es el identificador de una página que extiende la funcionalidad del mantenimiento de clínica.
- ◆ La aplicación se ha dividido en carpetas, según las opciones y funciones del sistema, en las que se encuentran las ASP, páginas HTM, scripts u otros

elementos cuya función corresponde a la descrita en el nombre de la carpeta, a continuación se detallan las carpetas existentes:

- *Carpeta Raíz:* contiene las paginas de funcionalidad general, por ejemplo:
 - *funter_inicio.asp*, es utilizada como página de inicio en la aplicación web.
 - *principal_ie.asp*, es utilizada como página de inicio del módulo SIAPD de la aplicación para el Internet Explorer.
 - *menu_ie.asp*, esta página posee la configuración del Menú de la aplicación para el Internet Explorer.
- *Header:* contiene las paginas que forman los encabezados de la aplicación.
- *Images:* contiene todas las imágenes utilizadas en la aplicación.
- *Menú:* contiene los scripts y gráficos utilizados para la definición del menú principal del Módulo SIAPD.
- *Páginas:* En esta carpeta se encuentran todas las páginas (htm y asp) y otros elementos que le dan la funcionalidad a las diferentes opciones de la aplicación web, estas páginas se han agrupado de acuerdo a dicha funcionalidad en las carpetas que se describen a continuación:
 - *Estilos:* contiene archivos cascade style sheets (css) que son utilizadas para dar la apariencia y formato (tipos de letras, colores, etc.) de las diferentes páginas de la Aplicación.
 - *Inicio:* contiene las diversas páginas principales o páginas de bienvenida de los diversos módulos de la aplicación.
 - *Login:* contiene las ASP que son utilizadas para que el usuario pueda iniciar una sesión dentro de la aplicación, es decir poder conectarse a la base de datos y tener acceso a las diferentes opciones de la Aplicación.
 - *Catálogos:* contiene las páginas (htm y asp) que conforman las opciones de los catálogos del Módulo SIAPD de la Aplicación.
 - *Mantenimientos:* contiene las páginas (htm y asp) que conforman las opciones de los mantenimientos del Módulo SIAPD de la Aplicación.

- *Procesos*: contiene las páginas (htm y asp) que conforman las opciones de los procesos del Módulo SIAPD de la Aplicación.
- *Reportes*: contiene las páginas (htm y asp) que conforman las opciones de los reportes o informes del Módulo SIAPD de la Aplicación.
- *Utilitarios*: contiene las páginas (htm y asp) que conforman las opciones de los utilitarios del Módulo SIAPD de la Aplicación.
- *Modulo_información*: contiene las páginas (htm y asp) que conforman las opciones del Módulo de Información de la Aplicación.
- *Modulo_pacientes*: contiene las páginas (htm y asp) que conforman las opciones del Módulo de Pacientes de la Aplicación.
- *Modulo_solicitudes*: contiene las páginas (htm y asp) que conforman las opciones del Módulo de Solicitudes de la Aplicación.

Estándares de la Base de Datos.

A continuación se presentan los estándares utilizados en el diseño y desarrollo de la Base de Datos del Sistema de Atención a Personas con Discapacidades:

- ◆ Los nombres de los objetos de la Base de Datos llevarán con el prefijo “SIAPD”, lo cual hará referencia a “Sistema de Atención a Pacientes con Discapacidades”; luego el nombre irá seguido por una letra que servirá para identificar el tipo de objeto de la base de datos, según lo siguiente:
 - U, para las tablas.
 - T, para los triggers.
 - P, para los procedimientos almacenados.
 - V, para las vistas.

Luego se colocará el descriptor de los objetos en base a los estándares que se detallan a continuación.

- ◆ Después del prefijo SIAPD y de la letra que identifica el tipo de objeto, se colocarán cuartetos de letras y/o números, separados por guiones bajos, que describan al objeto, hasta un máximo de cuatro cuartetos por nombre de objeto.
- ◆ Los nombres de los objetos de la Base de Datos y sus componentes deben de estar escritos con letras mayúsculas.
- ◆ En el caso de los nombres de los triggers de la Base de Datos, el primer cuarteto de estos servirá como indicador del tipo de trigger según se detalla a continuación:
 - ADIC, para triggers de adición (insert) de registros.
 - ELIM, para triggers de eliminación (delete) de registros.
 - MODI, para triggers de modificación (update) de registros.
- ◆ En los cuartetos de los nombres de los objetos de la Base de Datos se evitará poner preposiciones (a, ante, de,...), artículos (el, la, los, etc.), conjunciones (y, que, etc).
- ◆ Ejemplos nombres:
 - SIAPDU_TIPO_SERV, es el identificador de la Tabla de Tipos de Servicios.
 - SIAPDT_ADIC_DEPA, es el identificador del Trigger de Adición de la Tabla de Departamentos.
 - SIAPDP_GENE_EXPE_PACI, es el identificador del Procedimiento de Generación de Expedientes de Pacientes.

4. DISEÑO DE LA BASE DE DATOS.

4.1. Modelo Conceptual de Datos.

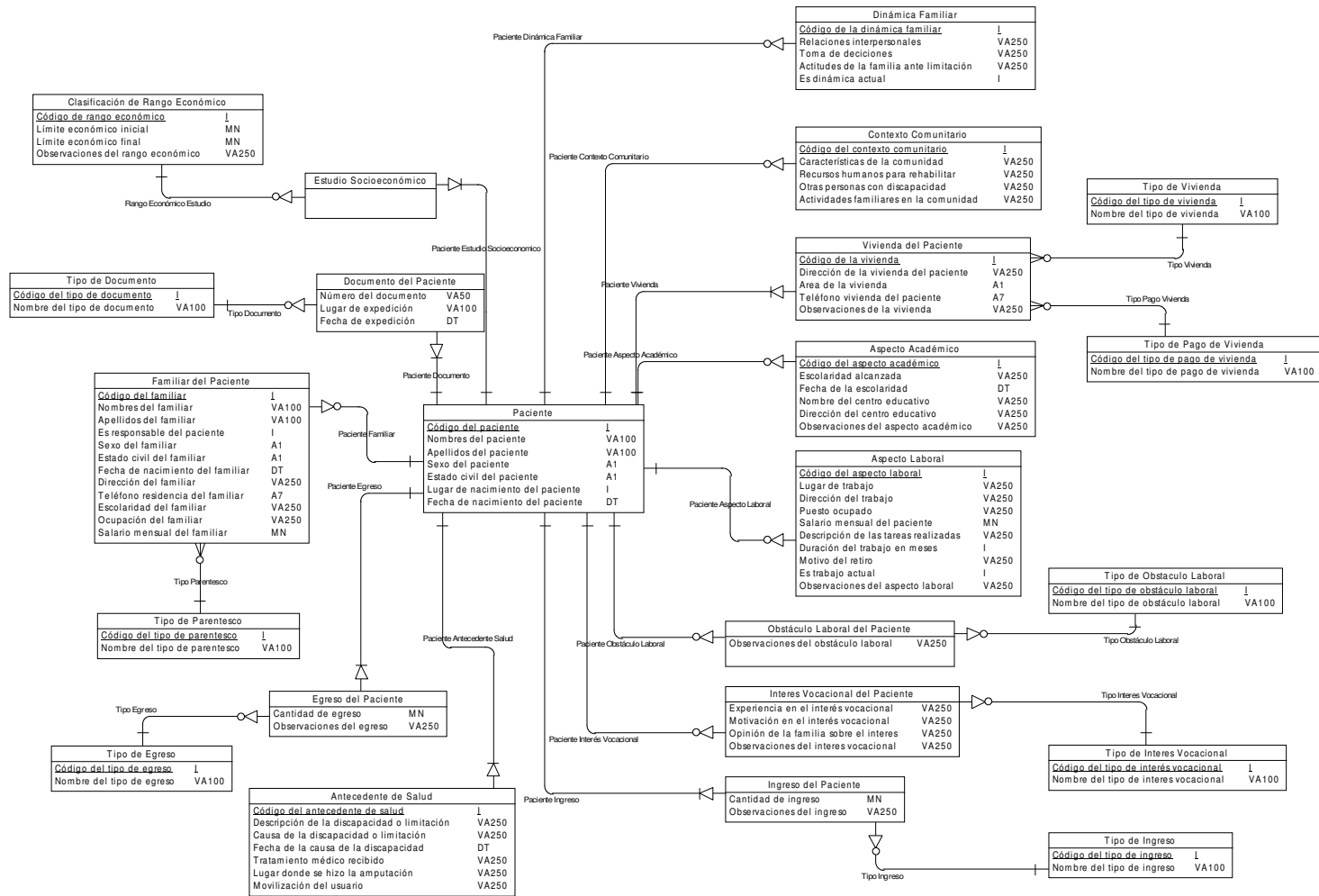
Un Modelo Conceptual de Datos representa la estructura lógica completa de una base de datos, la cual es independiente de cualquier software manejador de bases de datos. Un modelo conceptual frecuentemente contiene objetos de datos aún no implementados en una base de datos física.

Entre las principales funciones de un modelo conceptual de datos tenemos:

- ◆ Representar la organización de los datos en una forma gráfica.
- ◆ Verificar la validez del diseño de datos.
- ◆ Generar un modelo físico de datos con las especificaciones de la aplicación de una base de datos física.

A continuación se presenta el Modelo Conceptual de Datos del Sistema de Atención de Pacientes con Discapacidades Físicas de FUNTER (SIAPD), este se encuentra conformado por el Modelo General del Sistema (General del SIAPD) y por el Sub-modelo de Pacientes del Sistema (Pacientes del SIAPD).

4.1.2. Sub-Modelo Conceptual: Pacientes del SIAPD.



4.2. Modelo Físico de Datos.

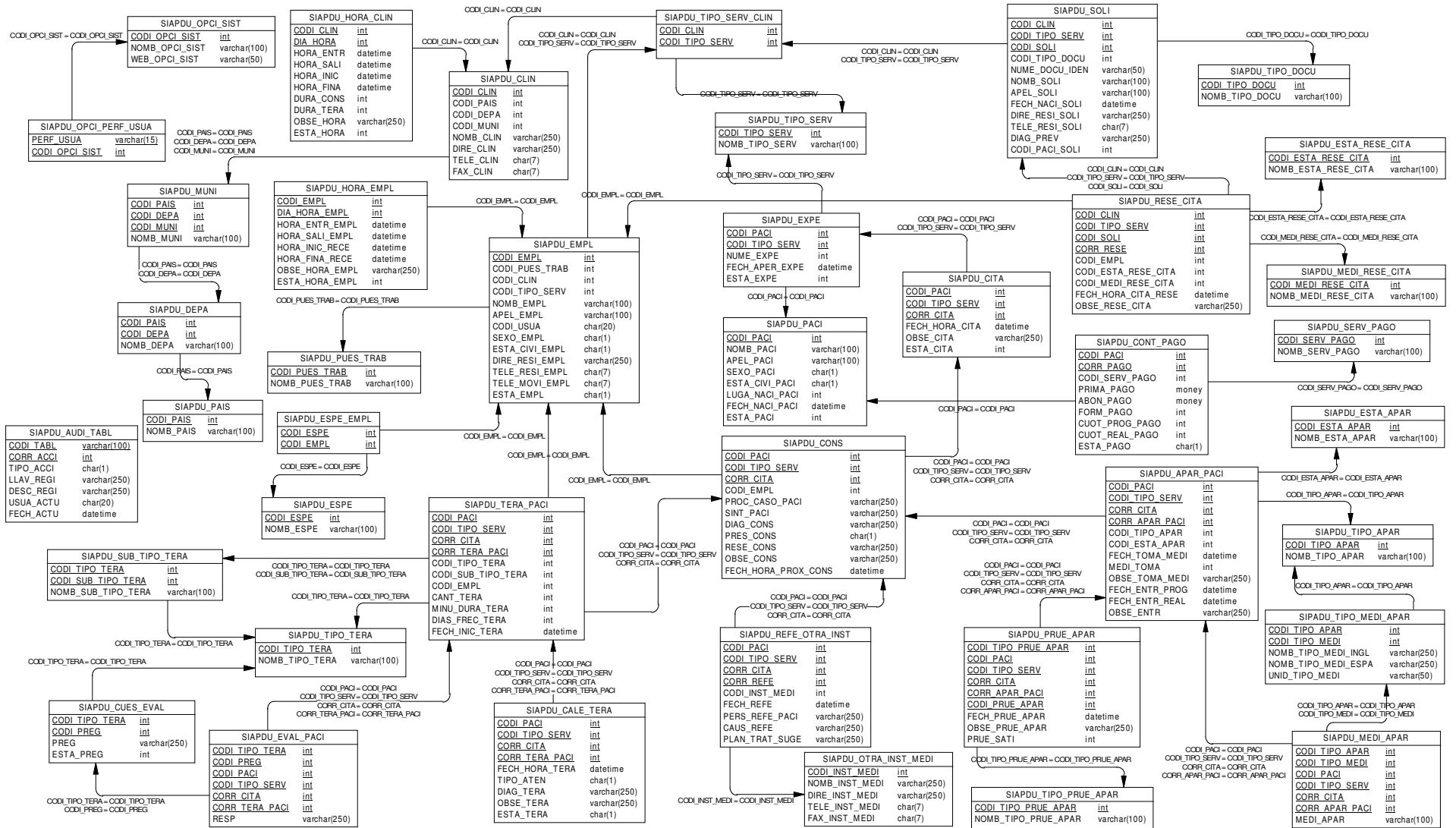
Un Modelo Físico de Datos especifica la implementación física de una base de datos. El modelo físico de datos, a diferencia del modelo conceptual, considera los detalles del software manejador de bases de datos.

Entre las principales funciones de un modelo físico de datos tenemos:

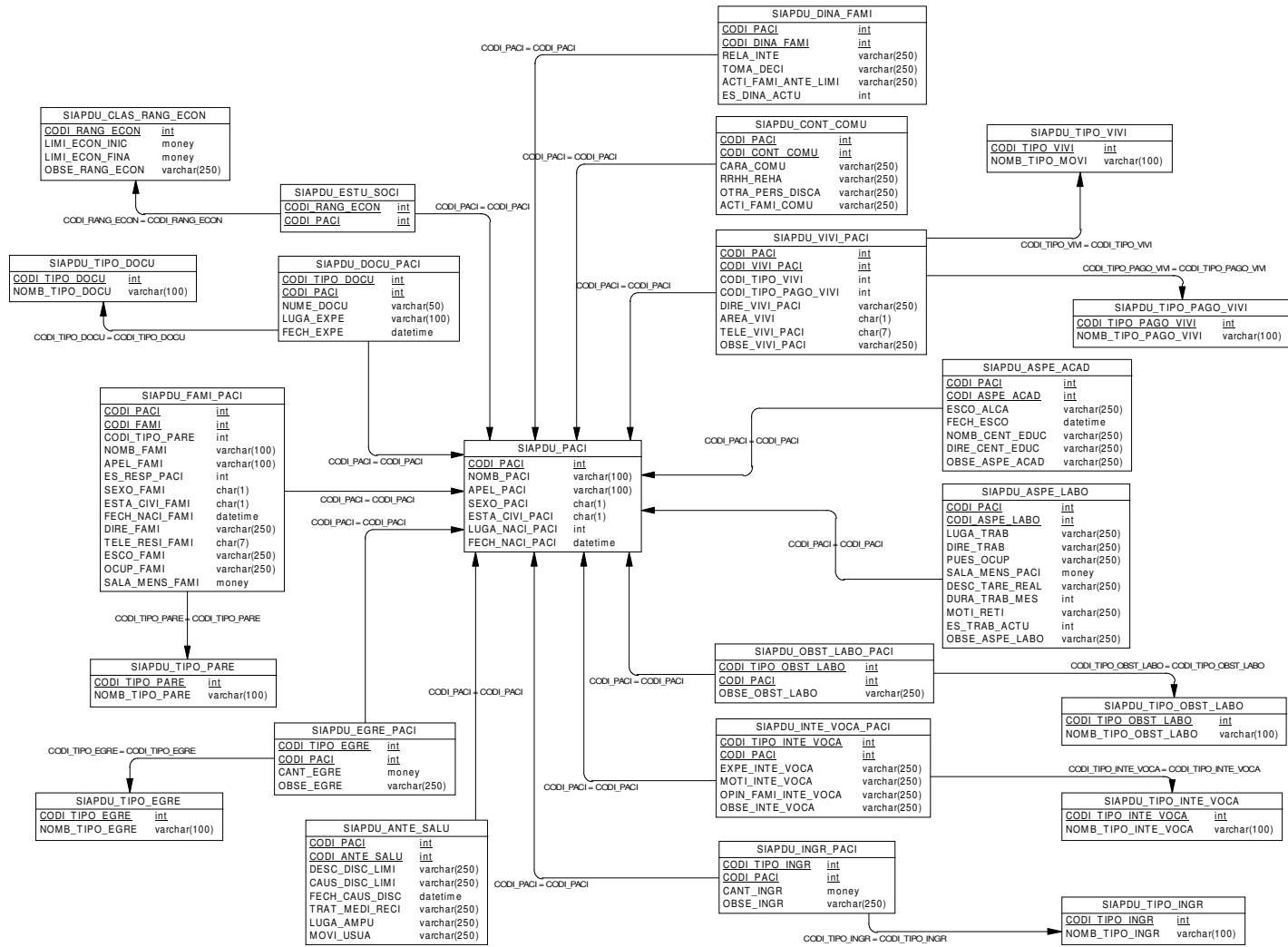
- ◆ Representar la organización física de los datos en una forma gráfica.
- ◆ Generar los scripts de creación y/o modificación de la base de datos.
- ◆ Definir la integridad referencial de los datos, a través de triggers o constraints .
- ◆ Generar atributos extendidos de la base de datos.
- ◆ Hacer ingenierías en reversa de una base de datos existente.
- ◆ Regenerar un modelo conceptual de datos.

A continuación se presenta el Modelo Físico de Datos del Sistema de Atención de Pacientes con Discapacidades Físicas de FUNTER (SIAPD), este se encuentra conformado por el Modelo General del Sistema (General del SIAPD) y por el Sub-modelo de Pacientes del Sistema (Pacientes del SIAPD).

4.2.1. Modelo Físico: General del SIAPD.



4.2.2. Sub-Modelo Físico: Pacientes del SIAPD.



4.3. Diccionario de Datos.

Los diccionarios de datos son referencias de datos acerca de ellos (esto es, metadatos) compilados por los analistas de sistemas para guiarse a través del análisis y diseño. El diccionario de datos recolecta, coordina y confirma lo que significa un término de datos específico para diferentes elementos de un sistema.

A continuación se presenta el diccionario de datos del Sistema de Atención a pacientes con Discapacidades Físicas de FUNTER (SIAPD):

4.3.1. Listado de Objetos de la Base de Datos.

4.3.1.1. Listado de Tablas:

A continuación se listan las tablas utilizadas en la base de datos del Sistema de Atención a pacientes con Discapacidades Físicas de FUNTER (SIAPD):

Nombre de la Tabla	Código de la Tabla
Antecedente de Salud	SIAPDU_ ANTE_ SALU
Aparato del Paciente	SIAPDU_ APAR_ PACI
Aspecto Académico	SIAPDU_ ASPE_ ACAD
Aspecto Laboral	SIAPDU_ ASPE_ LABO
Auditoría de Tablas	SIAPDU_ AUDI_ TABL
Cita	SIAPDU_ CITA
Clasificación de Rango Económico	SIAPDU_ CLAS_ RANG_ ECON
Clínica	SIAPDU_ CLIN
Consulta	SIAPDU_ CONS
Contexto Comunitario	SIAPDU_ CONT_ COMU
Control de Asistencia de Terapia	SIAPDU_ CALE_ TERA
Control de Pagos	SIAPDU_ CONT_ PAGO
Cuestionario para Evaluación	SIAPDU_ CUES_ EVAL
Departamento	SIAPDU_ DEPA
Dinámica Familiar	SIAPDU_ DINA_ FAMI
Documento del Paciente	SIAPDU_ DOCU_ PACI
Egreso del Paciente	SIAPDU_ EGRE_ PACI
Empleado	SIAPDU_ EMPL
Especialidad	SIAPDU_ ESPE

Nombre de la Tabla	Código de la Tabla
Especialidad de un Empleado	SIAPDU_ESPE_EMPL
Estatus de la Reservación de Cita	SIAPDU_ESTA_RESE_CITA
Estatus del Aparato	SIAPDU_ESTA_APAR
Estudio Socioeconómico	SIAPDU_ESTU_SOCI
Evaluación de Paciente	SIAPDU_EVAL_PACI
Expediente	SIAPDU_EXPE
Familiar del Paciente	SIAPDU_FAMI_PACI
Horarios de Clínicas	SIAPDU_HORA_CLIN
Horarios de los Empleados	SIAPDU_HORA_EMPL
Ingreso del Paciente	SIAPDU_INGR_PACI
Interés Vocacional del Paciente	SIAPDU_INTE_VOCA_PACI
Medidas del Aparato	SIAPDU_MEDI_APAR
Medio de Reservación de Cita	SIAPDU_MEDI_RESE_CITA
Municipio	SIAPDU_MUNI
Obstáculo Laboral del Paciente	SIAPDU_OBST_LABO_PACI
Opciones del Perfil de Usuario	SIAPDU_OPCI_PERF_USUA
Opciones del Sistema	SIAPDU_OPCI_SIST
Otra Institución Medica	SIAPDU_OTRA_INST_MEDI
Paciente	SIAPDU_PACI
País	SIAPDU_PAIS
Prueba del Aparato	SIAPDU_PRUE_APAR
Puesto de Trabajo	SIAPDU_PUES_TRAB
Referencia a Otra Institución	SIAPDU_REFE_OTRA_INST
Reservación de Cita	SIAPDU_RESE_CITA
Servicio a Pagar	SIAPDU_SERV_PAGO
Solicitante	SIAPDU_SOLI
Sub Tipo de Terapia	SIAPDU_SUB_TIPO_TERA
Terapia del Paciente	SIAPDU_TERA_PACI
Tipo de Aparato	SIAPDU_TIPO_APAR
Tipo de Documento	SIAPDU_TIPO_DOCU
Tipo de Egreso	SIAPDU_TIPO_EGRE
Tipo de Ingreso	SIAPDU_TIPO_INGR
Tipo de Interés Vocacional	SIAPDU_TIPO_INTE_VOCA
Tipo de Medida de Aparatos	SIPADU_TIPO_MEDI_APAR
Tipo de Obstáculo Laboral	SIAPDU_TIPO_OBST_LABO
Tipo de Pago de Vivienda	SIAPDU_TIPO_PAGO_VIVI
Tipo de Parentesco	SIAPDU_TIPO_PARE
Tipo de Prueba del Aparato	SIAPDU_TIPO_PRUE_APAR
Tipo de Servicio	SIAPDU_TIPO_SERV
Tipo de Servicio de una Clínica	SIAPDU_TIPO_SERV_CLIN
Tipo de Terapia	SIAPDU_TIPO_TERA
Tipo de Vivienda	SIAPDU_TIPO_VIVI
Vivienda del Paciente	SIAPDU_VIVI_PACI

4.3.1.2. Listado de Referencias:

Nombre de la Referencia	Código de la Referencia	Tabla Padre	Tabla Hija
Aparato Medida	APAR_MEDI_APAR	SIAPDU_APAR_PACI	SIAPDU_MEDI_APAR
Aparato Prueba	APAR_PRUE_APAR	SIAPDU_APAR_PACI	SIAPDU_PRUE_APAR
Cita Consulta	CITA_CONS	SIAPDU_CITA	SIAPDU_CONS
Clínica Tipo de Servicio	CLIN_TIPO_SERV	SIAPDU_CLIN	SIAPDU_TIPO_SERV_CLIN
Consulta Aparato	CONSI_APAR	SIAPDU_CONS	SIAPDU_APAR_PACI
Consulta Referencia	CONS_MEDI_REFE	SIAPDU_CONS	SIAPDU_REFE_OTRA_INST
Consulta Terapia	CONS_MEDI_TERA	SIAPDU_CONS	SIAPDU_TERA_PACI
Cuestionario Evaluación	CUEST_EVAL	SIAPDU_CUES_EVAL	SIAPDU_EVAL_PACI
Departamento Municipio	DEPA_MUNI	SIAPDU_DEPA	SIAPDU_MUNI
Empleado Consulta	EMPL_CONS	SIAPDU_EMPL	SIAPDU_CONS
Empleado Especialidad	EMPL_ESPE_EMPL	SIAPDU_EMPL	SIAPDU_ESPE_EMPL
Empleado Reservación de Cita	EMPL_RELA_CITA	SIAPDU_EMPL	SIAPDU_RESE_CITA
Empleado Terapia	EMPL_TERA	SIAPDU_EMPL	SIAPDU_TERA_PACI
Especialidad Esp de Empl	ESPE_ESPE_EMPL	SIAPDU_ESPE	SIAPDU_ESPE_EMPL
Estatus Aparato	ESTA_APAR	SIAPDU_ESTA_APAR	SIAPDU_APAR_PACI
Estatus de Reservación de Citas	ESTA_RESE_CITA	SIAPDU_ESTA_RESE_CITA	SIAPDU_RESE_CITA
Expediente Cita	EXPE_CITA	SIAPDU_EXPE	SIAPDU_CITA
Horario de Clínica	HORA_CLIN	SIAPDU_CLIN	SIAPDU_HORA

Nombre de la Referencia	Código de la Referencia	Tabla Padre	Tabla Hija
			CLIN
Horarios de Empleado	HORA_EMPL	SIAPDU_EMPL	SIAPDU_HORA_EMPL
Inst Medica Referencia	OTRA_INST_MEDI_REFE	SIAPDU_OTRA_INST_MEDI	SIAPDU_REFE_OTRA_INST
Medio de Reservación de Cita	MEDI_RESE_CITA	SIAPDU_MEDI_RESE_CITA	SIAPDU_RESE_CITA
Municipio Clínica	MUNI_CLIN	SIAPDU_MUNI	SIAPDU_CLIN
Opciones Perfil Usuario	OPCI_PERF_USUA	SIAPDU_OPCI_SIST	SIAPDU_OPCI_PERF_USUA
Paciente Antecedente Salud	PACI_ANTE_SALU	SIAPDU_PACI	SIAPDU_ANTE_SALU
Paciente Aspecto Académico	PACI_ACAD	SIAPDU_PACI	SIAPDU_ASPE_ACAD
Paciente Aspecto Laboral	PACI_LABO	SIAPDU_PACI	SIAPDU_ASPE_LABO
Paciente Contexto Comunitario	PACI_CONT_COMU	SIAPDU_PACI	SIAPDU_CONT_COMU
Paciente Dinámica Familiar	PACI_DINA_FAMI	SIAPDU_PACI	SIAPDU_DINA_FAMI
Paciente Documento	PACI_DOCU	SIAPDU_PACI	SIAPDU_DOCU_PACI
Paciente Egreso	PACI_EGRE	SIAPDU_PACI	SIAPDU_EGRE_PACI
Paciente Estudio Socioeconomico	PACI_ESTU_SOCI	SIAPDU_PACI	SIAPDU_ESTU_SOCI
Paciente Expediente	PACI_EXPE	SIAPDU_PACI	SIAPDU_EXPE
Paciente Familiar	PACI_FAMI	SIAPDU_PACI	SIAPDU_FAMI_PACI
Paciente Ingreso	PACI_INGR	SIAPDU_PACI	SIAPDU_INGR_PACI
Paciente Interés Vocacional	PACI_INTE_VOCA	SIAPDU_PACI	SIAPDU_INTE_VOCA_PACI
Paciente Obstáculo Laboral	PACI_OBST_LABO	SIAPDU_PACI	SIAPDU_OBST_LABO_PACI

Nombre de la Referencia	Código de la Referencia	Tabla Padre	Tabla Hija
Paciente Pago	PACI_PAGO	SIAPDU_PACI	SIAPDU_CONT_PAGO
Paciente Vivienda	PACI_VIVI	SIAPDU_PACI	SIAPDU_VIVI_PACI
Pago Servicio	PAGO_SERV	SIAPDU_SERV_PAGO	SIAPDU_CONT_PAGO
País Departamento	PAIS_DEPA	SIAPDU_PAIS	SIAPDU_DEPA
Puesto de Trabajo Empleado	PUES_TRAB_EMPL	SIAPDU_PUES_TRAB	SIAPDU_EMPL
Rango Económico Estudio	RANG_ECON_ESTU	SIAPDU_CLAS_RANG_ECON	SIAPDU_ESTU_SOCI
Solicitante Reservación de Cita	SOLI_RESE_CITA	SIAPDU_SOLI	SIAPDU_RESE_CITA
Sub Tipo Terapia	SUB_TIPO_TERA	SIAPDU_SUB_TIPO_TERA	SIAPDU_TERA_PACI
Terapia Control	TERA_CONT	SIAPDU_TERA_PACI	SIAPDU_CALE_TERA
Terapia Evaluación	TERA_EVAL	SIAPDU_TERA_PACI	SIAPDU_EVAL_PACI
Tipo Aparato	TIPO_APAR	SIAPDU_TIPO_APAR	SIAPDU_APAR_PACI
Tipo Aparato Medida	TIPO_APAR_TIPO_MEDI	SIAPDU_TIPO_APAR	SIPADU_TIPO_MEDI_APAR
Tipo de Medida	TIPO_MEDI_MEDI_APAR	SIPADU_TIPO_MEDI_APAR	SIAPDU_MEDI_APAR
Tipo de Servicio Clínica	TIPO_SERV_CLIN	SIAPDU_TIPO_SERV	SIAPDU_TIPO_SERV_CLIN
Tipo de Servicio Expediente	TIPO_SER_EXPE	SIAPDU_TIPO_SERV	SIAPDU_EXPE
Tipo de Servicio Solicitado	TIPO_SERV_SOLI	SIAPDU_TIPO_SERV_CLIN	SIAPDU_SOLI
Tipo Documento	TIPO_DOCU	SIAPDU_TIPO_DOCU	SIAPDU_DOCU_PACI
Tipo Documento Solicitate	TIPO_DOCU_SOLI	SIAPDU_TIPO_DOCU	SIAPDU_SOLI

Nombre de la Referencia	Código de la Referencia	Tabla Padre	Tabla Hija
Tipo Egreso	TIPO_EGRE	SIAPDU_TIPO_EGRE	SIAPDU_EGRE_PACI
Tipo Ingreso	TIPO_INGR	SIAPDU_TIPO_INGR	SIAPDU_INGR_PACI
Tipo Interes Vocacional	TIPO_INTE_VOCA	SIAPDU_TIPO_INTE_VOCA	SIAPDU_INTE_VOCA_PACI
Tipo Obstáculo Laboral	TIPO_OBST_LABO	SIAPDU_TIPO_OBST_LABO	SIAPDU_OBST_LABO_PACI
Tipo Pago Vivienda	TIPO_PAGO_VIVI	SIAPDU_TIPO_PAGO_VIVI	SIAPDU_VIVI_PACI
Tipo Sub Tipo Terapia	TIPO_TERA_SUB_TIPO	SIAPDU_TIPO_TERA	SIAPDU_SUB_TIPO_TERA
Tipo Tabla Padresco	TIPO_PARE	SIAPDU_TIPO_PARE	SIAPDU_FAMI_PACI
Tipo Prueba Aparato	TIPO_PRUE_APAR	SIAPDU_TIPO_PRUE_APAR	SIAPDU_PRUE_APAR
Tipo Servicio Clínica Empleado	TIPO_SERV_CLIN_EMPL	SIAPDU_TIPO_SERV_CLIN	SIAPDU_EMPL
Tipo Terapia	TIPO_TERA	SIAPDU_TIPO_TERA	SIAPDU_TERA_PACI
Tipo Terapia Cuestionario	TIPO_TERA_CUES	SIAPDU_TIPO_TERA	SIAPDU_CUES_EVAL
Tipo Vivienda	TIPO_VIVI	SIAPDU_TIPO_VIVI	SIAPDU_VIVI_PACI

4.3.1.1. Listado de Triggers.

En la siguiente tabla se listan los triggers⁴ utilizados en la base de datos, las tablas a las que corresponden y bajo que acción en la tabla se activan:

Código del Trigger	Tabla	Acción
SIAPDT_ADIC_CITA	SIAPDU_CITA	Inserción

⁴ El código de los triggers se detalla en el Anexo V del presente manual.

Código del Trigger	Tabla	Acción
SIAPDT_ADIC_DEPA	SIAPDU_DEPA	Inserción
SIAPDT_ADIC_ESPE	SIAPDU_ESPE	Inserción
SIAPDT_ADIC_ESTA_RESE_CITA	SIAPDU_ESTA_RESE_CITA	Inserción
SIAPDT_ADIC_ESTA_APAR	SIAPDU_ESTA_APAR	Inserción
SIAPDT_ADIC_EXPE	SIAPDU_EXPE	Inserción
SIAPDT_ADIC_MEDI_RESE_CITA	SIAPDU_MEDI_RESE_CITA	Inserción
SIAPDT_ADIC_MUNI	SIAPDU_MUNI	Inserción
SIAPDT_ADIC_PACI	SIAPDU_PACI	Inserción
SIAPDT_ADIC_PAIS	SIAPDU_PAIS	Inserción
SIAPDT_ADIC_PUEST_TRAB	SIAPDU_PUEST_TRAB	Inserción
SIAPDT_ADIC_RESE_CITA	SIAPDU_RESE_CITA	Inserción
SIAPDT_ADIC_SERV_PAGO	SIAPDU_SERV_PAGO	Inserción
SIAPDT_ADIC_SOLI	SIAPDU_SOLI	Inserción
SIAPDT_ADIC_SUB_TIPO_TERA	SIAPDU_SUB_TIPO_TERA	Inserción
SIAPDT_ADIC_TIPO_APAR	SIAPDU_TIPO_APAR	Inserción
SIAPDT_ADIC_TIPO_DOCU	SIAPDU_TIPO_DOCU	Inserción
SIAPDT_ADIC_TIPO_EGRE	SIAPDU_TIPO_EGRE	Inserción
SIAPDT_ADIC_TIPO_INGR	SIAPDU_TIPO_INGR	Inserción
SIAPDT_ADIC_TIPO_INTE_VOCA	SIAPDU_TIPO_INTE_VOCA	Inserción
SIPADU_TIPO_MEDI_APAR	SIPADU_TIPO_MEDI_APAR	Inserción
SIAPDT_ADIC_TIPO_OBST_LABO	SIAPDU_TIPO_OBST_LABO	Inserción
SIAPDT_ADIC_TIPO_PAGO_VIVI	SIAPDU_TIPO_PAGO_VIVI	Inserción
SIAPDT_ADIC_TIPO_PARE	SIAPDU_TIPO_PARE	Inserción
SIAPDT_ADIC_TIPO_PRUE_APAR	SIAPDU_TIPO_PRUE_APAR	Inserción
SIAPDT_ADIC_TIPO_SERV	SIAPDU_TIPO_SERV	Inserción
SIAPDT_ADIC_TIPO_TERA	SIAPDU_TIPO_TERA	Inserción
SIAPDT_ADIC_TIPO_VIVI	SIAPDU_TIPO_VIVI	Inserción

4.3.1.4. Listado de Procedimientos Almacenados.

En la siguiente tabla se listan los procedimientos almacenados⁵ utilizados en la base de datos del Sistema de Atención a pacientes con Discapacidades Físicas de FUNTER (SIAPD):

Nombre de la Procedimiento	Código del Procedimiento
Creación de Expediente y Confirmación de Cita	SIAPDP_CREA_EXPE_CONF_CITA
Búsqueda de Disponibilidad de Horarios	SIAPDP_DISP_HORA_CITA

⁵ El código de los procedimientos almacenados se detalla en el Anexo V del presente manual.

Nombre de la Procedimiento	Código del Procedimiento
para Citas	

4.3.2. Detalle de Tablas de la Base de Datos.

Para describir la información de cada una de las tablas que conforman la Aplicación se seguirá el siguiente esquema:

- ◆ Al Principio se colocará un recuadro con la información general de la tabla: nombre de la tabla, código y descripción.
- ◆ Luego se colocará una lista con el detalle de las columnas de la tabla: nombre de la columna, código de la columna, tipo de dato, si es llave primaria (P), si es columna obligatoria (O).
- ◆ Posteriormente, se colocará un recuadro con las verificaciones (check) de las columnas de la tabla: nombre de la columna, lista de valores permitidos, valor por default, valor máximo y valor mínimo.
- ◆ Después, se colocará una lista con el detalle de los índices de la tabla: código del índice, si es llave primaria (P), si es llave foránea (F), si es llave única (U), si es índice cluster (C), columnas que involucra el índice, tipo de ordenamiento (Ascendente ó Descendente).
- ◆ Finalmente, se colocarán dos listas con las referencias relacionadas a la tabla, tanto las referencias que esta hace a otra tabla, así como las tablas que hacen referencia a esta.

En el Anexo IV se detalla cada una de las tablas de la base de datos del Sistema de Atención a Pacientes con Discapacidades Físicas (SIAPD), según el esquema descrito anteriormente.

5. DESCRIPCIÓN DE LOS MÓDULOS DE LA APLICACIÓN.

La aplicación web propuesta para el manejo de información correspondiente a los procesos de rehabilitación de pacientes con discapacidades físicas que son atendidos por FUNTER, por sus objetivos, consta de dos tipos de módulos: Públicos y Privados.

Los módulos públicos son aquellos que pueden ser utilizados por cualquier persona que ingresa a la aplicación a través de Internet o de la Intranet de FUNTER, estos módulos son: Módulo de Información y Módulo de Solicitud de Servicios.

Los módulos privados, como su nombre lo dice, son aquellos módulos de acceso restringido, es decir que pueden ser utilizados solamente por personas autorizadas que ingresan a la aplicación a través de Internet o de la Intranet de FUNTER, estos módulos son: Módulo del Sistema de Atención a Pacientes con Discapacidades Físicas (Módulo SIAPD) y Módulo de Pacientes.

5.1. MÓDULOS PÚBLICOS DE LA APLICACIÓN.

5.1.1. Módulo de Información.

Este módulo contiene solamente páginas estáticas, es decir páginas cuyo contenido solo podrá cambiar a través de un proceso de mantenimiento y edición, haciendo necesario realizar, posteriormente a dicho mantenimiento, una regeneración del módulo de información (deployment), para poder hacer disponibles los cambios a los usuarios de este módulo.

El contenido de las páginas de este módulo será definido por FUNTER, con las respectivas recomendaciones y observaciones del equipo que desarrollará el proyecto.

Contenido del Módulo de Información:

El módulo de información está conformado por las páginas de contenido estático que se listan a continuación:

- ◆ Página Principal del Módulo.
- ◆ Organización Institucional de FUNTER.
- ◆ Historia de FUNTER.
- ◆ Objetivos de FUNTER.
- ◆ Población objetivo de FUNTER.
- ◆ Experiencia Institucional de FUNTER.
- ◆ Metodología de FUNTER.
- ◆ Tipos de Rehabilitación.
- ◆ Fábrica de Prótesis y Ortesis.
- ◆ Como Acceder a los Servicios.
- ◆ Contáctenos.

5.1.2. Módulo de Solicitud de Servicios.

Este módulo esta habilitado para pacientes de FUNTER como para otras personas que también deseen solicitar alguno de los servicios que la Institución presta.

Según los procesos de FUNTER, para poder brindar a un solicitante cualquier tipo de servicio, el primer requisito es que esta persona reciba una consulta médica, según el tipo de servicio que desea recibir; por lo tanto este módulo esta compuesto principalmente por un formulario mediante el cual la persona interesada puede reservar una cita para una consulta médica, previo a la verificación por parte del interesado de las disponibilidades de horarios de citas de una Clínica específica de la Institución.

5.2. MÓDULOS PRIVADOS DE LA APLICACIÓN.

5.2.1. Módulo SIAPD.

El Módulo SIAPD (Sistema de Atención a Pacientes con Discapacidades Físicas) funciona bajo el concepto de una Intranet, por tal motivo este módulo solamente está habilitado para el personal autorizado de la FUNTER.

A través de este módulo se permitirá el mantenimiento (creación, eliminación y modificación) de la información relacionada con los procesos de rehabilitación de los pacientes con discapacidades físicas que son atendidos en la Institución, y que estará almacenada en la base de datos de la Institución.

Contenido del Módulo SIAPD:

El Módulo SIAPD se encuentra conformado por una serie de opciones que permiten la manipulación de la información; según la funcionalidad de dichas opciones, estas se encuentran organizadas en 5 grupos, como se detalla a continuación:

◆ *Catálogos:*

Las opciones contenidas en este grupo se utilizarán para manipular la información considerada como requisitos mínimos para el buen funcionamiento del SIAPD; las opciones que conforman este grupo son:

- Estructura geopolítica de un país.
- Puestos de trabajo de FUNTER.
- Tipos de servicios.
- Especialidades.
- Tipos de terapias.
- Tipos de aparatos.
- Tipos de pruebas de aparatos.
- Estatus de aparatos.
- Servicios a pagar.
- Medios de reservación de citas.

- Estatus de reservación de citas.
 - Tipos de documentos.
 - Tipos de parentescos.
 - Tipos de ingresos.
 - Tipos de egresos.
 - Tipos de obstáculos laborales.
 - Tipos de intereses vocacionales.
 - Tipos de vivienda.
 - Tipos de pago de viviendas.
- ◆ *Mantenimientos:*
- Las opciones contenidas en este grupo se utilizarán para adicionar, modificar ó eliminar datos concernientes a la Institución, sus pacientes y los procesos de rehabilitación estos; las opciones que conforman este grupo son:
- Clínicas.
 - Tipos de servicios de clínicas.
 - Empleados.
 - Subtipos de terapias.
 - Cuestionarios de evaluación.
 - Otras instituciones médicas.
 - Pacientes.
 - Consultas médicas.
 - Terapias de un paciente.
 - Aparatos de un paciente.
 - Referencias a otras instituciones médicas.
- ◆ *Procesos:*
- Desde estas opciones, los usuarios podrán ejecutar una serie de procesos que modificarán la información concerniente a los pacientes de la Institución; las opciones que conforman este grupo son:
- Reservación de citas.

- Confirmación de citas.
 - Toma de medidas de paciente.
 - Prueba de aparatos.
 - Entrega de aparatos.
- ◆ *Reportes:*
- En este grupo de opciones se presentarán todos los informes y consultas del SIAPD; según las necesidades del usuario estos informes y consultas pueden ser visualizados en pantalla así como también pueden ser impresos; algunos de estos informes necesitarán que el usuario ingrese ciertos parámetros para poder presentar la información requerida. Las opciones que conforman este grupo son:
- Evaluación social del paciente.
 - Historial de consultas médicas de un paciente.
 - Historial de terapias de un paciente.
 - Historial de aparatos recibidos un paciente.
 - Historial de referencias a otras instituciones.
- ◆ *Utilitarios:*
- Las opciones contenidas en este grupo sirven como auxiliares de los usuarios del sistema en la utilización del mismo; las opciones que conforman este grupo son:
- Cambio de clínica.
 - Cambio de Clave de acceso (Password).
 - Ayuda del Sistema.

5.2.2. Módulo Pacientes

El Módulo Pacientes está habilitado para pacientes de FUNTER como para el personal autorizado de la Institución.

Este módulo está conformado por una serie de páginas dinámicas, es decir páginas que se actualizan con la información proveniente de la base de datos del Sistema de Atención a Pacientes con Discapacidades Físicas (SIAPD); éstas páginas contendrán información concerniente a los procesos de rehabilitación pacientes específicos de FUNTER.

Contenido del Módulo Pacientes:

El Módulo Pacientes está conformado por las páginas de contenido dinámico que se listan a continuación:

- ◆ Evaluación social del paciente.
- ◆ Historial de consultas médicas de un paciente.
- ◆ Historial de terapias de un paciente.
- ◆ Historial de aparatos recibidos un paciente.
- ◆ Historial de referencias a otras instituciones.

CAPITULO IV

SEGURIDAD DE LA APLICACIÓN

1. CONCEPTOS GENERALES.

El Sistema de Atención a Pacientes con Discapacidades Físicas (SIAPD) es construido con tecnologías de Internet y será accesible a través de esta red pública, por lo que estará expuesto a una variedad de riesgos, entre los cuales podemos mencionar:

- ◆ *Data – Tampering*, consiste en la captura y modificación de la información que circula entre la computadora cliente y el servidor web.

- ◆ *Enmascaramiento*, este riesgo consiste en que un usuario pretende ser alguien con más derechos sobre la información de los que realmente se le han establecido.

Con el objetivo de mantener segura la información en contra de estos riesgos, es común encontrar las siguientes técnicas:

- ◆ *Autenticación*, la cual consiste en verificar la identidad del usuario, y de acuerdo a esta otorgar acceso sobre los módulos y componentes de la aplicación.

- ◆ *Encriptación*, esta técnica codifica la información en su recorrido a través del Internet de tal forma que solo el usuario indicado pueda interpretarla adecuadamente.

2. ESQUEMA DE CONTROL DE ACCESOS.

La seguridad del Sistema de Atención a Pacientes con Discapacidades Físicas SIAPD, se basa en el método de autenticación, es decir en verificar la identidad del usuario y de acuerdo a esta otorgar acceso sobre los módulos y datos de la aplicación.

El sistema SIAPD, como ha sido concebido, permitirá el ingreso de usuarios que tendrán la facultad únicamente de leer o consultar la información publicada, estos usuarios denominados “Usuarios Públicos” no tendrán derechos para poder adicionar, modificar o eliminar datos.

Por otra parte, el sistema SIAPD dará acceso a una cantidad específica y bien definida de usuarios, que luego de pasar por un proceso de autenticación, se les permitirá dar mantenimiento a los datos, estos usuarios denominados “Operadores del Sistema” poseerán una clasificación dentro del sistema (doctores, secretarias, pacientes, etc.) y de acuerdo a esta clasificación tendrán derechos a acceder a las opciones del sistema y también a modificar, eliminar o adicionar datos.

3. SEGURIDAD A NIVEL DE LA APLICACIÓN.

El sistema SIAPD, consta de dos tipos de módulos: Públicos y Privados, de acuerdo a estos módulos están definidas las opciones a ser utilizadas por los diversos usuarios y por lo tanto las funciones que estos pueden realizar dentro de dichas opciones.

Los módulos públicos son aquellos que pueden ser utilizados por cualquier persona que ingresa a la aplicación a través de Internet o de la Intranet de FUNTER, estos módulos son: Módulo de Información y Módulo de Solicitud de Servicios.

Los módulos privados, como su nombre lo dice, son aquellos módulos de acceso restringido, es decir que pueden ser utilizados solamente por personas autorizadas que ingresan a la aplicación a través de Internet o de la Intranet de FUNTER, estos módulos son: Módulo del Sistema de Atención a Pacientes con Discapacidades Físicas (Módulo SIAPD) y Módulo de Pacientes.

4. SEGURIDAD A NIVEL DE LA BASE DE DATOS.

Para que un usuario pueda acceder a los datos contenidos en la base de datos del sistema SIAPD, es necesario que primero pase por un proceso de autenticación de la base de datos, el cual consiste en que este se conecte e inicie una sesión en la base de datos proporcionándole un nombre de inicio de sesión y una contraseña válida (estos datos se encuentran previamente establecidos en la base de datos), si estas credenciales son válidas entonces el usuario podrá conectarse a la base datos, mientras que si las credenciales no son válidas entonces se le negará el acceso al usuario.

Después de iniciar una sesión con éxito dentro de la base de datos, el usuario debe tener definidos permisos explícitos en ella para poder acceder y manipular sus diferentes elementos (tablas, procedimientos almacenados, vistas, etc.), y además debe poseer derechos de lectura, modificación eliminación y adición de datos.

En la base de datos del sistema SIAPD se encuentran definidos una serie de roles en los que se especifican los derechos para poder manipular los objetos y los datos de la base; todos los usuarios de la base poseen un rol determinado, el cual determina la forma en que este manipulará los datos. Por ejemplo se encuentra el rol de pacientes y los usuarios que poseen este rol tienen derechos de consulta sobre algunas tablas del sistema como las relacionadas con el estudio socio-económico del mismo; mientras que el por ejemplo los usuarios que tienen el rol de doctores tienen derechos a consultar, modificar y adicionar otras tablas del sistema como las relacionadas con las consultas y terapias de un paciente.

CONCLUSIONES

- ◆ Debido a la creciente demanda de los diversos servicios de rehabilitación que presta FUNTER a la población salvadoreña con discapacidades físicas, es necesario un sistema confiable y eficiente que permita manipular con facilidad la información concerniente a dichos servicios.
- ◆ El mejoramiento de las telecomunicaciones y la mayor accesibilidad de la población a estas debido al descenso de sus precios, han permitido que la tecnología utilizada en este rubro llegue a varias partes del territorio nacional, haciendo posible que un mayor número de personas puedan acceder a la Internet, convirtiéndola en un medio factible para el flujo de información, razón por la cual se utilizará esta tecnología para el desarrollo de esta proyecto.
- ◆ Una aplicación web que pueda ser accedida desde cualquier parte del país puede expandir la cobertura de los servicios que presta FUNTER a las personas con discapacidades físicas.
- ◆ Una de las limitantes principales en el desarrollo de cualquier aplicación es el hardware y el software a ser instalado en las computadoras de los usuarios finales, sin embargo una aplicación web no necesita grandes recursos en cuanto a hardware y software, lo cual la convierte en una ventaja en cuanto a economía y facilidad de uso.

- ◆ La aplicación web propuesta ayudará a mejorar la atención de los pacientes de FUNTER ya que se agilizarán los procesos que actualmente se realizan, permitiendo así que más personas sean atendidas diariamente en la Institución. También dicha aplicación será una fuente valiosa de información que permitirá a los usuarios tener acceso a datos actualizados, mediante los diferentes reportes y consultas puestos a su disposición; además el uso de la aplicación permitirá al personal de FUNTER responsable de la toma de decisiones realizar con mayor rapidez y eficacia dichas acciones.

FUENTES DE INFORMACIÓN

- ♦ ANÁLISIS Y DISEÑO DE SISTEMAS. Kenneth Kendall y Julie Kendall. Tercera Edición; Prentice Hall Hispanoamericana; México, 1997.
- ♦ THE RATIONAL UNIFIED PROCESS AN INTRODUCCTION. Philippe Kruchten y Longman. Segunda Edición; Addison Wesley Longman Inc.; Estados Unidos, 1999.
- ♦ THE UNIFIED SOFTWARE DEVELOPMENT PROCESS. Ivar Jacobson, Grady Booch y James Rumbaugh. Primera Edición; Addison Wesley Longman Inc.; Estados Unidos, 1999.
- ♦ APRENDIENDO MICROSOFT SQL SERVER 7.0 EN 21 DÍAS. Richard Waymire, Primera Edición, Prentice Hall Hispanoamericana, México, 2000.
- ♦ GUÍA PARA ELABORAR LA TESIS. Santiago Torres Xamar; Segunda Edición; McGraw-Hill Interamericana; México, 1992.
- ♦ MASTERING VISUAL BASIC 6 FUNDAMENTALS. Microsoft; Estados Unidos, 1999.
- ♦ MASTERING VISUAL BASIC 6 DEVELOPMENT. Microsoft; Estados Unidos, 1999.
- ♦ BILDING WEB APLICATIONS USING VISUAL INTERDEV 6.0. Microsoft; Estados Unidos, 2000.
- ♦ ENTERPRISE APLICATION SERVER ADMINISTRATION, STUDENT GUIDE. Versión 1.0. Sybase Inc.; Estados Unidos, 2000.

- ♦ BUILDING APLICACIONES USING POWERBUILDER AND ENTERPRISE APPLICATION SERVER, STUDENT GUIDE. Versión 1.0. Sybase Inc.; Estados Unidos, 1999.

- ♦ PERFIL INSTITUCIONAL DE LA FUNDACIÓN TELETÓN PRO-REHABILITACIÓN. FUNTER; El Salvador, 2000.

- ♦ Sitio web de Microsoft: <http://www.microsoft.com>

- ♦ Sitio web de Sybase: <http://www.sybase.com>

- ♦ Sitio web de Oracle: <http://www.oracle.com>

- ♦ Sitio web de Rational: <http://www.rational.com>

- ♦ Sitio web de Revista PC World: <http://www.pcworld.com>

GLOSARIO

ASP

Active Server Pages, Tecnología Microsoft, que consiste básicamente en un lenguaje de tratamiento de textos basado en VBScript, que puede utilizarse conjuntamente con HTML y Javascript para realizar tareas interactivas y en tiempo real con el cliente.

Bases de Datos

Conjunto de información, tablas y otros objetos organizados y presentados para servir a un propósito concreto, como facilitar la búsqueda, clasificación y reorganización de los datos.

Casos de Uso

Elementos que proveen un significado para expresar un problema en un método que es entendible por un amplio rango de stakeholders: usuarios, desarrolladores, y arquitectos; es una secuencia de acciones en un sistema funcional, que produce resultados observables de valor a un alguien en particular que interactúa con el sistema.

Diagramas de Flujos de Datos - DFD

Representación gráfica de los procesos y flujos de datos en un sistema de negocios. En su estado original los DFD muestran el panorama más amplio posible de entradas, procesos y salidas de un sistema de información; tienen como finalidad poder dar a entender el flujo de la información entre diferentes entidades, detallando los cambios que esta información pueda tener en los diferentes procesos descritos en los mismos.

Enlace

Conocido también como Link, es una palabra, una frase o un gráfico de un hipertexto que conecta con otra información.

HTML

HyperText Markup Language, es un lenguaje simple utilizado para crear documentos de hipertexto para WWW.

Internet

Red global de computadoras que está formada por todas las redes individuales enlazadas entre sí en todo el mundo utilizando el protocolo TCP/IP.

Intranet

Redes tipo Internet pero que son de uso interno, por ejemplo, la red corporativa de una empresa que utilizara protocolo TCP/IP y servicios similares como WWW.

Middleware

Es software que se coloca entre una computadora más pequeña, denominada cliente, y una más grande, denominada servidor.

Navegador

Conocido también como Browser, es un software de aplicación usado para visualizar documentos WWW y navegar por Internet. Los dos navegadores más populares son Netscape Navigator y Microsoft Internet Explorer.

Proceso Unificador Rational (RUP)

El Proceso Unificador de Rational (RUP, *Rational Unified Proces*), es una metodología que sirve para levantar información, administrar requerimientos, controlar los cambios, reutilizar de componentes, hacer pruebas sistemáticas, administrar los recursos; todo esto con el fin de gestionar de una forma eficiente los proyectos informáticos en el menor tiempo posible y de la manera mas efectiva.

Servidor Web

Computadora que envía o sirve páginas Web.

Sitio Web

Es una localización en el World Wide Web.

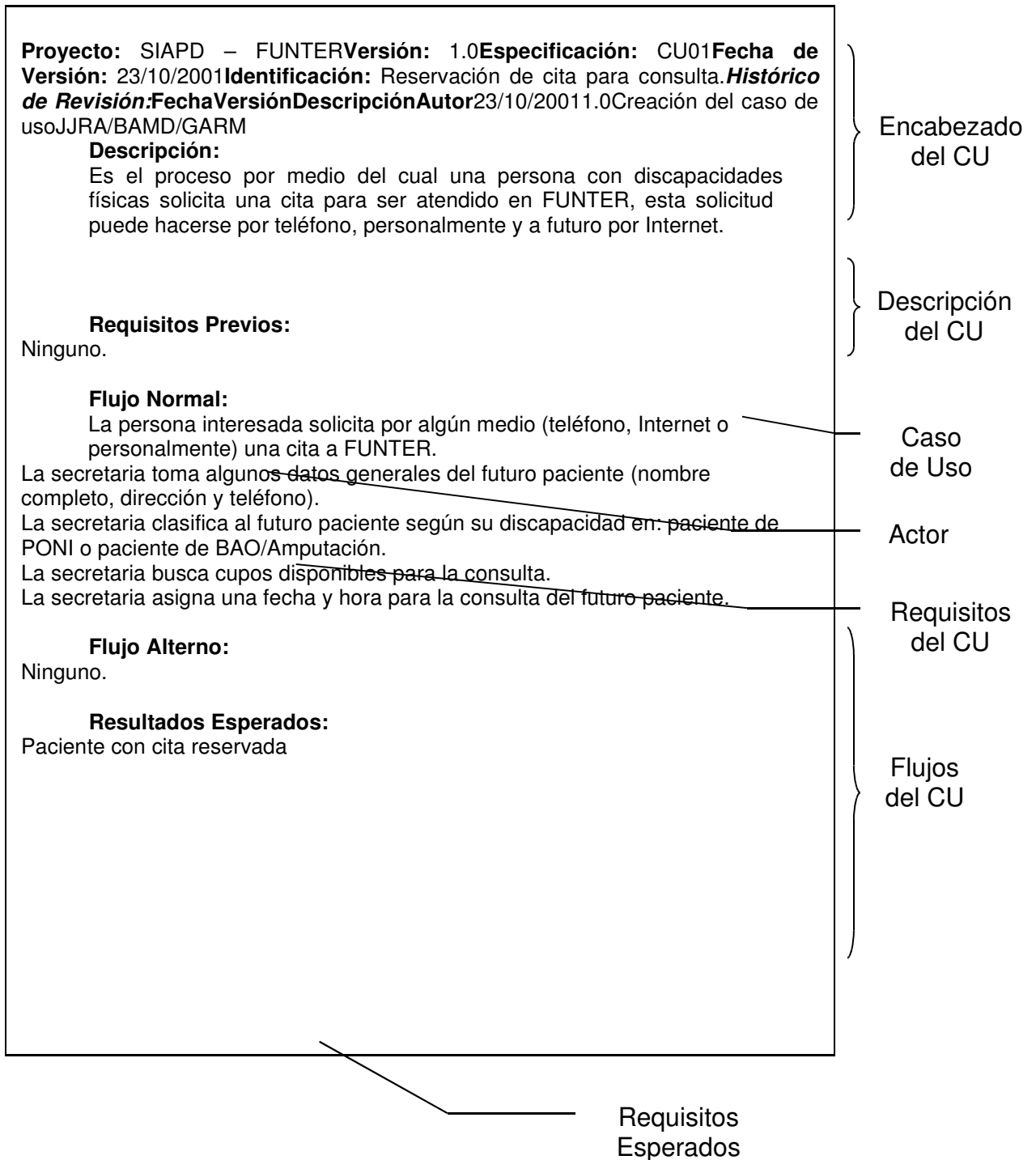
Sistemas Operativos

Es el encargado de brindar al usuario una forma amigable y sencilla de operar, interpretar, codificar y emitir las ordenes al procesador central para que este realice las tareas necesarias y especificas para completar una orden.

WWW

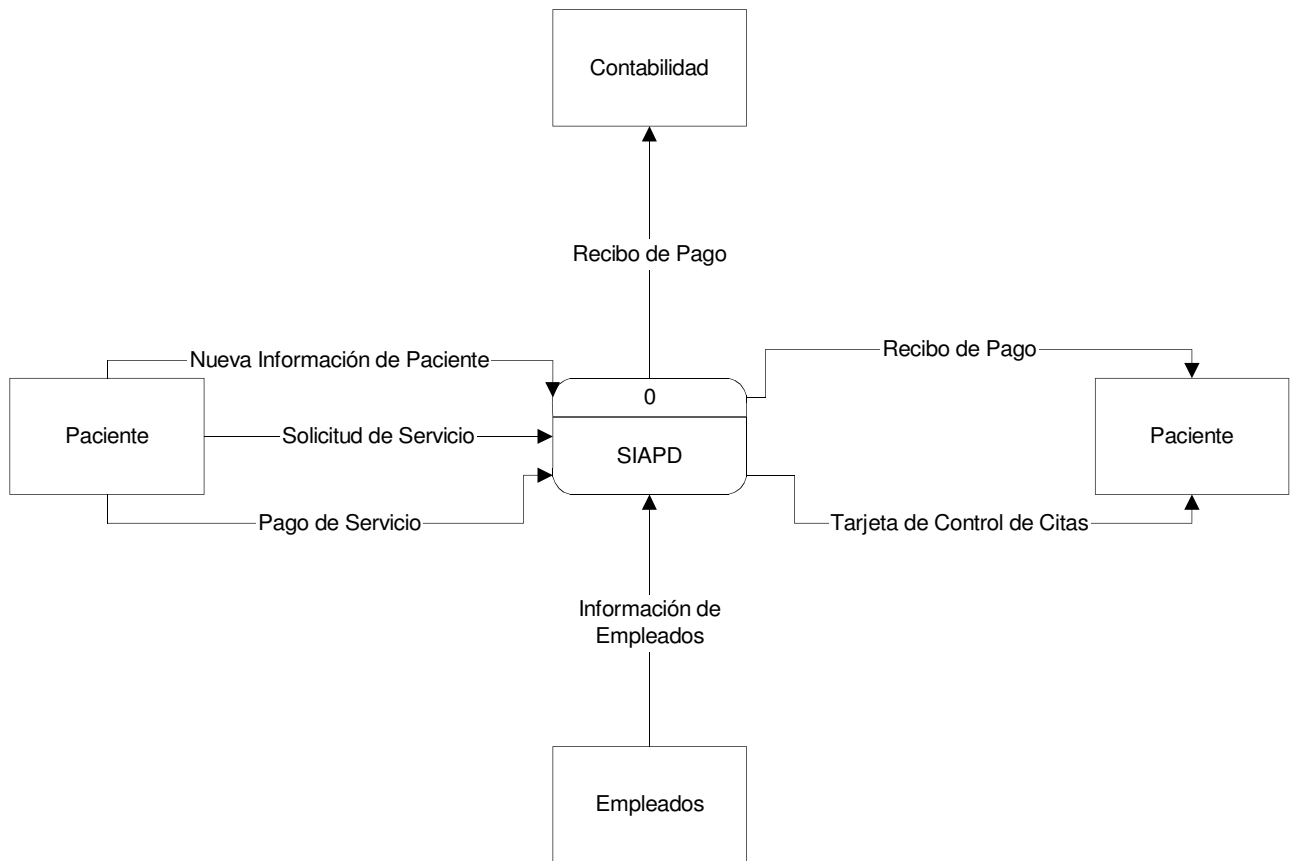
World Wide Web, es un sistema de información que permite acceder remotamente a documentos electrónicos organizados en forma de hipertexto que incorporan imágenes y sonido.

ANEXO I: Plantilla para describir los Casos de Uso (CU).

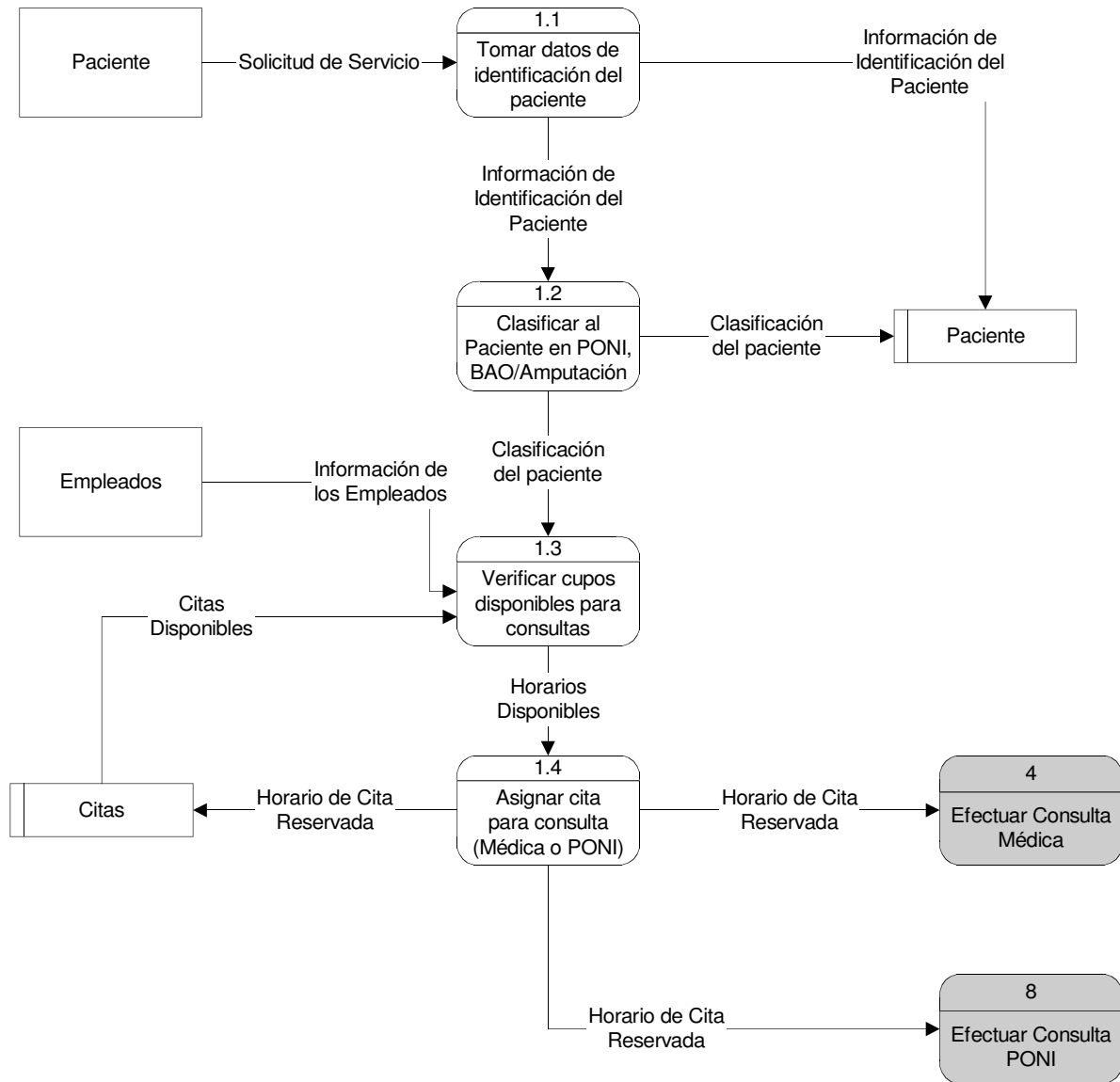


ANEXO II: DIAGRAMAS DE FLUJOS DE DATOS DE LA APLICACIÓN.

1. DFD: de Contexto del Sistema de Atención a Pacientes con Discapacidades.

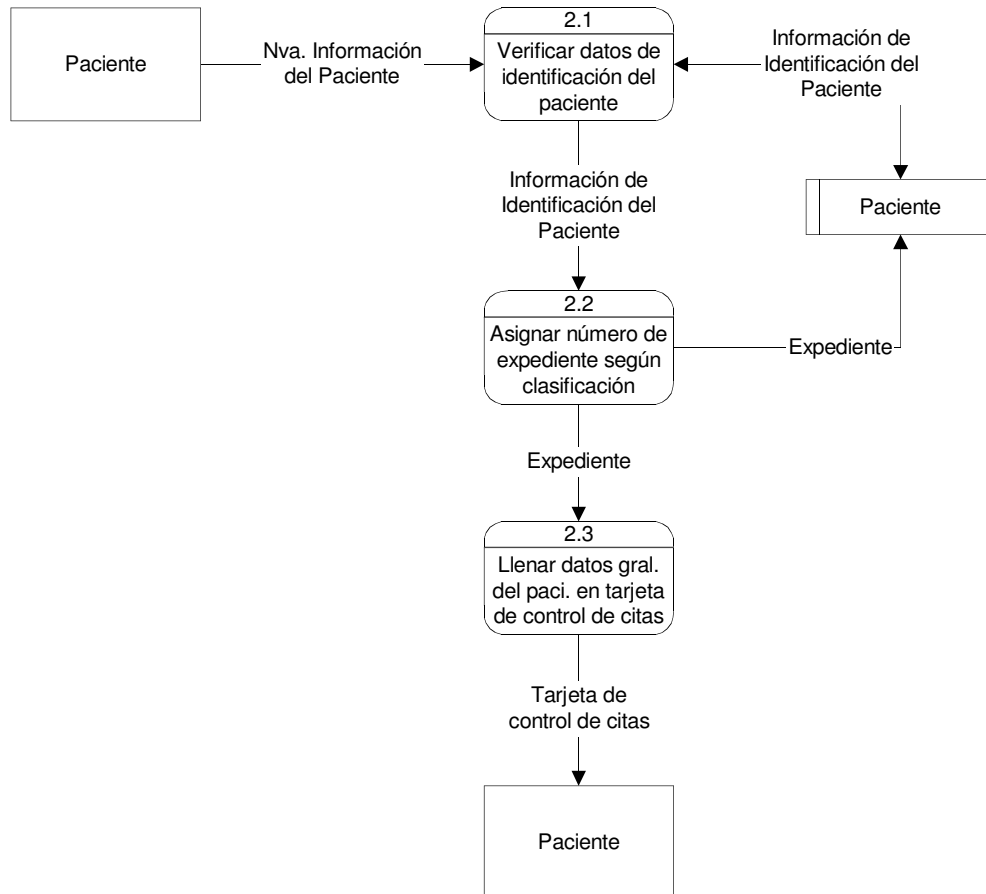


3. DFD: Reservar Cita.



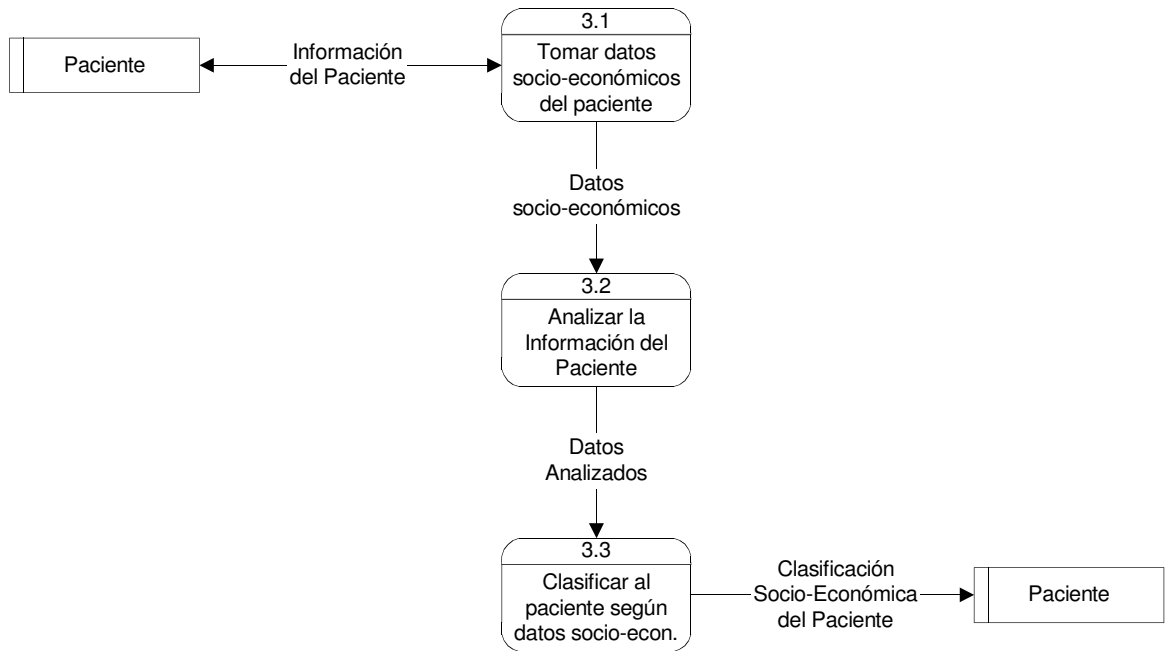
Nivel 0

4. DFD: Registrar al Paciente.



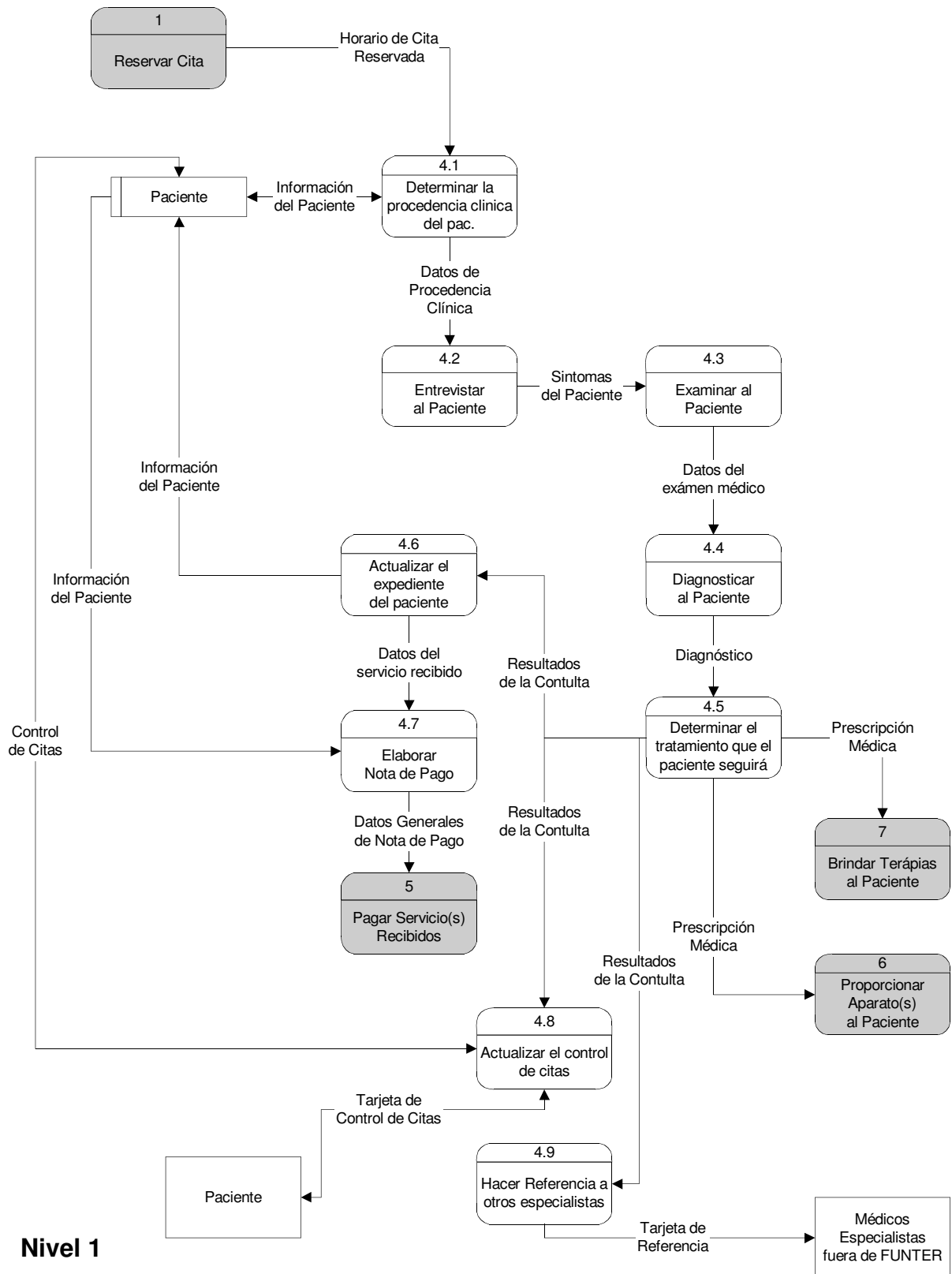
Nivel 1

5. DFD: Realizar Estudio Socio-Económico del Paciente.

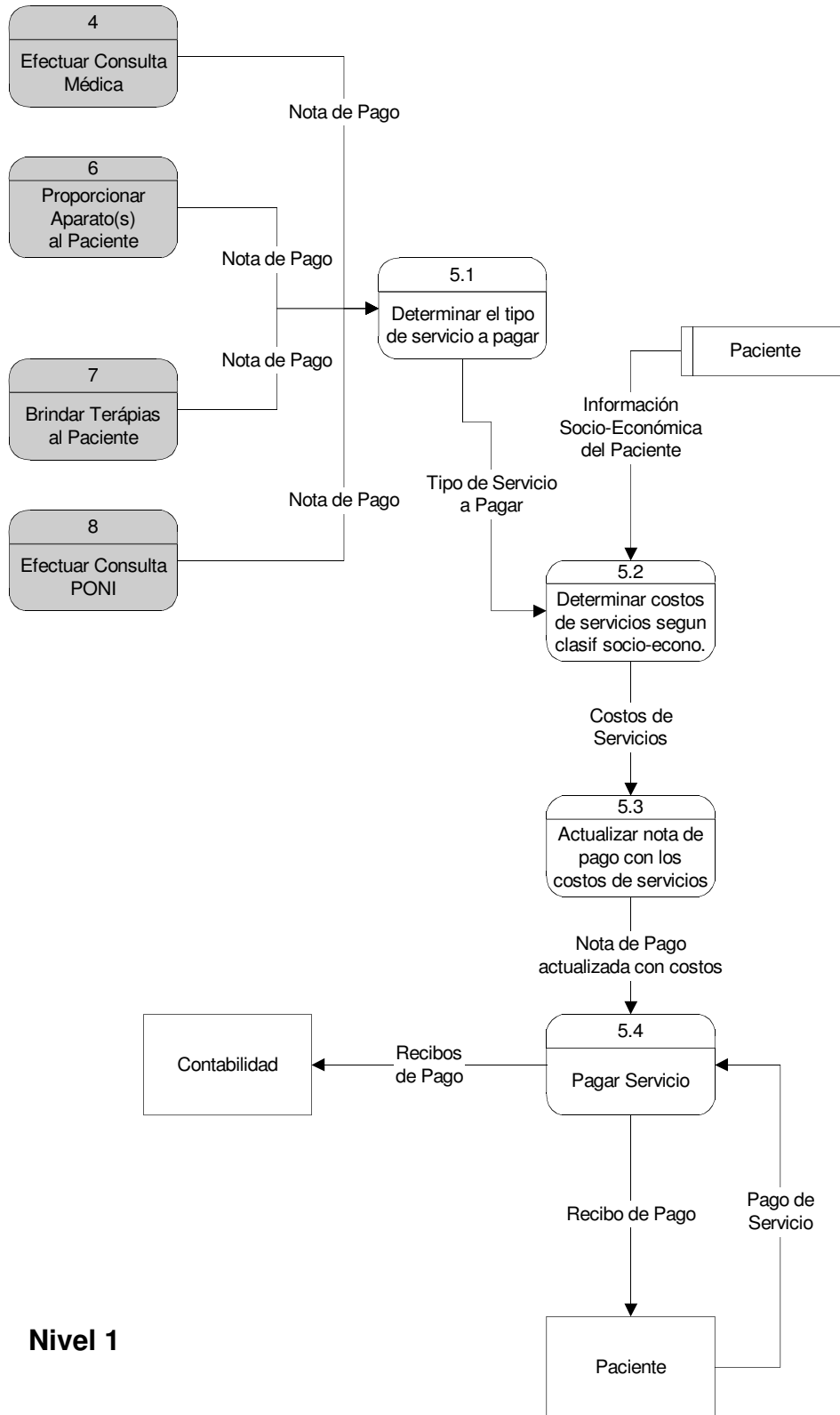


Nivel 1

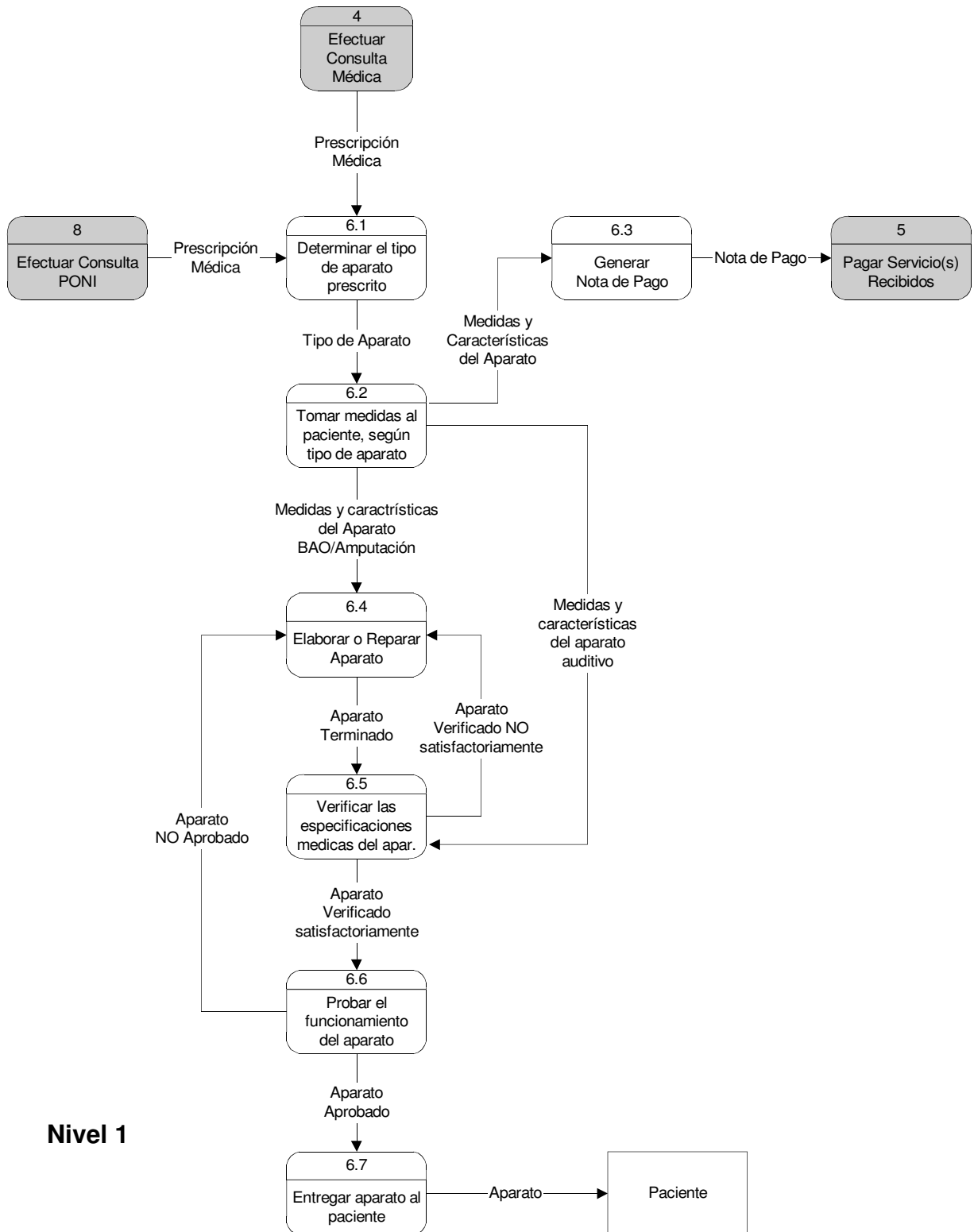
6. DFD: Efectuar Consulta Médica.



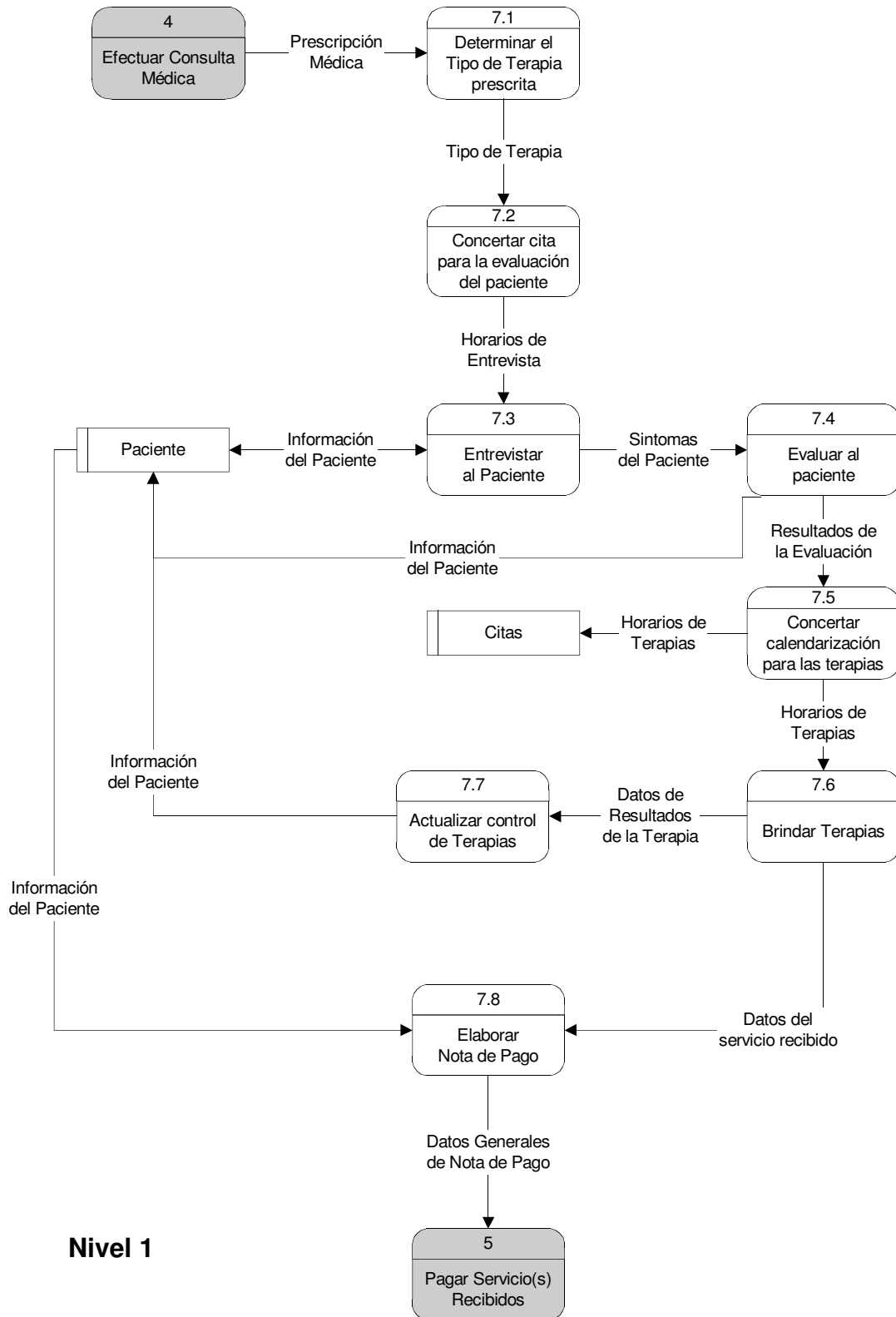
7. DFD: Pagar Servicios Recibidos.



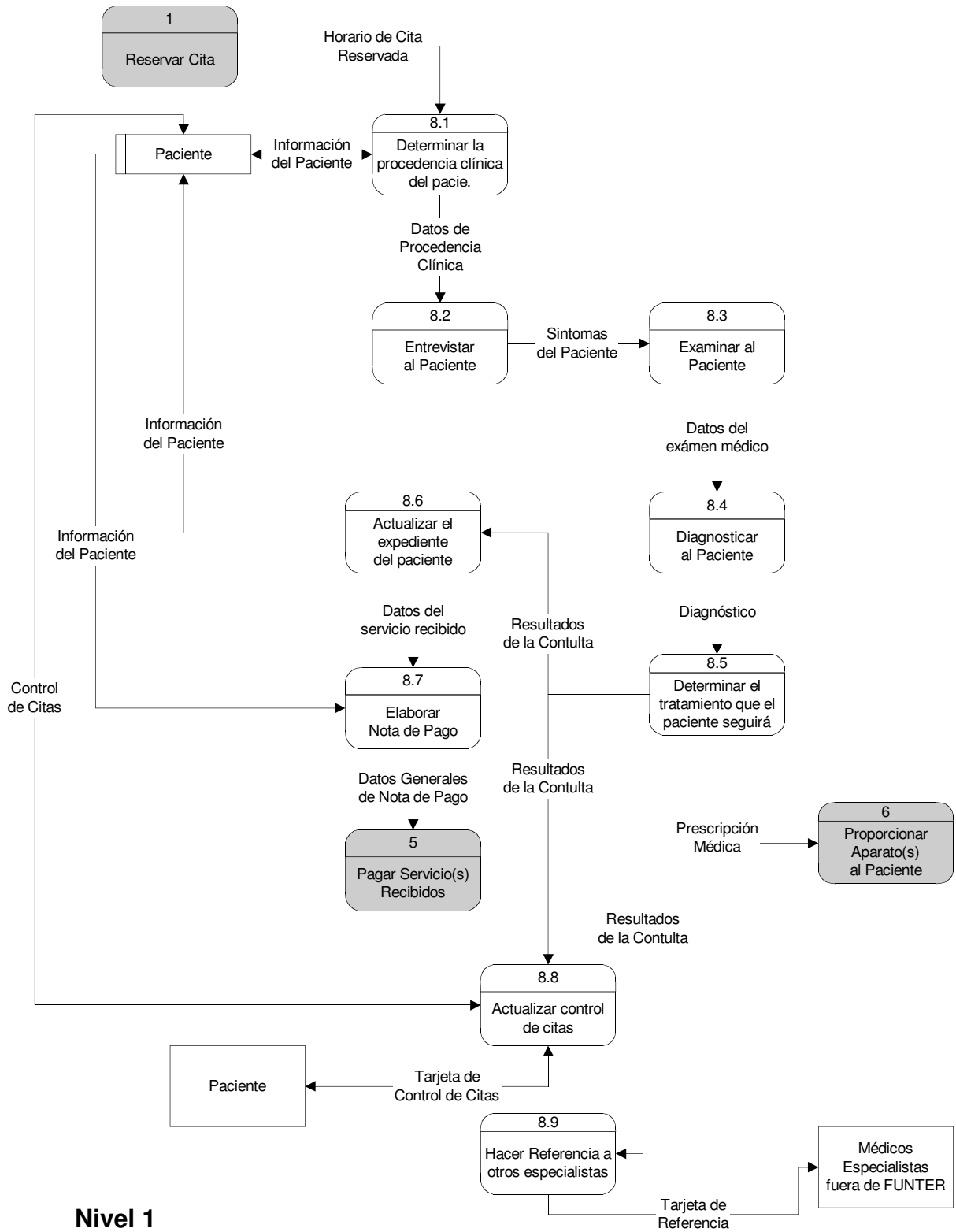
8. DFD: Proporcionar Aparatos al Paciente.



9. DFD: Brindar Terapias al Paciente.



10. DFD: Efectuar Consulta PONI.



ANEXO III. Casos de Uso de la Aplicación.

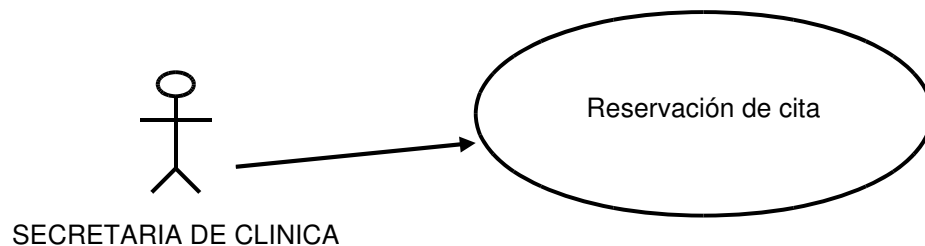
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU01	Fecha de Versión: 23/10/2001
Identificación: Reservación de cita para consulta.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
23/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Es el proceso por medio del cual una persona con discapacidades físicas solicita una cita para ser atendido en FUNTER, esta solicitud puede hacerse por teléfono, personalmente y a futuro por Internet.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ La persona interesada solicita por algún medio (teléfono, Internet o personalmente) una cita a FUNTER.
- ◆ La secretaria toma algunos datos generales del futuro paciente (nombre completo, dirección y teléfono).
- ◆ La secretaria clasifica al futuro paciente según su discapacidad en: paciente de PONI o paciente de BAO/Amputación.
- ◆ La secretaria busca cupos disponibles para la consulta.

- ◆ La secretaria asigna una fecha y hora para la consulta del futuro paciente.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados:

- ◆ Paciente con cita reservada.

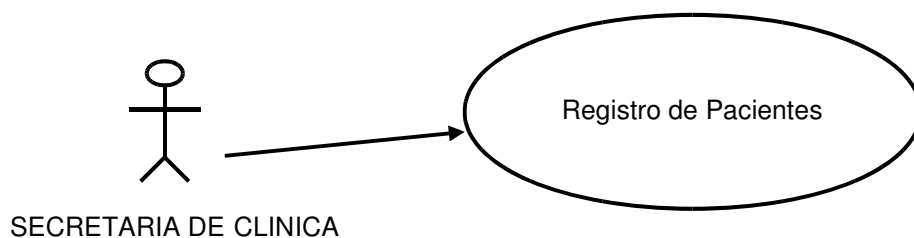
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU02	Fecha de Versión: 30/09/2001
Identificación: Registro de Pacientes.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/09/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

En este proceso se registrará al paciente que tiene alguna discapacidad física por falta de algún miembro o por disfunción física; en este proceso se le asignará un número de expediente al paciente, si aun no lo tuviera.



Requisitos Previos:

- ◆ Reservación de cita (caso de uso CU01).
- ◆ El paciente presentará a la secretaria un documento de identificación personal.

Flujo Normal:

- ◆ Se identifica al paciente según su documento de identidad personal.
- ◆ Se verifica que el paciente tenga una cita reservada.
- ◆ Se asigna al paciente un número de expediente, si aun no lo tuviere, según su clasificación (PONI o BAO/Amputación).
- ◆ Se llena la “hoja de control de citas” con los datos generales del paciente y se le entrega a este.

Flujo Alternativo:

- ◆ Si el paciente ya tiene expediente abierto, entonces solo se actualiza la tarjeta del control de citas.

Resultados Esperados:

- ◆ Asignación de número de expediente al paciente.
- ◆ Entrega al paciente de la tarjeta de citas con los datos generales del mismo.

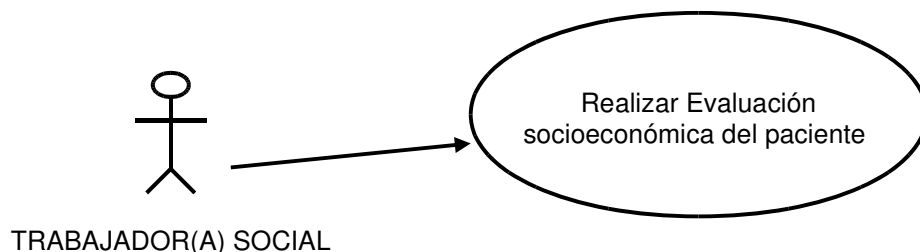
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU03	Fecha de Versión: 24/10/2001
Identificación: Evaluación socio-económica del paciente.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
24/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

En este proceso se evalúa la capacidad económica del paciente, con el fin de poder clasificarlo según dicha capacidad.



Requisitos Previos:

- ◆ Que el paciente haya pasado con la secretaria de la clínica y se le haya abierto un expediente, además que se haya clasificado al paciente según su discapacidad física en: paciente PONI o paciente BAO/Amputación.

Flujo Normal:

- ◆ Se toman datos del paciente clasificados como “Datos de Identificación general”:
 - Nombre completo.
 - Sexo.
 - Estado civil.
 - Lugar y fecha de nacimiento.
 - Documentos de identidad (número, lugar y fecha de expedición).
 - Dirección de residencia.

- Teléfono de residencia.
- ◆ Se toman datos del paciente clasificados como “Aspectos socioeconómicos”:
 - Composición del grupo familiar del paciente:
 - Nombre completo del familiar.
 - Parentesco con el paciente.
 - Lugar y fecha de nacimiento.
 - Dirección y teléfono de residencia del familiar.
 - Escolaridad y ocupación del familiar.
 - Salario mensual del familiar.
 - Determinar si el familiar es el responsable o encargado del paciente.
 - Dinámica familiar:
 - Relaciones interpersonales en el hogar.
 - ¿Quién y como se toman las decisiones?
 - Actitudes de los miembros de la familia ante la limitación del usuario.
 - En cuales actividades familiares participa el usuario y de que manera
 - Contexto comunitario:
 - Características de la comunidad donde vive el usuario
 - Recursos humanos locales que el usuario conoce y que podrían apoyar acciones de rehabilitación dirigidos a grupos o personas individuales.
 - ¿Conoce a otras personas con discapacidades en su comunidad?
 - Descripción de actividades en que el usuario y su familia participan dentro de la comunidad.
 - Antecedentes de salud:
 - Causa de la discapacidad.
 - Fecha en que ocurrió la causa,
 - Limitación de otras funciones a demás de las discapacidad señalada.
 - Tratamiento medico recibido o que recibe.
 - En caso de Amputaciones:
 - Lugar donde se le hizo la amputación.
 - ¿Cómo se moviliza el usuario en el momento de la entrevista inicial?

- Antecedentes académicos:
 - Escolaridad alcanzada.
 - Fecha de la escolaridad alcanzada.
 - Nombre y dirección del centro educativo.
 - Capacitaciones técnicas recibidas.
 - Observaciones.
- ◆ Se toman datos del paciente clasificados como “Aspectos Laborales”:
 - Experiencia de trabajo:
 - Empresa donde trabajo.
 - Puesto ocupado.
 - Salario mensual.
 - Duración (fecha de inicio y fecha de finalización).
 - Motivo del retiro.
 - Descripción de Tareas realizadas.
 - Si no trabaja actualmente, ¿por qué motivo?
 - Ocupación e intereses vocacionales:
 - Intereses y motivaciones expresadas por el usuario hacia su formación profesional.
 - Opinión de la familia acerca de las inclinaciones vocacionales del usuario.
 - Labores que le gustaría desempeñar al usuario según área de actividad (Comercio, Industria, Servicios, Agricultura, Otros)
 - Tiene alguna experiencia sobre los oficios expresados? (especificar tareas)
 - Qué lo motiva a optar por ese oficio?
 - Obstáculos que se han presentado al usuario para la integración al trabajo:
 - Físicos.
 - Sociales.
 - Conductuales.
 - Familiares.

- Gastos y Egresos:
 - Vivienda (Tipo de vivienda y forma de pago)
 - Alimentación.
 - Cuotas Escolares.
 - Servicio de energía eléctrica.
 - Servicio de agua.
 - Vestuario.
 - Vivienda.
 - Deudas.
 - Otros.
 - Tipo de Financiamiento (protésico).
- ◆ Se toman datos del paciente clasificados como “Datos complementarios”:
 - Actitudes de usuario durante la entrevista.
 - Conclusiones y Recomendaciones.
- ◆ Se clasifica al paciente en rangos, según su capacidad económica, esto con el fin de poder establecer posteriormente la cuota de pago del paciente.

Flujo Alternativo:

- ◆ Actualización de datos del expediente, a solicitud del paciente para posible reclasificación del paciente.

Resultados Esperados:

- ◆ Tener actualizado el expediente con datos referentes a la identificación del paciente y a su estudio socio-económico.
- ◆ Que el paciente esté clasificado en un rango, según su capacidad de pago.

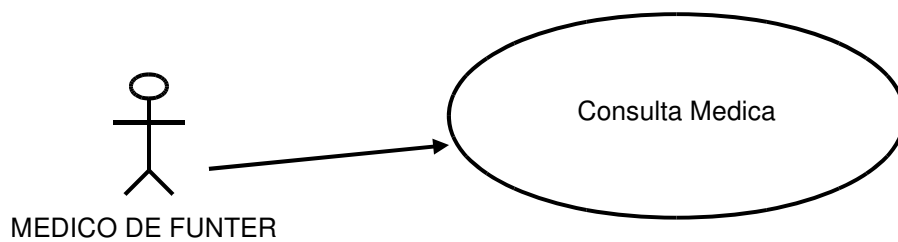
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU04	Fecha de Versión: 24/10/2001
Identificación: Consulta Médica.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
24/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

El medico de FUNTER revisa al paciente para determinar cual es el tipo de discapacidad física que este presenta y cual será el tratamiento a seguir: ortesis, prótesis o terapia de rehabilitación (física o lenguaje).



Requisitos Previos:

- ◆ Que el paciente tenga abierto un expediente.
- ◆ Que al paciente se le haya realizado el estudio socioeconómico y se le haya clasificado en un rango según su capacidad de pago.
- ◆ Que el paciente tenga reservada una cita para la consulta.

Flujo Normal:

- ◆ Determinar la procedencia del paciente en cuanto a la consulta: Si es la primera consulta en FUNTER, si el paciente está en tratamiento de terapia y/o si el paciente está siendo referido de una consulta PONI.
- ◆ El paciente dice al doctor cuales son sus molestias (síntomas).
- ◆ El medico examina al paciente y determina el problema de este. (discapacidad, mal funcionamiento de algún miembro, problemas auditivos, etc.)

- ◆ El medico determina el tipo de tratamiento para el paciente:
 - Referencia a alguna terapia (física, lenguaje, etc), en este caso el médico llenará la hoja de “Servicio de medicina física y rehabilitación”:
 - Nombre del paciente.
 - Diagnostico.
 - Área.
 - Fecha de ingreso.
 - Número de expediente del paciente.
 - Terapia indicada.
 - Prescripción de órtesis y/o prótesis, y otros medicamentos; en este caso el médico llenará la hoja de “Prescripción medica”:
 - Fecha.
 - Nombre del paciente.
 - Número de expediente del paciente.
 - Receta.
 - Firma del médico.
 - Referencia a otros especialistas, en este caso el médico debe llenar la tarjeta de “Tarjeta de referencia”:
 - Fecha de referencia.
 - Nombre del paciente.
 - Edad del paciente.
 - Diagnóstico.
 - Institución médica a la que el paciente será referido.
 - Plan de tratamiento o consultas sugeridas.
 - Médico que refiere al paciente.
- ◆ Se actualizan los datos en la tarjeta de control de citas del paciente.
- ◆ Se elabora la nota de pago por el servicio recibido (no se le coloca el costo del servicio), con la siguiente información:
 - Nombre del paciente.
 - Número de expediente del paciente.
 - Servicio recibido.

Flujo Alternativo:

- ◆ Que el paciente sea referido por una consulta PONI a una consulta médica.
- ◆ Que el paciente que está recibiendo terapia tenga una consulta de control.

Resultados Esperados:

- ◆ Hoja de Servicio de medicina física y rehabilitación.
- ◆ Hoja de prescripción médica (medicamentos, prótesis, ortesis, etc.).
- ◆ Referencia a otras entidades especializadas en la atención a pacientes con discapacidades físicas.
- ◆ Tarjeta de control de citas actualizada.
- ◆ Nota de pago elaborada.

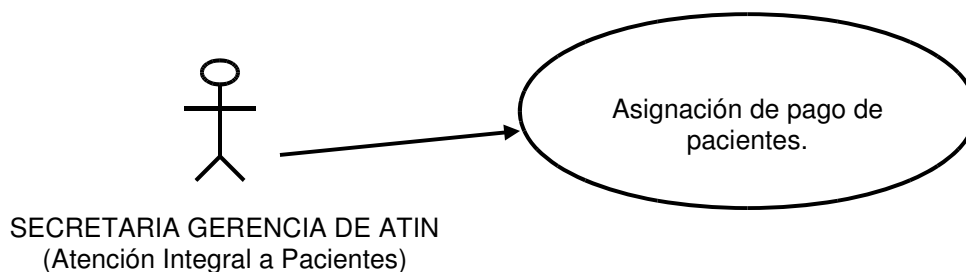
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU05	Fecha de Versión: 2210/2001
Identificación: Asignación de pagos de pacientes.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
2210/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

En este proceso se hace el calculo exacto de la cantidad que deberá cancelar el paciente por cada uno de los servicios y/o aparatos recibidos, según la clasificación en que se encuentre por su capacidad de pago.



Requisitos Previos:

- ◆ Que el paciente haya pasado con la trabajadora social y se le haya asignado el rango de contribución.
- ◆ Que el paciente haya recibido algún servicio y/o aparato de parte de FUNTER.
- ◆ Que exista una nota de pago con los datos del servicio recibido por el paciente.

Flujo Normal:

- ◆ Determinar el tipo de servicio recibido por el paciente, en base a nota de pago.
- ◆ Determinar el costo de cada servicio y/o aparato recetado al paciente, según tabla de costos de FUNTER, en base a las clasificaciones de los pacientes.
- ◆ Complementar la información de la nota de pago:
 - Prima y/o abono de pago (por cada uno de los servicios recibidos).
 - Forma de pago (crédito o contado).

- Cantidad de cuotas a pagar, en el caso de que el pago sea al crédito.
- ◆ El paciente efectuará su pago en la caja, según los montos reflejados en la nota de pago.

Flujo Alternativo:

- ◆ Si el paciente no tiene como pagar los servicios y/o aparatos recibidos será atendido de forma gratuita, siempre y cuando haya sido determinado por la trabajadora social.

Resultados Esperados:

- ◆ Nota de pago llena y cancelada.

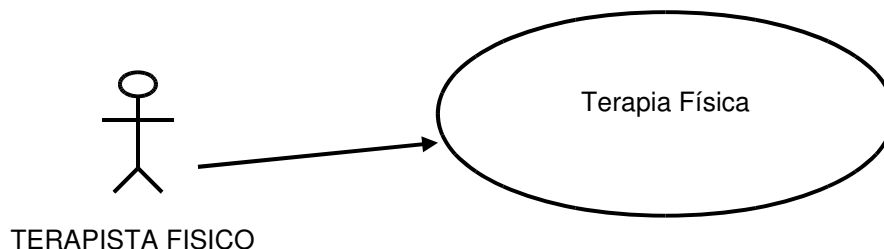
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU06	Fecha de Versión: 24/10/2001
Identificación: Terapia Física.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
24/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

El paciente se presenta a recibir su terapia física con el terapeuta en la fecha y hora acordada previamente.



Requisitos Previos:

- ◆ Que el paciente haya recibido la consulta médica y se le prescriba alguna terapia en FUNTER.
- ◆ Que el paciente tenga actualizada su Hoja de “Servicio de medicina física y rehabilitación” para el control de las terapias.
- ◆ Que el paciente tenga un calendario de citas programadas para sus terapias, incluyendo una cita para la evaluación que realizará el terapeuta.

Flujo Normal:

- ◆ En la primera cita, el terapeuta entrevista al paciente para llenar la Hoja de Evaluación de Fisioterapia:
 - Nombre del paciente.
 - Sexo del paciente.
 - Edad del paciente.
 - Ocupación del paciente.

- Dirección del paciente.
- Diagnóstico médico.
- Causa de amputación.
- Tratamiento de Fisioterapia.
- ◆ Se le brinda la terapia del paciente.
- ◆ Se actualiza la hoja de control de terapias:
 - Nombre del paciente.
 - Número de expediente.
 - Edad.
 - Sexo.
 - Tipo de atención (entrevista, subsiguiente)
 - Diagnóstico.
 - Hora de cita.
- ◆ Se actualiza la Hoja de “Servicio de Medicina Física y Rehabilitación”
- ◆ El terapeuta puede remitir al paciente a consulta médica, para que el médico evalúe cómo el paciente está respondiendo a la terapia.
- ◆ Se elabora la nota de pago por el servicio recibido (no se le coloca el costo del servicio), con la siguiente información:
 - Nombre del paciente y Número de expediente del paciente.
 - Servicio recibido.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados:

- ◆ Hoja de Evaluación de Fisioterapia del paciente llena con sus datos correspondientes.
- ◆ Hoja de Control de Terapias del paciente llena con sus datos correspondientes.
- ◆ Hoja de Servicio de medicina física y rehabilitación actualizada.

Proyecto: SIAPD – FUNTER	Versión: 1.0
---------------------------------	---------------------

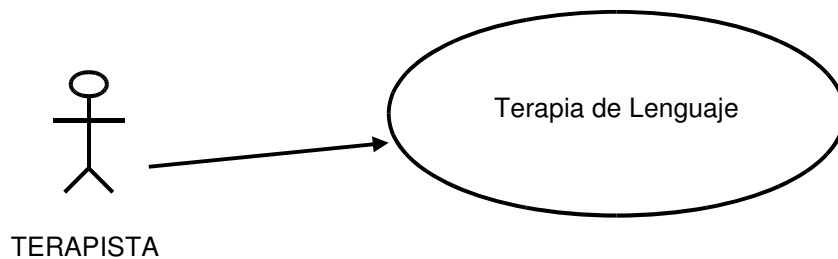
Especificación: CU07	Fecha de Versión: 23/10/2001
Identificación: Terapia de lenguaje.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
23/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Se remiten a este tipo de terapia al paciente que muestra problemas de comunicación referente al habla, por lo general es el medico quien determina que la persona necesita este tipo de terapia.



Requisitos Previos:

- ◆ Que el medico haya dado una referencia para que el paciente reciba esta terapia. (caso de uso CU04).
- ◆ Que el paciente tenga un calendario de citas programadas para sus terapias, incluyendo una cita para la evaluación que realizará el terapeuta.

Flujo Normal:

- ◆ El terapeuta de lenguaje evalúa al paciente y llena la respectiva hoja de evaluación:
 - Nombre del paciente.
 - Número de expediente.
 - Fecha de nacimiento del paciente.
 - Fecha de evaluación.
 - Antecedentes personales.
 - Fecha que inicio el problema.

- Tipo de tratamiento que ha recibido hasta el momento (descripción, lugar, tiempo).
- ¿Por qué el paciente dejó de asistir a recibir el tratamiento?
- Cambios observados con los tratamientos.
- Alimentación:
 - El paciente babea.
 - El paciente come sólidos.
 - El paciente toma líquidos.
 - El paciente traga o devuelve la comida.
- Lenguaje:
 - Formas en que el paciente se comunica actualmente.
 - ¿Emite algún sonido o palabras?, ¿Cuáles?
 - ¿Es capaz de mantener conversaciones o expresar lo que quiere?
 - ¿Coordina bien sus ideas al expresarse?
 - ¿Se bloquea al hablar?, ¿con que frecuencia? (alta, media, baja)
- Control de movimientos orofaciales:
 - Lengua.
 - Labios.
 - Mandíbula.
 - Cierre bucal.
- Articulaciones de fonemas.
- Observaciones.
- ◆ Se le brinda la terapia del paciente.
- ◆ Se actualiza la hoja de historial clínico:
 - Nombre del paciente.
 - Número de expediente.
 - Edad.
 - Terapeuta.
 - Antecedentes del embarazo, familiares y ambientales.
 - Alimentación.
 - Motricidad, sueño, dentición y controles esfinterianos.

- Enfermedades u operaciones.
- Evolución del lenguaje y aprendizaje.
- Motivo de la evaluación.
- Observaciones del evaluador.
- ◆ Se actualiza la hoja de control de terapias:
 - Nombre del paciente.
 - Número de expediente.
 - Frecuencia de la terapia.
 - Firma del terapeuta.
 - Control de asistencia:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ENE																															
:																															
DIC																															

- ◆ El terapeuta puede remitir al paciente a consulta médica, para que el médico evalúe cómo el paciente está respondiendo a la terapia.
- ◆ Se elabora la nota de pago por el servicio recibido (no se le coloca el costo del servicio), con la siguiente información:
 - Nombre del paciente.
 - Número de expediente del paciente.
 - Servicio recibido.

Flujo Alternativo:

- ◆ Si el terapeuta no tiene horas disponibles para un paciente específico, entonces se busca hacer grupos de pacientes con problemas comunes.
- ◆ Si no es posible hacer grupos de pacientes con problemas comunes, el paciente debe esperar hasta que haya cupo disponible.

Resultados Esperados:

- ◆ Hoja de Evaluación del paciente llena con sus datos correspondientes.
- ◆ Calendario de terapias actualizado.
- ◆ Hoja de historial clínico del paciente actualizada.

- ◆ Tarjeta de control de asistencias a terapias del paciente actualizada.

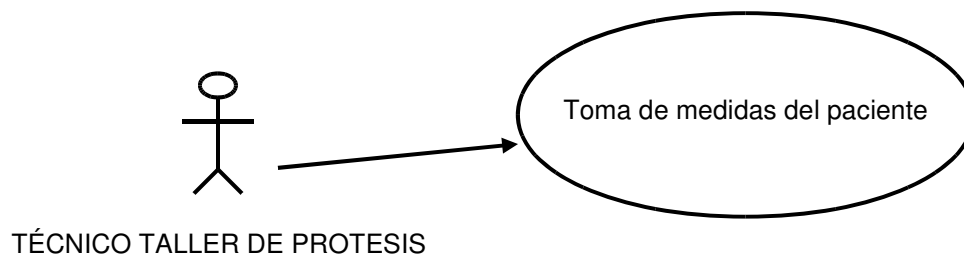
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU08	Fecha de Versión: 24/10/2001
Identificación: Toma de medidas del paciente p/ la elaboración de su aparato.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
24/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Se toman las medidas del paciente según el tipo de aparato que se le haya prescrito (ortesis o prótesis) y según el lugar de su cuerpo en el cual lo necesita (extremidades inferiores o extremidades superiores).



Requisitos Previos:

- ◆ Que el paciente haya recibido su consulta medica y que el medico haya prescrito el uso de algún aparato.

Flujo Normal:

- ◆ El paciente se presenta el día establecido por FUNTER, para que le tomen las medidas.
- ◆ El técnico verifica la información del aparato prescrito al paciente:
 - Nombre del paciente.
 - Número de expediente del paciente.
 - Datos de la consulta médica:
 - Fecha de la consulta.
 - Prescripción del aparato a fabricar.
 - Médico que prescribió el aparato.

- ◆ Se verifica que se haya efectuado la toma de medidas del paciente según el lugar de su cuerpo donde se le ha prescrito el aparato.
- ◆ Se determina la fecha en que el paciente y el medico probaran el aparato a ser elaborado.
- ◆ Se elabora la nota de pago por el servicio recibido (no se le coloca el costo del servicio), con la siguiente información:
 - Nombre del paciente.
 - Número de expediente del paciente.
 - Servicio recibido.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados:

- ◆ Queda registrado en el sistema que el paciente se presento a tomarse las medidas para el aparato prescrito y se le asigno una fecha y hora para que llegara ha hacerse la prueba del aparato.

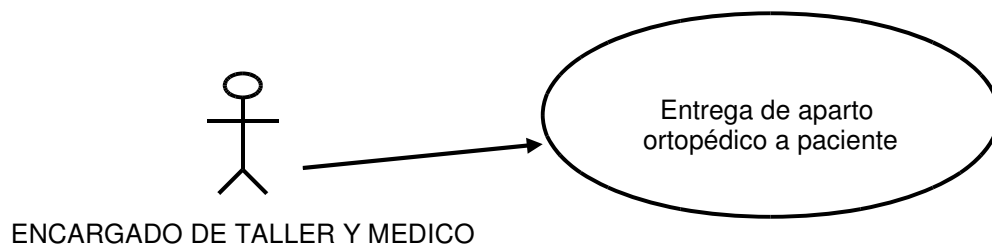
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU09	Fecha de Versión: 24/10/2001
Identificación: Entrega del aparato a paciente.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
24/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Después de terminar la fabricación o reparación de un aparato, se evalúa por parte del medico, para comprobar que cumple con las indicaciones especificadas, si el medico aprueba el aparato y el paciente da su visto bueno del mismo y además a cancelado el aparato entonces se le hace entrega del mismo.



Requisitos Previos:

- ◆ Que el técnico haya asignado previamente al paciente la fecha en que este probará su aparato.
- ◆ Que el aparato este listo para ser entregado.

Flujo Normal:

- ◆ El medico comprueba que el aparato ha sido elaborado o reparado bajo las especificaciones señaladas.
- ◆ El paciente se prueba el aparato.
- ◆ El técnico deberá llenar la hoja de “Producto terminado”:
 - Orden de producción.
 - Reparación / Fabricación.

- Tipo de Prótesis.
 - Ortesis.
 - Otros.
 - Nombre del protesista.
 - Nombre del paciente.
 - Observaciones.
- ◆ Se entrega el aparato al paciente.

Flujo Alternativo:

- ◆ Si el aparato no pasa la aprobación por parte del médico, entonces deben hacerse los ajustes necesarios, para corregir el aparato. En este caso el técnico debe ponerse de acuerdo con el paciente en el día y la hora en que nuevamente regrese para probarse el aparato.

Resultados Esperados:

- ◆ Hoja de "producto terminado" llena con los datos correspondientes.
- ◆ Hoja "Entrega de producto terminado a paciente" impresa en tres copias para: contabilidad, paciente y bodega.

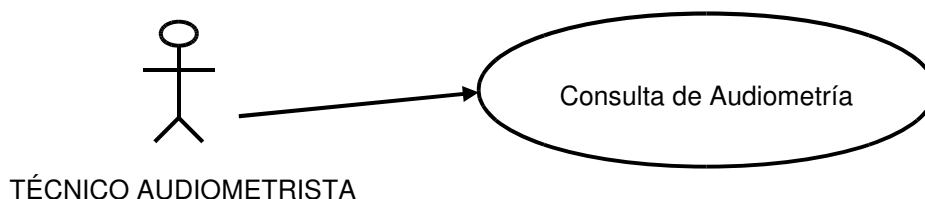
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU10	Fecha de Versión: 30/10/2001
Identificación: Consulta de Audiometría (PONI).	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

El paciente pasa consulta con el técnico audiometrista cuando tiene molestias relacionadas al oído, ya sea sordera, obstrucción por objetos extraños, golpes, etc.



Requisitos Previos:

- ◆ Que el paciente tenga abierto un expediente.
- ◆ Que al paciente se le haya realizado el estudio socioeconómico y se le haya clasificado en un rango según su capacidad de pago.
- ◆ Que el paciente tenga reservada una cita para la consulta.

Flujo Normal:

- ◆ Se determina la procedencia del paciente en cuanto a la consulta: Si es la primera consulta en FUNTER, si el paciente está en tratamiento de terapia y/o si el paciente está siendo referido de una consulta PONI.
- ◆ El técnico audiometrista actualiza el expediente PONI (Programa Protejamos el Oído de nuestros Niños) del paciente, con datos como los siguientes:
 - Datos de Identificación:
 - Nombre del paciente.

- Fecha de nacimiento.
- Edad.
- Sexo.
- Estado civil.
- Dirección.
- Teléfono.
- Ocupación.
- Lugar de trabajo.
- Escolaridad.
- Situación del paciente (posterior al examen del técnico):
 - Diagnostico.
 - Hospital/Clinica donde fue atendido..
 - Causa de la discapacidad.
 - Instituto que refiere al paciente.
 - Tratamiento Recibido.
 - Tipo de atención que necesita (Audiometría, Lavados auditivos, Cambio de cable, Cambio de audífono, Auxiliar auditivo RAMI-3, Auxiliar auditivo RAMI-7)
- Aportación económica del paciente.
- Forma de pago.
- Conformación del grupo familiar del paciente.
- Observaciones.
- ◆ El paciente dice al técnico cuales son sus molestias (síntomas).
- ◆ El técnico examina al paciente (examen de audiograma, vía aérea y/o ósea) y determina el problema de este.
- ◆ El medico determina el tipo de tratamiento para el paciente:
 - Tipo de atención que necesita (Audiometría, Lavados auditivos, Cambio de cable, Cambio de audífono, Auxiliar auditivo RAMI-3, Auxiliar auditivo RAMI-7)
 - Prescripción de aparato auditivo; en este caso el médico llenará la hoja de “Prescripción medica”:

- Fecha.
- Nombre del paciente.
- Número de expediente del paciente.
- Receta.
- Firma del médico.
- Referencia a otros especialistas, en este caso el médico debe llenar la tarjeta de “Tarjeta de referencia”:
 - Fecha de referencia.
 - Nombre del paciente.
 - Edad del paciente.
 - Diagnóstico.
 - Institución médica a la que el paciente será referido.
 - Plan de tratamiento o consultas sugeridas.
 - Médico que refiere al paciente.
- ◆ Se actualizan los datos en la tarjeta de control de citas del paciente.
- ◆ Se elabora la nota de pago por el servicio recibido (no se le coloca el costo del servicio), con la siguiente información:
 - Nombre del paciente.
 - Número de expediente del paciente.
 - Servicio recibido.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados:

- ◆ Actualización del expediente de consulta PONI.
- ◆ Audiograma del paciente realizado.
- ◆ Hoja de prescripción médica (medicamentos, prótesis, ortesis, etc.).
- ◆ Referencia a otras entidades especializadas en la atención a pacientes con discapacidades físicas.
- ◆ Tarjeta de control de citas actualizada.

- ◆ Nota de pago elaborada.

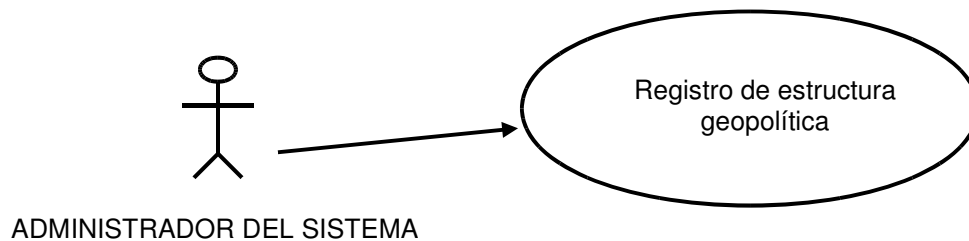
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU11	Fecha de Versión: 27/092001
Identificación: Registro de estructura geopolítica del país en el que se utilizará el SIAPD.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a la estructura geopolítica del país en el que se utilizará el Sistema de Atención a Pacientes con Discapacidades (SIAPD), indicando sus departamentos y municipios. Cada uno de los municipios solo puede pertenecer a un departamento.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ El administrador del sistema registra los datos concernientes al país.
- ◆ El administrador del sistema registra los datos de los diferentes departamentos del país.
- ◆ El administrador del sistema registra los datos de diversos municipios del país, indicando el departamento al que pertenece cada municipio.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados:

- ◆ Registro de países, departamentos y municipios en el sistema.

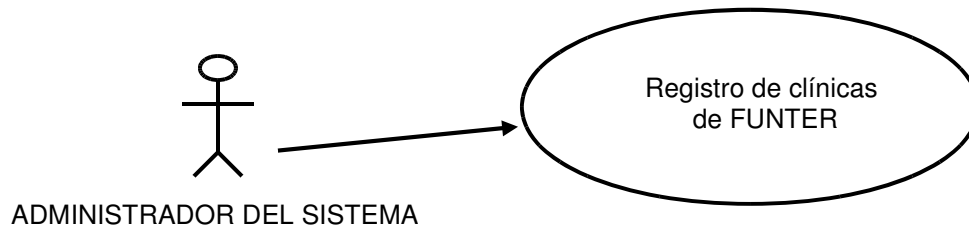
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU12	Fecha de Versión: 27/09/2001
Identificación: Registro de clínicas de FUNTER	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a los diversas clínicas de FUNTER que prestan servicios en el país.



Requisitos Previos:

- ◆ Que ya se encuentren registrados en el sistema los datos correspondientes a la estructura geopolítica del país en el que se utilizará el sistema.

Flujo Normal:

- ◆ El administrador del sistema registra los datos concernientes a la identificación de la clínica de FUNTER (nombre de la clínica).
- ◆ Se registra en el sistema los datos concernientes a la ubicación geográfica de la clínica de FUNTER, como por ejemplo su dirección, municipio, departamento, país, etc.
- ◆ Se registra en el sistema otros datos relacionados a la clínica de FUNTER, como por ejemplo sus números de teléfono, fax, etc.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados

- ◆ Datos concernientes a las clínicas de FUNTER registrados en el sistema.

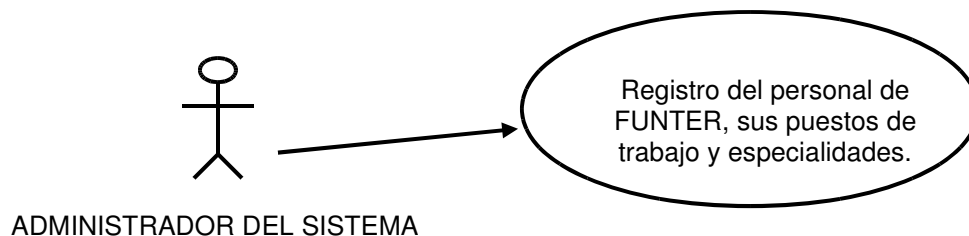
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU13	Fecha de Versión: 27/092001
Identificación: Registro del personal de FUNTER, sus puestos de trabajo y especialidades.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de datos de los diferentes empleados de FUNTER vinculados al sistema, sus respectivos puestos de trabajo y sus especialidades.



Requisitos Previos:

- ◆ Que ya se encuentren registrados en el sistema los datos correspondientes a las Clínicas de FUNTER.

Flujo Normal:

- ◆ Se registra en el sistema los datos de los diversos puestos de trabajo de FUNTER.
- ◆ Se registra en el sistema los datos de las diferentes especialidades que puede tener un empleado de la Institución.
- ◆ Se registra en el sistema los datos generales del empleado de FUNTER, su respectivo puesto de trabajo y las diferentes especialidades que este pueda tener.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados:

- ◆ Empleados de FUNTER, su puesto de trabajo y sus especialidades registradas en el sistema.

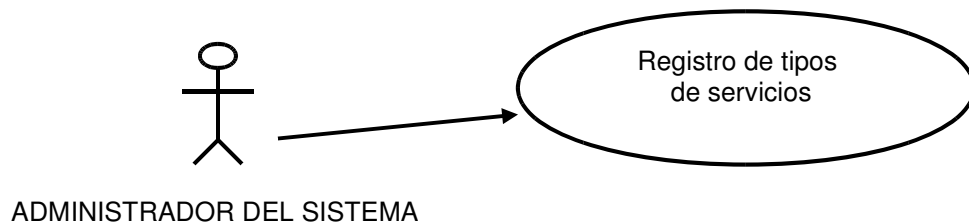
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU14	Fecha de Versión: 27/092001
Identificación: Registro de tipos de servicios brindados en FUNTER.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a los diversos tipos servicios de rehabilitación que presta la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Se identifica el tipo de servicio a ser registrado en el sistema.
- ◆ Se registra en el sistema los diversos tipos servicios de rehabilitación que presta FUNTER.

Flujo Alterno

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de servicios que se brindan en FUNTER registrados en el sistema.

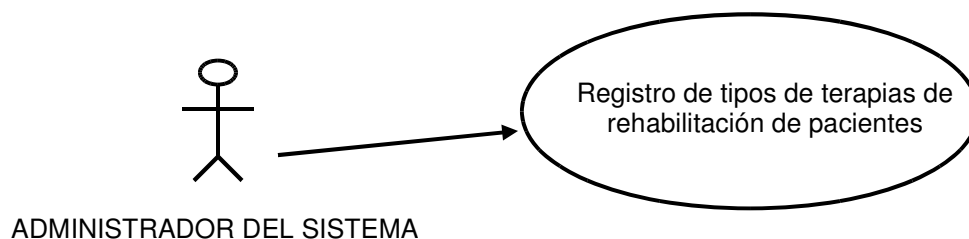
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU15	Fecha de Versión: 27/092001
Identificación: Registro de tipos de terapias de rehabilitación de pacientes	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a los diversos tipos de terapias de rehabilitación de pacientes con discapacidades que se brindan en la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Se identifican los tipos de terapia a ser registrados en el sistema.
- ◆ Se registran en el sistema los diversos tipos de terapias de rehabilitación destinadas a los pacientes con discapacidades físicas.

Flujo Alternativo:

- ◆ Ninguno.

Resultados Esperados:

- ◆ Tipos de terapias de rehabilitación de pacientes registradas en el sistema.

Proyecto: SIAPD – FUNTER	Versión: 1.0
---------------------------------	---------------------

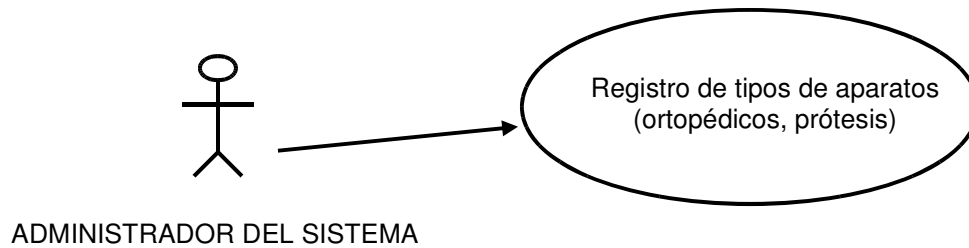
Especificación: CU16	Fecha de Versión: 27/092001
Identificación: Registro de tipos de aparatos	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a los diversos tipos de aparatos, tanto ortopédicos como protésicos, que serán diseñados y fabricados en FUNTER para que puedan ser utilizados por los diversos pacientes con discapacidades físicas en sus procesos de rehabilitación.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Se Identifican los tipos de aparatos a ser registrados en el sistema.
- ◆ Se registran en el sistema los diversos tipos de aparatos, tanto ortopédicos como protésicos, que serán diseñados y fabricados en FUNTER y que servirán en el proceso de rehabilitación de sus pacientes.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de aparato registrados en el sistema.

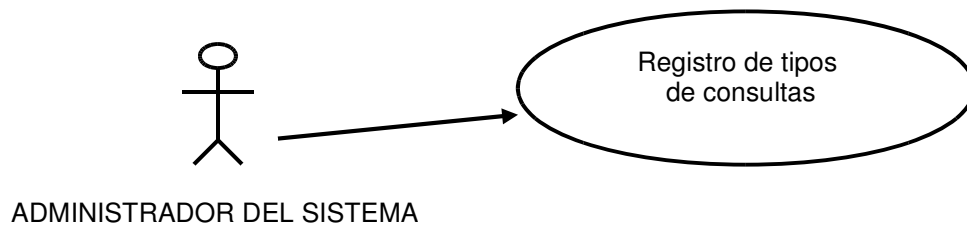
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU17	Fecha de Versión: 27/092001
Identificación: Registro de tipos de consultas brindadas en FUNTER.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a los diversos tipos de consultas que se prestan en la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Se Identifican los tipos de consultas a ser registrados en el sistema.
- ◆ Se registran en el sistema los diversos tipos consultas que brinda FUNTER.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de consultas que brinda FUNTER registrados en el sistema.

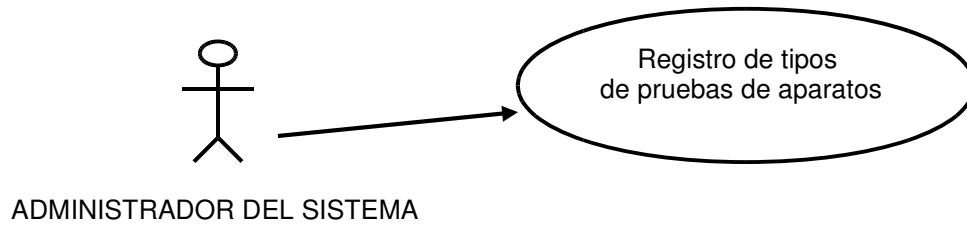
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU18	Fecha de Versión: 27/092001
Identificación: Registro de tipos de pruebas de aparatos.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a los diversos tipos de pruebas que se realizan a los aparatos ortopédicos o protésicos elaborados o reparados en la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Se identifica del tipo de prueba de aparatos a ser registrado en el sistema.
- ◆ Se registra en el sistema los diversos tipos de pruebas de los aparatos fabricados o reparados en FUNTER.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de pruebas de aparatos registrados en el sistema.

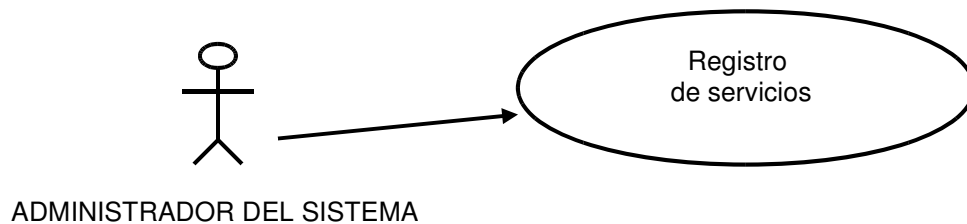
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU19	Fecha de Versión: 27/092001
Identificación: Registro de los servicios brindados en FUNTER.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los datos concernientes a los diversos servicios que se prestan en la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Se identifican los servicios a ser registrado en el sistema.
- ◆ Se registra en el sistema los diversos servicios que se prestan en FUNTER.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Servicios que se brindan en FUNTER registrados en el sistema.

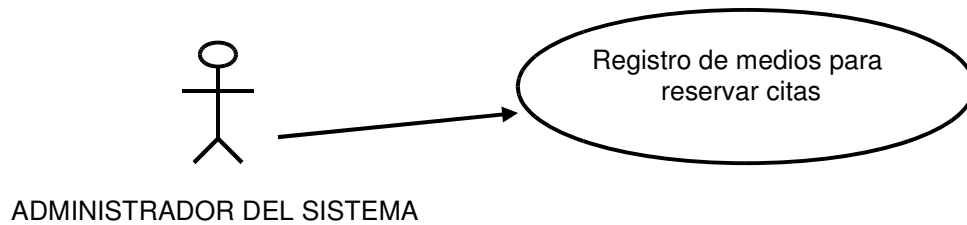
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU20	Fecha de Versión: 27/092001
Identificación: Registro de los medios para reservar citas en FUNTER.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes medios a través de los cuales una persona con discapacidades físicas puede reservar una cita para su atención en la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diversos medios por medio de los cuales se puede reservar una cita.
- ◆ Registrar en el sistema los diversos medios por medio de los cuales se puede reservar una cita en FUNTER.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Medios para reservar citas en FUNTER registrados en el sistema.

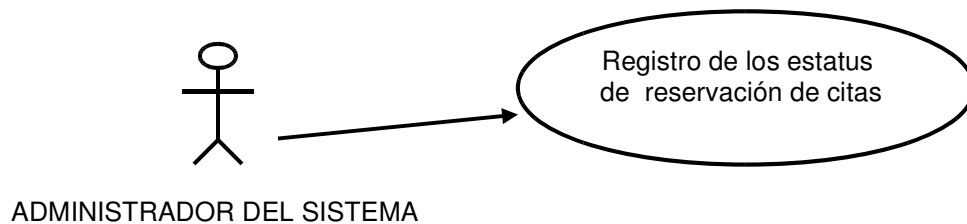
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU21	Fecha de Versión: 27/092001
Identificación: Registro de los estatus de reservación citas en FUNTER.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes estatus que puede tener, en el Sistema, la reservación de una cita solicitada a la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diversos estatus que puede tener la reservación de una cita en el Sistema.
- ◆ Registrar en el sistema los diversos estatus que puede tener la reservación de una cita registrada en este.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Estatus que puede tener la reservación de una cita registrados en el sistema.

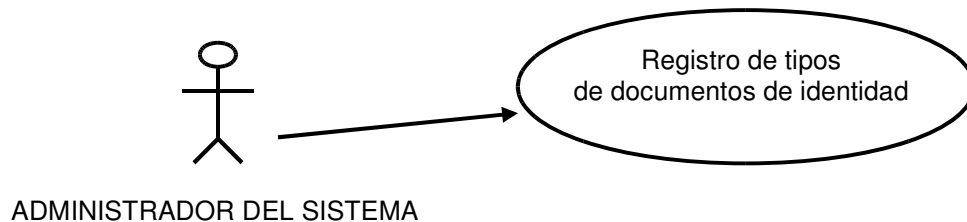
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU22	Fecha de Versión: 27/092001
Identificación: Registro de tipos documentos de identidad personal.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes tipos de documentos que un paciente puede presentar en la Institución para identificarse.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diversos tipos de documentos con los cuales un paciente puede ser identificado en la Institución.
- ◆ Registrar en el sistema los diversos tipos de documentos.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de documentos de identidad personal registrados en el sistema.

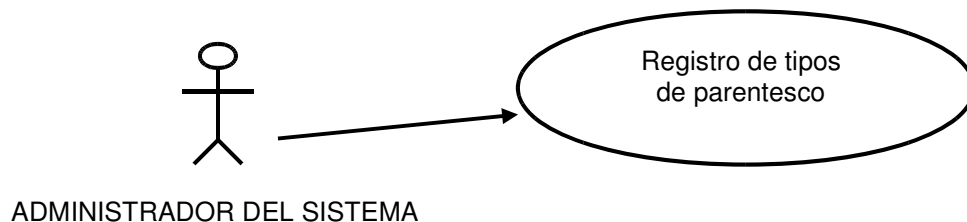
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU23	Fecha de Versión: 27/092001
Identificación: Registro de tipos de parentesco.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes tipos de parentesco que puede tener una persona que sea registrada en el Sistema como familiar de algún paciente de la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diversos tipos de parentesco que puede tener una persona que sea registrada en el Sistema como familiar de algún paciente.
- ◆ Registrar en el sistema los diversos tipos parentescos.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de parentescos registrados en el sistema.

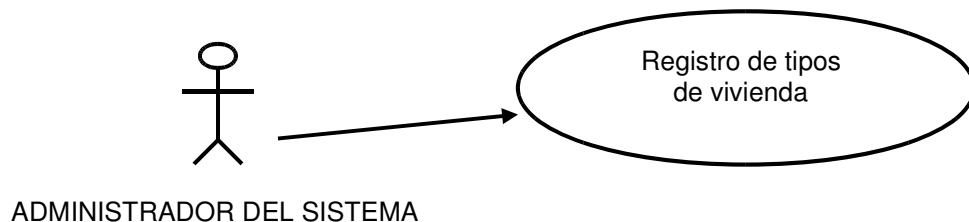
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU24	Fecha de Versión: 27/092001
Identificación: Registro de tipos viviendas.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes tipos de vivienda en los que se puede clasificar la vivienda de un paciente de la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diversos tipos de vivienda en los que puede ser clasificada la vivienda de un paciente de FUNTER.
- ◆ Registrar en el sistema los diversos tipos de vivienda.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de vivienda registrados en el sistema.

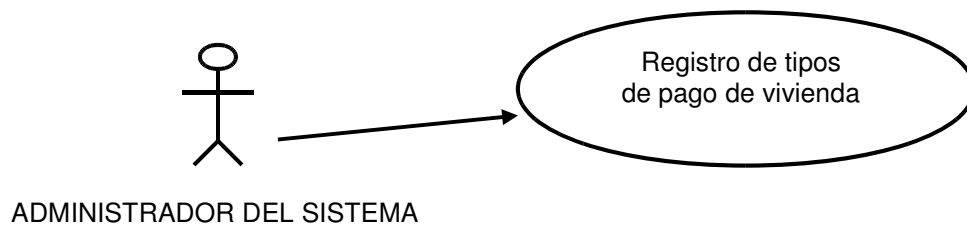
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU25	Fecha de Versión: 27/092001
Identificación: Registro de tipos de pago de vivienda.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de las diferentes formas en que un paciente de FUNTER paga su vivienda.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de las diferentes formas en que un paciente de paga su vivienda.
- ◆ Registrar en el sistema los diversos tipos de pago de vivienda.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de pago de vivienda registrados en el sistema.

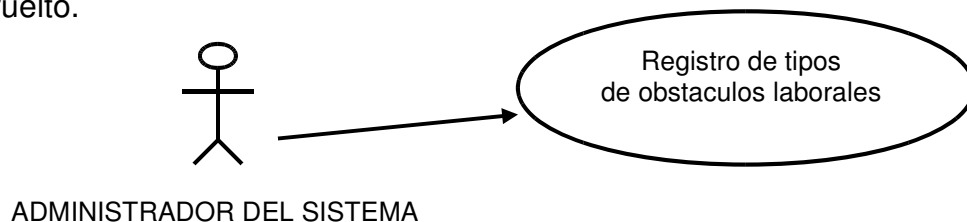
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU26	Fecha de Versión: 27/092001
Identificación: Registro de tipos de obstáculos laborales.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes tipos de obstáculos laborales que un paciente de FUNTER haya podido tener en los diferentes ambientes de trabajo en los que se haya desenvuelto.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diferentes tipos de obstáculos laborales que un paciente de FUNTER pueda tener.
- ◆ Registrar en el sistema los diversos tipos de obstáculos laborales.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de obstáculos laborales, que pueda tener un paciente de FUNTER, registrados en el sistema.

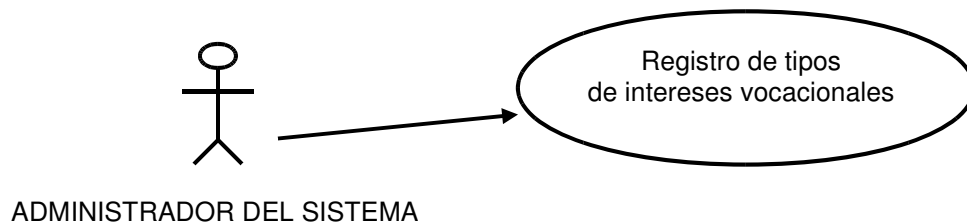
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU27	Fecha de Versión: 27/092001
Identificación: Registro de tipos de intereses vocacionales.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes tipos de intereses vocacionales que pueda tener un paciente de la Institución.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diversos tipos de intereses vocacionales que podría tener un paciente de la Institución.
- ◆ Registrar en el sistema los diversos tipos de intereses vocacionales.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de intereses vocacionales que podría tener un paciente registrados en el sistema.

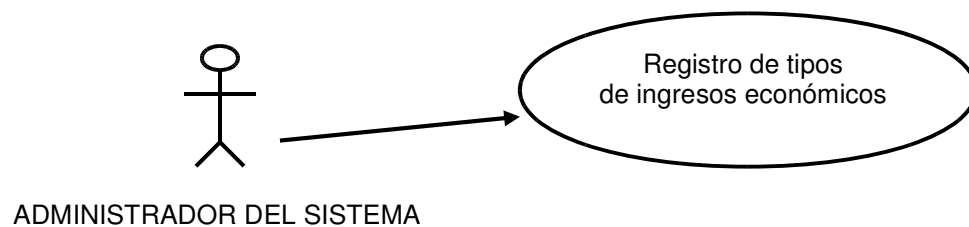
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU28	Fecha de Versión: 27/092001
Identificación: Registro de tipos de ingresos económicos.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes tipos de ingresos económicos que un paciente de la Institución podría tener.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diferentes tipos de ingresos económicos que un paciente de la Institución podría tener.
- ◆ Registrar en el sistema los diversos tipos de ingresos económicos.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

- ◆ Tipos de ingresos económicos de los pacientes registrados en el sistema.

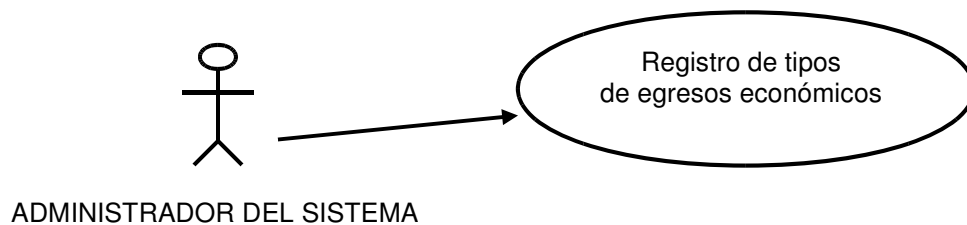
Proyecto: SIAPD – FUNTER	Versión: 1.0
Especificación: CU29	Fecha de Versión: 27/092001
Identificación: Registro de tipos de egresos económicos.	

Histórico de Revisión:

Fecha	Versión	Descripción	Autor
30/10/2001	1.0	Creación del caso de uso	JJRA/BAMD/GARM

Descripción:

Registro de los diferentes tipos de egresos económicos que un paciente de la Institución podría tener.



Requisitos Previos:

- ◆ Ninguno.

Flujo Normal:

- ◆ Identificación de los diferentes tipos de egresos económicos que un paciente de la Institución podría tener.
- ◆ Registrar en el sistema los diversos tipos de egresos económicos.

Flujo Alternativo

- ◆ Ninguno.

Resultados Esperados

Tipos de egresos económicos de los pacientes registrados en el sistema.

ANEXO IV: Detalle de las tablas de la Base de Datos.

1. TABLA ANTECEDENTE DE SALUD.

Nombre:	Antecedente de Salud
Código:	SIAPDU_ANTE_SALU
Descripción:	Antecedentes de Salud de un Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del antecedente de salud	CODI_ANTE_SALU	entero	Si	Si
Descripción de la discapacidad o limitación	DESC_DISC_LIMI	varchar(250)	No	Si
Causa de la discapacidad o limitación	CAUS_DISC_LIMI	varchar(250)	No	No
Fecha de la causa de la discapacidad	FECH_CAUS_DISC	datetime	No	No
Tratamiento médico recibido	TRAT_MEDI_RECIB	varchar(250)	No	No
Lugar donde se hizo la amputación	LUGA_AMP	varchar(250)	No	No
Movilización del usuario	MOVI_USUA	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ANTE_SALU	Si	No	Si	No	CODI_PACI	ASC
U					CODI_ANTE_SALU	ASC
FK_PACI_ANTE_SALU	No	Si	No	No	CODI_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI

2. TABLA APARATO DEL PACIENTE.

Nombre:	Aparato del Paciente
Código:	SIAPDU_APAR_PACI
Descripción:	Aparatos a ser fabricados para un Paciente, como prescripción de una consulta.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Correlativo del aparato	CORR_APAR_PACI	entero	Si	Si
Código del tipo de aparato	CODI_TIPO_APAR	entero	No	Si
Código del estatus del aparato	CODI_ESTA_APAR	entero	No	Si
Fecha de toma de medida	FECH_TOMA_MEDI	datetime	No	Si
Medidas tomadas	MEDI_TOMA	entero	No	Si
Observaciones de la toma de medidas	OBSE_TOMA_MEDI	varchar(250)	No	No
Fecha de entrega programada	FECH_ENTR_PROG	datetime	No	No
Fecha de entrega real	FECH_ENTR_REAL	datetime	No	No
Observaciones de la entrega	OBSE_ENTR	varchar(250)	No	No

Verificaciones de Columnas (Check):

Columna:	MEDI_TOMA		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	0		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	0	Medidas no tomadas	
	1	Medidas tomadas	

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_APAR_PACI	Si	No	Si	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
					CORR_APAR_PACI	ASC
FK_TIPO_APAR	No	Si	No	No	CODI_TIPO_APAR	ASC
FK_CONSI_APAR	No	Si	No	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
FK_ESTA_APAR	No	Si	No	No	CORR_CITA	ASC
					CODI_ESTA_APAR	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_CONS	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
SIAPDU_ESTA_APAR	CODI_ESTA_APAR	CODI_ESTA_APAR
SIAPDU_TIPO_APAR	CODI_TIPO_APAR	CODI_TIPO_APAR

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_MEDI_APAR	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_PRUE_APAR	CORR_APAR_PACI CODI_PACI CODI_TIPO_SERV CORR_CITA CORR_APAR_PACI	CORR_APAR_PACI CODI_PACI CODI_TIPO_SERV CORR_CITA CORR_APAR_PACI

3. TABLA ASPECTO ACADÉMICO.

Nombre:	Aspecto Académico
Código:	SIAPDU_ASPE_ACAD
Descripción:	Aspectos Académicos del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del aspecto académico	CODI_ASPE_ACAD	entero	Si	Si
Escolaridad alcanzada	ESCO_ALCA	varchar(250)	No	Si
Fecha de la escolaridad	FECH_ESCO	datetime	No	Si
Nombre del centro educativo	NOMB_CENT_EDUC	varchar(250)	No	Si
Dirección del centro educativo	DIRE_CENT_EDUC	varchar(250)	No	No
Observaciones del aspecto académico	OBSE_ASPE_ACAD	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ASPE_ACA	Si	No	Si	No	CODI_PACI	ASC
D					CODI_ASPE_ACAD	ASC
FK_PACI_ACAD	No	Si	No	No	CODI_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI

4. TABLA ASPECTO LABORAL.

Nombre:	Aspecto Laboral
Código:	SIAPDU_ASPE_LABO
Descripción:	Aspectos Laborales del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del aspecto laboral	CODI_ASPE_LABO	entero	Si	Si
Lugar de trabajo	LUGA_TRAB	varchar(250)	No	Si
Dirección del trabajo	DIRE_TRAB	varchar(250)	No	No
Puesto ocupado	PUES_OCUP	varchar(250)	No	Si
Salario mensual del paciente	SALA_MENS_PACI	money	No	Si
Descripción de las tareas realizadas	DESC_TARE_REAL	varchar(250)	No	Si
Duración del trabajo en meses	DURA_TRAB_MES	entero	No	No
Motivo del retiro	MOTI_RETI	varchar(250)	No	No
Es trabajo actual	ES_TRAB_ACTU	entero	No	Si
Observaciones del aspecto laboral	OBSE_ASPE_LABO	varchar(250)	No	No

Verificaciones de Columnas (Check):

Columna:	ES_TRAB_ACTU
Valor Mínimo:	
Valor Máximo:	

Valor Default:	1			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	0	No		
	1	Si		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ASPE_LAB	Si	No	Si	No	CODI_PACI	ASC
O					CODI_ASPE_LABO	ASC
FK_PACI_LABO	No	Si	No	No	CODI_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI

5. TABLA AUDITORIA DE TABLAS.

Nombre:	Auditoria de Tablas
Código:	SIAPDU_AUDI_TABL
Descripción:	Auditoria de las diversas acciones ejecutadas sobre una Tabla.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la tabla	CODI_TABL	varchar(100)	Si	Si
Correlativo de la acción	CORR_ACCI	entero	Si	Si
Tipo de acción	TIPO_ACCI	char(1)	No	Si
Llave del registro	LLAV_REGI	varchar(250)	No	Si
Descriptor del registro	DESC_REGI	varchar(250)	No	Si

Nombre	Código	Tipo	P	O
Usuario que actualiza	USUA_ACTU	char(20)	No	Si
Fecha de actualización	FECH_ACTU	datetime	No	Si

Verificaciones de Columnas (Check):

Columna:	TIPO_ACCI			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	I			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	I	Insert		
	D	Delete		
	U	Update		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_AUDI_TABL	Si	No	Si	No	CODI_TABL	ASC
					CORR_ACCI	ASC

6. TABLA CITA.

Nombre:	Cita
Código:	SIAPDU_CITA
Descripción:	Citas de un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	M
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Fecha y hora de la cita	FECH_HORA_CITA	datetime	No	Si
Observaciones de la cita	OBSE_CITA	varchar(250)	No	No
Estatus de la cita	ESTA_CITA	entero	No	Si

Verificaciones de Columnas (Check):

Columna:	ESTA_CITA
----------	-----------

Valor Mínimo:			
Valor Máximo:			
Valor Default:	1		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	0	Anulada	
	1	Confirmada	
	2	Realizada	

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CITA	Si	No	Si	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
FK_EXPE_CITA	No	Si	No	No	CORR_CITA	ASC
					CODI_PACI	ASC
					CODI_TIPO_SERV	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_EXPE	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_CONS	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA

7. TABLA CLASIFICACIÓN DE RANGO ECONÓMICO.

Nombre:	Clasificación de Rango Económico
Código:	SIAPDU_CLAS_RANG_ECON
Descripción:	Clasificación del Rango Económico que podría tener un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de rango económico	CODI_RANG_ECON	entero	Si	Si
Límite económico inicial	LIMI_ECON_INIC	money	No	Si
Límite económico final	LIMI_ECON_FINI	money	No	Si
Observaciones del rango económico	OBSE_RANG_ECON	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CLAS_RAN G_ECON	Si	No	Si	No	CODI_RANG_ECON	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_ESTU_SOCIO	CODI_RANG_ECON	CODI_RANG_ECON

8. TABLA CLÍNICA.

Nombre:	Clínica
Código:	SIAPDU_CLIN
Descripción:	Clínicas de FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la clínica	CODI_CLIN	entero	Si	Si
Código del país	CODI_PAIS	entero	No	Si
Código del departamento	CODI_DEPA	entero	No	Si
Código del municipio	CODI_MUNI	entero	No	Si
Nombre de la clínica	NOMB_CLIN	varchar(250)	No	Si
Dirección de la clínica	DIRE_CLIN	varchar(250)	No	Si
Teléfono de la clínica	TELE_CLIN	char(7)	No	Si
Fax de la clínica	FAX_CLIN	char(7)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CLIN	Si	No	Si	No	CODI_CLIN	ASC
FK_MUNI_CLIN	No	Si	No	No	CODI_PAIS	ASC
					CODI_DEPA	ASC
					CODI_MUNI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_MUNI	CODI_PAIS	CODI_PAIS
	CODI_DEPA	CODI_DEPA

Referencia a	Llave Primaria	Llave Foránea
	CODI_MUNI	CODI_MUNI

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_TIPO_SERV_CLIN	CODI_CLIN	CODI_CLIN
SIAPDU_HORA_CLIN	CODI_CLIN	CODI_CLIN

9. TABLA CONSULTA.

Nombre:	Consulta
Código:	SIAPDU_CONS
Descripción:	Consulta de un Paciente (Medica o PONI).

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Código del empleado	CODI_EMPL	entero	No	Si
Procedencia del caso del paciente	PROC_CASO_PACI	varchar(250)	No	No
Síntomas del paciente	SENTERO_PACI	varchar(250)	No	Si
Diagnóstico	DIAG_CONS	varchar(250)	No	Si
Prescripción	PRES_CONS	char(1)	No	Si
Receta	RESE_CONS	varchar(250)	No	No
Observaciones de la consulta	OBSE_CONS	varchar(250)	No	No
Fecha y hora de próxima consulta	FECH_HORA_PROX_CON	datetime	No	No
	S			

Verificaciones de Columnas (Check):

Columna:	PRES_CONS		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	T		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	A	Aparato	

T	Terapia
R	Referencia a otra Institución

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CONS	Si	Si	Si	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
FK_EMPL_CONS	No	Si	No	No	CODI_EMPL	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_CITA	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
SIAPDU_EMPL	CODI_EMPL	CODI_EMPL

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_APAR_PACI	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
SIAPDU_REFE_OTRA_INS	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
T SIAPDU_TERA_PACI	CORR_CITA	CORR_CITA
	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA

10. TABLA CONTEXTO COMUNITARIO.

Nombre:	Contexto Comunitario
Código:	SIAPDU_CONT_COMU
Descripción:	Contexto de la Comunidad en la que vive el paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del contexto comunitario	CODI_CONT_COMU	entero	Si	Si
Características de la comunidad	CARA_COMU	varchar(250)	No	No
Recursos humanos para rehabilitar	RRHH_REHA	varchar(250)	No	No
Otras personas con discapacidad	OTRA_PERS_DISCA	varchar(250)	No	No
Actividades familiares en la comunidad	ACTI_FAMI_COMU	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CONT_COMU	Si	No	Si	No	CODI_PACI	ASC
U					CODI_CONT_COMU	ASC
FK_PACI_CONT_COMU	No	Si	No	No	CODI_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI

11. TABLA DE CONTROL DE ASISTENCIA DE TERAPIA.

Nombre:	Control de Asistencia de Terapia
Código:	SIAPDU_CALE_TERA
Descripción:	Control de Asistencia de las diversas Terapias de un Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Correlativo de la terapia	CORR_TERA_PACI	entero	Si	Si
Fecha y hora de la terapia	FECH_HORA_TERA	datetime	No	Si

Nombre	Código	Tipo	P	O
Tipo de atención	TIPO_ATEN	char(1)	No	Si
Diagnóstico de la terapia	DIAG_TERA	varchar(250)	No	No
Observaciones de la terapia	OBSE_TERA	varchar(250)	No	No
Estatus de la terapia	ESTA_TERA	char(1)	No	Si

Verificaciones de Columnas (Check):

Columna:	TIPO_ATEN			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	E			
Letras Mayúsculas:	Si	Letras Minúsculas:	No	
Lista de Valores:	E	Entrevista		
	S	Subsiguiente		

Columna:	ESTA_TERA			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	A			
Letras Mayúsculas:	Si	Letras Minúsculas:	No	
Lista de Valores:	A	Asistió		
	P	Permiso		
	I	Inasistencia		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CALE_TER	Si	Si	Si	No	CODI_PACI	ASC
A					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
					CORR_TERA_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_TERA_PACI	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
	CORR_TERA_PACI	CORR_TERA_PACI

12. TABLA CONTROL DE PAGOS.

Nombre:	Control de Pagos
Código:	SIAPDU_CONT_PAGO
Descripción:	Control de Pago de los diversos servicios recibidos por un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Correlativo de pago	CORR_PAGO	entero	Si	Si
Código del servicio a pagar	CODI_SERV_PAGO	entero	No	Si
Prima de pago	PRIMA_PAGO	money	No	Si
Abono de pago	ABON_PAGO	money	No	Si
Forma de pago	FORM_PAGO	entero	No	Si
Cuotas programadas a pagar	CUOT_PROG_PAGO	entero	No	No
Cuotas pagadas	CUOT_REAL_PAGO	entero	No	No
Estatus de pago	ESTA_PAGO	char(1)	No	Si

Verificaciones de Columnas (Check):

Columna:	PRIMA_PAGO		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	0.00		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:			

Columna:	ABON_PAGO		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	0.00		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:			

Columna:	FORM_PAGO		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	1		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	1	Contado	
	2	Crédito	

Columna:	ESTA_PAGO		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	C		
Letras Mayúsculas:	Si	Letras Minúsculas:	No
Lista de Valores:	C	Cancelado	
	P	Pendiente	

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CONT_PAG	Si	No	Si	No	CODI_PACI	ASC
O					CORR_PAGO	ASC
FK_PACI_PAGO	No	Si	No	No	CODI_PACI	ASC
FK_PAGO_SERV	No	Si	No	No	CODI_SERV_PAGO	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI
SIAPDU_SERV_PAGO	CODI_SERV_PAGO	CODI_SERV_PAGO

13. TABLA CUESTIONARIO PARA EVALUACIÓN.

Nombre:	Cuestionario para Evaluación
Código:	SIAPDU_CUES_EVAL
Descripción:	Cuestionario para la Evaluación del Paciente al que se le efectuarán las Terapias.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de terapia	CODI_TIPO_TERA	entero	Si	Si
Código de la pregunta	CODI_PREG	entero	Si	Si
Pregunta	PREG	varchar(250)	No	Si
Estatus de la pregunta	ESTA_PREG	entero	No	Si

Verificaciones de Columnas (Check):

Columna:	ESTA_PREG		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	1		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	1	Vigente	
	0	No Vigente	

Lista de Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_CUES_EVA	Si	No	Si	No	CODI_TIPO_TERA	ASC
L					CODI_PREG	ASC
FK_TIPO_TERA_CUES	No	Si	No	No	CODI_TIPO_TERA	ASC

Lista de Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_TIPO_TERA	CODI_TIPO_TERA	CODI_TIPO_TERA

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_EVAL_PACI	CODI_TIPO_TERA	CODI_TIPO_TERA
	CODI_PREG	CODI_PREG

14. TABLA DEPARTAMENTO.

Nombre:	Departamento
Código:	SIAPDU_DEPA
Descripción:	Departamentos de los Países en los que se puede utilizar el SIAPD.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del país	CODI_PAIS	entero	Si	Si
Código del departamento	CODI_DEPA	entero	Si	Si
Nombre del departamento	NOMB_DEPA	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_DEPA	Si	No	Si	No	CODI_PAIS	ASC
FK_PAIS_DEPA	No	Si	No	No	CODI_DEPA	ASC
					CODI_PAIS	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PAIS	CODI_PAIS	CODI_PAIS

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_MUNI	CODI_PAIS	CODI_PAIS
	CODI_DEPA	CODI_DEPA

15. TABLA DINÁMICA FAMILIAR.

Nombre:	Dinámica Familiar
Código:	SIAPDU_DINA_FAMI
Descripción:	Dinámica Familiar del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código de la dinámica familiar	CODI_DINA_FAMI	entero	Si	Si
Relaciones enteroerpersonales	RELA_ENTEROE	varchar(250)	No	No
Toma de decisiones	TOMA_DECI	varchar(250)	No	No
Actitudes de la familia ante limitación	ACTI_FAMI_ANTE_LIMI	varchar(250)	No	No
Es dinámica actual	ES_DINA_ACTU	entero	No	Si

Verificaciones de Columnas (Check):

Columna:	ES_DINA_ACTU			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	0			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	0		No	
	1		Si	

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_DINA_FAMI	Si	No	Si	No	CODI_PACI	ASC
					CODI_DINA_FAMI	ASC
FK_PACI_DINA_FAMI	No	Si	No	No	CODI_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI

16. TABLA DOCUMENTO DEL PACIENTE.

Nombre:	Documento del Paciente
Código:	SIAPDU_DOCU_PACI
Descripción:	Documentos del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de documento	CODI_TIPO_DOCU	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Número del documento	NUME_DOCU	varchar(50)	No	Si
Lugar de expedición	LUGA_EXPE	varchar(100)	No	Si
Fecha de expedición	FECH_EXPE	datetime	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_DOCU_PAC	Si	No	Si	No	CODI_TIPO_DOCU	ASC
I					CODI_PACI	ASC
FK_PACI_DOCU	No	Si	No	No	CODI_PACI	ASC
FK_TIPO_DOCU	No	Si	No	No	CODI_TIPO_DOCU	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI
SIAPDU_TIPO_DOCU	CODI_TIPO_DOCU	CODI_TIPO_DOCU

17. TABLA EGRESO DEL PACIENTE.

Nombre:	Egreso del Paciente
Código:	SIAPDU_EGRE_PACI
Descripción:	Egresos económicos del paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de egreso	CODI_TIPO_EGRE	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Cantidad de egreso	CANT_EGRE	money	No	Si
Observaciones del egreso	OBSE_EGRE	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_EGRE_PACI	Si	No	Si	No	CODI_TIPO_EGRE	ASC
					CODI_PACI	ASC
FK_TIPO_EGRE	No	Si	No	No	CODI_TIPO_EGRE	ASC
FK_PACI_EGRE	No	Si	No	No	CODI_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI
SIAPDU_TIPO_EGRE	CODI_TIPO_EGRE	CODI_TIPO_EGRE

18. TABLA EMPLEADO.

Nombre:	Empleado
Código:	SIAPDU_EMPL
Descripción:	Empleados de FUNTER relacionados con el SIAPD.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del empleado	CODI_EMPL	entero	Si	Si
Código del puesto de trabajo	CODI_PUES_TRAB	entero	No	Si
Código de la clínica	CODI_CLIN	entero	No	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	No	Si
Nombres del empleado	NOMB_EMPL	varchar(100)	No	Si
Apellidos del empleado	APEL_EMPL	varchar(100)	No	Si
Código de usuario del siapd	CODI_USUA	char(20)	No	No
Sexo del empleado	SEXO_EMPL	char(1)	No	Si
Estado civil del empleado	ESTA_CIVI_EMPL	char(1)	No	Si
Dirección de residencia del empleado	DIRE_RESI_EMPL	varchar(250)	No	Si
Teléfono de residencia del empleado	TELE_RESI_EMPL	char(7)	No	No
Teléfono móvil del empleado	TELE_MOVI_EMPL	char(7)	No	No
Estatus del empleado	ESTA_EMPL	char(1)	No	Si

Verificaciones de Columnas (Check):

Columna:	SEXO_EMPL		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	S		
Letras Mayúsculas:	Si	Letras Minúsculas:	No
Lista de Valores:	F	Femenino	
	M	Masculino	

Columna:	ESTA_CIVI_EMPL		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	S		

Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	A	Acompañado	
	C	Casado	
	D	Divorciado	
	S	Soltero	
	V	Viudo	

Columna:	ESTA_EMPL		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	A		
Letras Mayúsculas:	Si	Letras Minúsculas:	No
Lista de Valores:	A	Activo	
	D	Despedido	
	R	Renuncio	

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_EMPL	Si	No	Si	No	CODI_EMPL	ASC
FK_PUES_TRAB_EMPL	No	Si	No	No	CODI_PUES_TRAB	ASC
FK_TIPO_SERV_CLIN_E	No	Si	No	No	CODI_CLIN	ASC
MPL					CODI_TIPO_SERV	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PUES_TRAB	CODI_PUES_TRAB	CODI_PUES_TRAB
SIAPDU_TIPO_SERV_CLIN	CODI_CLIN	CODI_CLIN
	CODI_TIPO_SERV	CODI_TIPO_SERV

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_CONS	CODI_EMPL	CODI_EMPL

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_ESPE_EMPL	CODI_EMPL	CODI_EMPL
SIAPDU_RESE_CITA	CODI_EMPL	CODI_EMPL
SIAPDU_TERA_PACI	CODI_EMPL	CODI_EMPL
SIAPDU_HORA_EMPL	CODI_EMPL	CODI_EMPL

19. TABLA ESPECIALIDAD.

Nombre:	Especialidad
Código:	SIAPDU_ESPE
Descripción:	Especialidades que podría tener un empleado de FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la especialidad	CODI_ESPE	entero	Si	Si
Nombre de la especialidad	NOMB_ESPE	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ESPE	Si	No	Si	No	CODI_ESPE	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_ESPE_EMPL	CODI_ESPE	CODI_ESPE

20. TABLA ESPECIALIDAD DE UN EMPLEADO.

Nombre:	Especialidad de un Empleado
Código:	SIAPDU_ESPE_EMPL
Descripción:	Registro de las Especialidades de un empleado específico de FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la especialidad	CODI_ESPE	entero	Si	Si
Código del empleado	CODI_EMPL	entero	Si	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ESPE_EMP	Si	No	Si	No	CODI_ESPE	ASC
L					CODI_EMPL	ASC
FK_EMPL_ESPE_EMPL	No	Si	No	No	CODI_EMPL	ASC
FK_ESPE_ESPE_EMPL	No	Si	No	No	CODI_ESPE	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_EMPL	CODI_EMPL	CODI_EMPL
SIAPDU_ESPE	CODI_ESPE	CODI_ESPE

21. TABLA ESTATUS DE LA RESERVACIÓN DE CITA.

Nombre:	Estatus de la Reservación de Cita
Código:	SIAPDU_ESTA_RESE_CITA
Descripción:	Estatus de las Reservaciones de Citas para Consultas (Médicas o PONI).

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del estatus de reservación de cita	CODI_ESTA_RESE_CITA	entero	Si	Si
Nombre del estatus de reservación de cita	NOMB_ESTA_RESE_CITA	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ESTA_RES E_CITA	Si	No	Si	No	CODI_ESTA_RESE_CITA	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_RESE_CITA	CODI_ESTA_RESE_CITA	CODI_ESTA_RESE_CITA

22. TABLA ESTATUS DEL APARATO.

Nombre:	Estatus del Aparato
Código:	SIAPDU_ESTA_APAR
Descripción:	Estatus del proceso de elaboracion o reparacion de Aparatos.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del estatus del aparato	CODI_ESTA_APAR	entero	Si	Si
Nombre del estatus del aparato	NOMB_ESTA_APAR	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ESTA_APA R	Si	No	Si	No	CODI_ESTA_APAR	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_APAR_PACI	CODI_ESTA_APAR	CODI_ESTA_APAR

23. TABLA ESTUDIO SOCIOECONÓMICO.

Nombre:	Estudio Socioeconómico
Código:	SIAPDU_ESTU_SOCIO
Descripción:	Estudio Socioeconómico del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de rango económico	CODI_RANG_ECON	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_ESTU_SOCIO	Si	No	Si	No	CODI_RANG_ECON	ASC
					CODI_PACI	ASC
FK_PACI_ESTU_SOCIO	No	Si	No	No	CODI_PACI	ASC

Código del Índice	P	F	U	C	Columnas	Orden
FK_RANG_ECON_ESTU	No	Si	No	No	CODI_RANG_ECON	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI SIAPDU_CLAS_RANG_EC ON	CODI_PACI CODI_RANG_ECON	CODI_PACI CODI_RANG_ECON

24. TABLA EVALUACIÓN DE PACIENTE.

Nombre:	Evaluación de Paciente
Código:	SIAPDU_EVAL_PACI
Descripción:	Evaluación de Paciente al que se le efectuarán las Terapias.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de terapia	CODI_TIPO_TERA	entero	Si	Si
Código de la pregunta	CODI_PREG	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Correlativo de la terapia	CORR_TERA_PACI	entero	Si	Si
Respuesta	RESP	varchar(250)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_EVAL_PACI	Si	No	Si	No	CODI_TIPO_TERA	ASC
					CODI_PREG	ASC
					CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
FK_CUEST_EVAL	No	Si	No	No	CORR_TERA_PACI	ASC
					CODI_TIPO_TERA	ASC
					CODI_PREG	ASC

Código del Índice	P	F	U	C	Columnas	Orden
FK_TERA_EVAL	No	Si	No	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
					CORR_TERA_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_CUES_EVAL	CODI_TIPO_TERA	CODI_TIPO_TERA
SIAPDU_TERA_PACI	CODI_PREG	CODI_PREG
	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
	CORR_TERA_PACI	CORR_TERA_PACI

25. TABLA EXPEDIENTE.

Nombre:	Expediente
Código:	SIAPDU_EXPE
Descripción:	Expedientes que puede tener un Paciente, según los servicios recibidos.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Número de expediente	NUME_EXPE	entero	No	Si
Fecha de apertura de expediente	FECH_APER_EXPE	datetime	No	Si
Estatus del expediente	ESTA_EXPE	entero	No	Si

Verificaciones de Columnas (Check):

Columna:	FECH_APER_EXPE			
Valor Mínimo:				
Valor Máximo:	getdate()			
Valor Default:	0.00			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:				

Columna:	ESTA_EXPE			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	1			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	1	Activo		
	0	Incativo		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_EXPE	Si	No	Si	No	CODI_PACI	ASC
FK_TIPO_SERV_EXPE	No	Si	No	No	CODI_TIPO_SERV	ASC
FK_PACI_EXPE	No	Si	No	No	CODI_PACI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI
SIAPDU_TIPO_SERV	CODI_TIPO_SERV	CODI_TIPO_SERV

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_CITA	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV

26. TABLA FAMILIAR DEL PACIENTE.

Nombre:	Familiar del Paciente
Código:	SIAPDU_FAMI_PACI
Descripción:	Familiares del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del familiar	CODI_FAMI	entero	Si	Si
Código del tipo de Tabla	CODI_TIPO_PARE	entero	No	Si
Padreesco				
Nombres del familiar	NOV_FAMI	varchar(100)	No	Si
Apellidos del familiar	APEL_FAMI	varchar(100)	No	Si
Es responsable del paciente	ES_RESP_PACI	entero	No	Si
Sexo del familiar	SEXO_FAMI	char(1)	No	Si
Estado civil del familiar	ESTA_CIVI_FAMI	char(1)	No	Si
Fecha de nacimiento del familiar	FECH_NACI_FAMI	datetime	No	Si
Dirección del familiar	DIRE_FAMI	varchar(250)	No	Si
Teléfono residencia del familiar	TELE_RESI_FAMI	char(7)	No	No
Escolaridad del familiar	ESCO_FAMI	varchar(250)	No	No
Ocupación del familiar	OCUP_FAMI	varchar(250)	No	No

Nombre	Código	Tipo	P	O
Salario mensual del familiar	SALA_MENS_FAMI	money	No	No

Verificaciones de Columnas (Check):

Columna:	ES_RESP_PACI			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	1			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	1	Si		
	0	No		

Columna:	SEXO_FAMI			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	M			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	M	Masculino		
	F	Femenino		

Columna:	ES_RESP_PACI			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	1			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	1	Si		
	0	No		

Columna:	ESTA_CIVI_FAMI			
Valor Mínimo:				

Valor Máximo:			
Valor Default:	S		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	A	Acompañado	
	C	Casado	
	D	Divorciado	
	S	Soltero	
	V	Viudo	

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_FAMI_PACI	Si	No	Si	No	CODI_PACI	ASC
					CODI_FAMI	ASC
FK_PACI_FAMI	No	Si	No	No	CODI_PACI	ASC
FK_TIPO_PARE	No	Si	No	No	CODI_TIPO_PARE	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI
SIAPDU_TIPO_PARE	CODI_TIPO_PARE	CODI_TIPO_PARE

27. TABLA HORARIOS DE CLÍNICAS.

Nombre:	Horarios de Clínicas
Código:	SIAPDU_HORA_CLIN
Descripción:	Horarios de Atención y Servicio a Pacientes en una Clínica.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la clínica	CODI_CLIN	entero	Si	Si
Día horario	DIA_HORA	entero	Si	Si
Hora entrada	HORA_ENTR	datetime	No	No
Hora salida	HORA_SALI	datetime	No	No
Hora inicio receso	HORA_INIC	datetime	No	No
Hora fin receso	HORA_FINA	datetime	No	No
Duración de consulta	DURA_CONS	entero	No	No
Duración de terapia	DURA_TERA	entero	No	No
Observaciones horario clínica	OBSE_HORA	varchar(250)	No	No

Nombre	Código	Tipo	P	O
Estatus horario clínica	ESTA_HORA	entero	No	No

Verificaciones de Columnas (Check):

Columna:	DIA_HORA			
Valor Mínimo:	1			
Valor Máximo:	7			
Valor Default:	1			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	1	Lunes		
	2	Martes		
	3	Miércoles		
	4	Jueves		
	5	Viernes		
	6	Sábado		
	7	Domingo		

Columna:	ESTA_HORA			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	1			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	1	Activo		
	0	Inactivo		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_HORA_CLIN	Si	No	Si	No	CODI_CLIN	ASC
FK_HORA_CLIN	No	Si	No	No	DIA_HORA CODI_CLIN	ASC ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_CLIN	CODI_CLIN	CODI_CLIN

28. TABLA HORARIOS DE LOS EMPLEADOS.

Nombre:	Horarios de los Empleados
Código:	SIAPDU_HORA_EMPL
Descripción:	Horarios de Trabajo de un Empleado de FUNTER relacionados con el SIAPD.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del empleado	CODI_EMPL	entero	Si	Si
Día horario del empleado	DIA_HORA_EMPL	entero	Si	Si
Hora entrada del empleado	HORA_ENTR_EMPL	datetime	No	No
Hora salida del empleado	HORA_SALI_EMPL	datetime	No	No
Hora inicio receso del empleado	HORA_INIC_RECE	datetime	No	No
Hora fin receso del empleado	HORA_FINA_RECE	datetime	No	No
Observaciones horario empleado	OBSE_HORA_EMPL	varchar(250)	No	No
Estatus horario empleado	ESTA_HORA_EMPL	entero	No	No

Verificaciones de Columnas (Check):

Columna:	DIA_HORA_EMPL		
Valor Mínimo:	1		
Valor Máximo:	7		
Valor Default:	1		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	1	Lunes	
	2	Martes	
	3	Miércoles	
	4	Jueves	
	5	Viernes	
	6	Sábado	
	7	Domingo	

Columna:	ESTA_HORA_EMPL
----------	----------------

Valor Mínimo:				
Valor Máximo:				
Valor Default:	1			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	1	Activo		
	0	Inactivo		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_HORA_EMP	Si	No	Si	No	CODI_EMPL	ASC
L					DIA_HORA_EMPL	ASC
FK_HORA_EMPL	No	Si	No	No	CODI_EMPL	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_EMPL	CODI_EMPL	CODI_EMPL

29. TABLA INGRESO DEL PACIENTE.

Nombre:	Ingreso del Paciente
Código:	SIAPDU_INGR_PACI
Descripción:	Ingresos económicos del paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de ingreso	CODI_TIPO_INGR	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Cantidad de ingreso	CANT_INGR	money	No	Si
Observaciones del ingreso	OBSE_INGR	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_INGR_PACI	Si	No	Si	No	CODI_TIPO_INGR	ASC
					CODI_PACI	ASC
FK_PACI_INGR	No	Si	No	No	CODI_PACI	ASC

Código del Índice	P	F	U	C	Columnas	Orden
FK_TIPO_INGR	No	Si	No	No	CODI_TIPO_INGR	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI SIAPDU_TIPO_INGR	CODI_PACI CODI_TIPO_INGR	CODI_PACI CODI_TIPO_INGR

30. TABLA INTERÉS VOCACIONAL DEL PACIENTE.

Nombre:	Interés Vocacional del Paciente
Código:	SIAPDU_INTE_VOCA_PACI
Descripción:	Interés Vocacionales del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de interés vocacional	CODI_TIPO_INTE_VOCA	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Experiencia en el interés vocacional	EXPE_INTE_VOCA	varchar(250)	No	No
Motivación en el interés vocacional	MOTI_INTE_VOCA	varchar(250)	No	No
Opinión de la familia sobre el interés vocacional	OPIN_FAMI_INTE_VOCA	varchar(250)	No	No
Observaciones del interés vocacional	OBSE_INTE_VOCA	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_INTE_VOCA_PACI	Si	No	Si	No	CODI_TIPO_INTE_VOCA CODI_PACI	ASC ASC
FK_PACI_INTE_VOCA	No	Si	No	No	CODI_PACI	ASC
FK_TIPO_INTE_VOCA	No	Si	No	No	CODI_TIPO_INTE_VOCA	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI SIAPDU_TIPO_INTE_VOC A	CODI_PACI CODI_TIPO_INTE_VOCA	CODI_PACI CODI_TIPO_INTE_VOCA

31. TABLA MEDIDAS DEL APARATO.

Nombre:	Medidas del Aparato
Código:	SIAPDU_MEDI_APAR
Descripción:	Medidas del aparato a ser fabricado para un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de aparato	CODI_TIPO_APAR	entero	Si	Si
Código del tipo de medida	CODI_TIPO_MEDI	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Correlativo del aparato	CORR_APAR_PACI	entero	Si	Si
Medida del aparato	MEDI_APAR	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_MEDI_APA R	Si	No	Si	No	CODI_TIPO_APAR CODI_TIPO_MEDI CODI_PACI CODI_TIPO_SERV CORR_CITA	ASC ASC ASC ASC ASC
FK_APAR_MEDI_APAR	No	Si	No	No	CORR_APAR_PACI CODI_PACI CODI_TIPO_SERV CORR_CITA	ASC ASC ASC ASC
FK_TIPO_MEDI_MEDI_A PAR	No	Si	No	No	CORR_APAR_PACI CODI_TIPO_APAR CODI_TIPO_MEDI	ASC ASC ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_APAR_PACI	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
	CORR_APAR_PACI	CORR_APAR_PACI
SIPADU_TIPO_MEDI_APA	CODI_TIPO_APAR	CODI_TIPO_APAR
R	CODI_TIPO_MEDI	CODI_TIPO_MEDI

32. TABLA MEDIO DE RESERVACIÓN DE CITA.

Nombre:	Medio de Reservación de Cita
Código:	SIAPDU_MEDI_RESE_CITA
Descripción:	Medios utilizados para las Reservaciones de Citas para Consultas (Médicas o PONI).

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del medio de reservación	CODI_MEDI_RESE_CITA	entero	Si	Si
Nombre del medio de reservación	NOMB_MEDI_RESE_CITA	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_MEDI_RES E_CITA	Si	No	Si	No	CODI_MEDI_RESE_CITA	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_RESE_CITA	CODI_MEDI_RESE_CITA	CODI_MEDI_RESE_CITA

33. TABLA MUNICIPIO.

Nombre:	Municipio
Código:	SIAPDU_MUNI
Descripción:	Municipios de un departamento de un país específico.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del país	CODI_PAIS	entero	Si	Si
Código del departamento	CODI_DEPA	entero	Si	Si
Código del municipio	CODI_MUNI	entero	Si	Si
Nombre del municipio	NOMB_MUNI	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_MUNI	Si	No	Si	No	CODI_PAIS	ASC
					CODI_DEPA	ASC
FK_DEPA_MUNI	No	Si	No	No	CODI_MUNI	ASC
					CODI_PAIS	ASC
					CODI_DEPA	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_DEPA	CODI_PAIS	CODI_PAIS
	CODI_DEPA	CODI_DEPA

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_CLIN	CODI_PAIS	CODI_PAIS
	CODI_DEPA	CODI_DEPA
	CODI_MUNI	CODI_MUNI

34. TABLA OBSTÁCULO LABORAL DEL PACIENTE.

Nombre:	Obstáculo Laboral del Paciente
Código:	SIAPDU_OBST_LABO_PACI
Descripción:	Obstáculos Laborales del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de obstáculo laboral	CODI_TIPO_OBST_LABO	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Observaciones del obstáculo laboral	OBSE_OBST_LABO	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_OBST_LAB	Si	No	Si	No	CODI_TIPO_OBST_LABO	ASC
O_PACI					CODI_PACI	ASC
FK_PACI_OBST_LABO	No	Si	No	No	CODI_PACI	ASC
FK_TIPO_OBST_LABO	No	Si	No	No	CODI_TIPO_OBST_LABO	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI
SIAPDU_TIPO_OBST_LAB	CODI_TIPO_OBST_LABO	CODI_TIPO_OBST_LABO
O		

35. TABLA OTRA INSTITUCIÓN MEDICA.

Nombre:	Otra Institución Medica
Código:	SIAPDU_OTRA_INST_MEDI
Descripción:	Registro de Otras Instituciones Médicas relacionadas con FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la institución	CODI_INST_MEDI	entero	Si	Si
Nombre de la institución	NOMB_INST_MEDI	varchar(250)	No	Si
Dirección de la institución	DIRE_INST_MEDI	varchar(250)	No	No
Teléfono de la institución	TELE_INST_MEDI	char(7)	No	No
Fax de la institución	FAX_INST_MEDI	char(7)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_OTRA_INST_MEDI	Si	No	Si	No	CODI_INST_MEDI	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_REFE_OTRA_INST	CODI_INST_MEDI	CODI_INST_MEDI

36. TABLA PACIENTE.

Nombre:	Paciente
Código:	SIAPDU_PACI
Descripción:	Pacientes de FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Nombres del paciente	NOMB_PACI	varchar(100)	No	Si
Apellidos del paciente	APEL_PACI	varchar(100)	No	Si
Sexo del paciente	SEXO_PACI	char(1)	No	Si
Estado civil del paciente	ESTA_CIVI_PACI	char(1)	No	Si
Lugar de nacimiento del paciente	LUGA_NACI_PACI	entero	No	Si
Fecha de nacimiento del paciente	FECH_NACI_PACI	datetime	No	Si
Estatus del Paciente	ESTA_PACI	entero	No	Si

Verificaciones de Columnas (Check):

Columna:	SEXO_PACI		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	M		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	M	Masculino	
	F	Femenino	

Columna:	ESTA_CIVI_PACI		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	S		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	A	Acompañado	
	C	Casado	
	D	Divorciado	
	S	Soltero	
	V	Viudo	

Columna:	ESTA_PACI		
Valor Mínimo:			
Valor Máximo:			
Valor Default:	1		
Letras Mayúsculas:	No	Letras Minúsculas:	No
Lista de Valores:	1	Activo	
	0	Incativo	

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_PACI	Si	No	Si	No	CODI_PACI	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_ANTE_SALU	CODI_PACI	CODI_PACI
SIAPDU_ASPE_ACAD	CODI_PACI	CODI_PACI
SIAPDU_ASPE_LABO	CODI_PACI	CODI_PACI
SIAPDU_CONT_COMU	CODI_PACI	CODI_PACI
SIAPDU_DINA_FAMI	CODI_PACI	CODI_PACI
SIAPDU_DOCU_PACI	CODI_PACI	CODI_PACI
SIAPDU_EGRE_PACI	CODI_PACI	CODI_PACI
SIAPDU_ESTU_SOCU	CODI_PACI	CODI_PACI
SIAPDU_EXPE	CODI_PACI	CODI_PACI

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_FAMI_PACI	CODI_PACI	CODI_PACI
SIAPDU_INGR_PACI	CODI_PACI	CODI_PACI
SIAPDU_INTE_VOCA_PAC	CODI_PACI	CODI_PACI
I		
SIAPDU_OBST_LABO_PA	CODI_PACI	CODI_PACI
CI		
SIAPDU_CONT_PAGO	CODI_PACI	CODI_PACI
SIAPDU_VIVI_PACI	CODI_PACI	CODI_PACI

37. TABLA PAÍS.

Nombre:	País
Código:	SIAPDU_PAIS
Descripción:	Países en los que se puede utilizar el SIAPD.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del país	CODI_PAIS	entero	Si	Si
Nombre del país	NOMB_PAIS	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_PAIS	Si	No	Si	No	CODI_PAIS	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_DEPA	CODI_PAIS	CODI_PAIS

38. TABLA PRUEBA DEL APARATO.

Nombre:	Prueba del Aparato
Código:	SIAPDU_PRUE_APAR
Descripción:	Pruebas del aparato que realiza el paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de prueba de aparato	CODI_TIPO_PRUE_APAR	entero	Si	Si
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Correlativo del aparato	CORR_APAR_PACI	entero	Si	Si
Código de la prueba	CODI_PRUE_APAR	entero	Si	Si
Fecha de la prueba	FECH_PRUE_APAR	datetime	No	Si
Observaciones de la prueba	OBSE_PRUE_APAR	varchar(250)	No	No
Prueba satisfactoria	PRUE_SATI	entero	No	Si

Verificaciones de Columnas (Check):

Columna:	PRUE_SATI			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	0			
Letras Mayúsculas:	No	Letras Minúsculas:	No	
Lista de Valores:	1	Satisfactorio		
	0	No Satisfactorio		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_PRUE_APA	Si	No	Si	No	CODI_TIPO_PRUE_APA	ASC
R					R	ASC
					CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
					CORR_APAR_PACI	ASC
FK_APAR_PRUE_APAR	No	Si	No	No	CODI_PRUE_APAR	ASC
					CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
					CORR_APAR_PACI	ASC
FK_TIPO_PRUE_APAR	No	Si	No	No	CODI_TIPO_PRUE_APA	ASC
					R	

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_APAR_PACI	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
	CORR_APAR_PACI	CORR_APAR_PACI
SIAPDU_TIPO_PRUE_APA	CODI_TIPO_PRUE_APAR	CODI_TIPO_PRUE_APAR
R		

39. TABLA PUESTO DE TRABAJO.

Nombre:	Puesto de Trabajo
Código:	SIAPDU_PUES_TRAB
Descripción:	Puestos de Trabajo que pueden existir en una clínica de FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del puesto de trabajo	CODI_PUES_TRAB	entero	Si	Si
Nombre del puesto de trabajo	NOMB_PUES_TRAB	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_PUES_TRA B	Si	No	Si	No	CODI_PUES_TRAB	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_EMPL	CODI_PUES_TRAB	CODI_PUES_TRAB

40. TABLA REFERENCIA A OTRA INSTITUCIÓN.

Nombre:	Referencia a Otra Institución
Código:	SIAPDU_REFE_OTRA_INST
Descripción:	Referencias de pacientes a Otras Instituciones, generadas de una consulta de un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Correlativo de la referencia	CORR_REFE	entero	Si	Si
Código de la institucion	CODI_INST_MEDI	entero	No	Si
Fecha de la referencia	FECH_REFE	datetime	No	Si
Persona a que se refiere al	PERS_REFE_PACI	varchar(250)	No	No

Nombre	Código	Tipo	P	O
paciente				
Causa de la referencia	CAUS_REFE	varchar(250)	No	No
Plan de tratamiento sugerido	PLAN_TRAT_SUGE	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_REFE_OTR	Si	No	Si	No	CODI_PACI	ASC
A_INST					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
					CORR_REFE	ASC
FK_CONS_MEDI_REFE	No	Si	No	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
FK_OTRA_INST_MEDI_R	No	Si	No	No	CODI_INST_MEDI	ASC
EFE						

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_CONS	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
SIAPDU_OTRA_INST_MED	CODI_INST_MEDI	CODI_INST_MEDI

41. TABLA RESERVACIÓN DE CITA.

Nombre:	Reservación de Cita
Código:	SIAPDU_RESE_CITA
Descripción:	Reservaciones de Citas para Consultas (Médicas o PONI).

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la clínica	CODI_CLIN	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Código del solicitante	CODI_SOLI	entero	Si	Si

Nombre	Código	Tipo	P	O
Correlativo de la reservación	CORR_RESE	entero	Si	Si
Código del empleado	CODI_EMPL	entero	No	Si
Código del estatus de reservación de cita	CODI_ESTA_RESE_CITA	entero	No	Si
Código del medio de reservación	CODI_MEDI_RESE_CITA	entero	No	Si
Fecha y hora de la cita reservada	FECH_HORA_CITA_RESE	datetime	No	Si
Observaciones de reservación	OBSE_RESE_CITA	varchar(250)	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_RESE_CITA	Si	Si	Si	No	CODI_CLIN CODI_TIPO_SERV CODI_SOLI CORR_RESE	ASC ASC ASC ASC
FK_EMPL_RELA_CITA	No	Si	No	No	CODI_EMPL	ASC
FK_ESTA_RESE_CITA	No	Si	No	No	CODI_ESTA_RESE_CITA	ASC
FK_MEDI_RESE_CITA	No	Si	No	No	CODI_MEDI_RESE_CITA	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_EMPL	CODI_EMPL	CODI_EMPL
SIAPDU_ESTA_RESE_CIT	CODI_ESTA_RESE_CITA	CODI_ESTA_RESE_CITA
A SIAPDU_MEDI_RESE_CIT	CODI_MEDI_RESE_CITA	CODI_MEDI_RESE_CITA
A SIAPDU_SOLI	CODI_CLIN CODI_TIPO_SERV CODI_SOLI	CODI_CLIN CODI_TIPO_SERV CODI_SOLI

42. TABLA SERVICIO A PAGAR.

Nombre:	Servicio a Pagar
Código:	SIAPDU_SERV_PAGO

Descripción: Servicios por los cuales el paciente realizará un pago.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del servicio a pagar	CODI_SERV_PAGO	entero	Si	Si
Nombre del servicio a pagar	NOMB_SERV_PAGO	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_SERV_PAG O	Si	No	Si	No	CODI_SERV_PAGO	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_CONT_PAGO	CODI_SERV_PAGO	CODI_SERV_PAGO

43. TABLA SOLICITANTE.

Nombre:	Solicitante
Código:	SIAPDU_SOLI
Descripción:	Solicitante de Servicio de FUNTER (futuro Paciente de FUNTER)

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la clínica	CODI_CLIN	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Código del solicitante	CODI_SOLI	entero	Si	Si
Código del tipo de documento	CODI_TIPO_DOCU	entero	No	Si
Nombres del solicitante	NOMB_SOLI	varchar(100)	No	Si
Apellidos del solicitante	APEL_SOLI	varchar(100)	No	Si
Fecha de Nacimiento del solicitante	FECH_NACI_SOLI	datetime	No	Si
Número documento	NUME_DOCU_IDEN	varchar(50)	No	Si
Dirección de residencia del solicitante	DIRE_RESI_SOLI	varchar(250)	No	Si

Nombre	Código	Tipo	P	O
Teléfono de residencia del solicitante	TELE_RESI_SOLI	char(7)	No	No
Diagnóstico previo	DIAG_PREV	varchar(250)	No	No
Código de paciente	CODI_PACI_SOLI	entero	No	No

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_SOLI	Si	No	Si	No	CODI_CLIN	ASC
					CODI_TIPO_SERV	ASC
FK_TIPO_SERV_SOLI	No	Si	No	No	CODI_SOLI	ASC
					CODI_CLIN	ASC
FK_TIPO_DOCU_SOLI	No	Si	No	No	CODI_TIPO_SERV	ASC
					CODI_TIPO_DOCU	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_TIPO_SERV_CLIN	CODI_CLIN	CODI_CLIN
SIAPDU_TIPO_DOCU	CODI_TIPO_SERV CODI_TIPO_DOCU	CODI_TIPO_SERV CODI_TIPO_DOCU

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_RESE_CITA	CODI_CLIN CODI_TIPO_SERV CODI_SOLI	CODI_CLIN CODI_TIPO_SERV CODI_SOLI

44. TABLA TERAPIA DEL PACIENTE.

Nombre:	Terapia del Paciente
Código:	SIAPDU_TERA_PACI
Descripción:	Terapias de un Paciente, generadas como prescripcion de una consulta médica.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Correlativo de la cita	CORR_CITA	entero	Si	Si
Correlativo de la terapia	CORR_TERA_PACI	entero	Si	Si
Código del tipo de terapia	CODI_TIPO_TERA	entero	No	Si
Código del sub tipo de terapia	CODI_SUB_TIPO_TERA	entero	No	Si
Código del empleado	CODI_EMPL	entero	No	Si
Cantidad de terapias	CANT_TERA	entero	No	Si
Minutos de duración de terapia	MINU_DURA_TERA	entero	No	Si
Días de frecuencia de terapia	DIAS_FREQ_TERA	entero	No	Si
Fecha de inicio de terapia	FECH_INIC_TERA	datetime	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TERA_PACI	Si	No	Si	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
					CORR_TERA_PACI	ASC
FK_TIPO_TERA	No	Si	No	No	CODI_TIPO_TERA	ASC
FK_CONS_MEDI_TERA	No	Si	No	No	CODI_PACI	ASC
					CODI_TIPO_SERV	ASC
					CORR_CITA	ASC
FK_EMPL_TERA	No	Si	No	No	CODI_EMPL	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_CONS	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
SIAPDU_EMPL	CODI_EMPL	CODI_EMPL
SIAPDU_TIPO_TERA	CODI_TIPO_TERA	CODI_TIPO_TERA

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_CALE_TERA	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV
	CORR_CITA	CORR_CITA
	CORR_TERA_PACI	CORR_TERA_PACI
SIAPDU_EVAL_PACI	CODI_PACI	CODI_PACI
	CODI_TIPO_SERV	CODI_TIPO_SERV

Referenciada por	Llave Primaria	Llave Foránea
	CORR_CITA	CORR_CITA
	CORR_TERA_PACI	CORR_TERA_PACI

45. TABLA TIPO DE APARATO.

Nombre:	Tipo de Aparato
Código:	SIAPDU_TIPO_APAR
Descripción:	Tipos de aparatos que se fabrican en FUNTER para la rehabilitación de pacientes.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de aparato	CODI_TIPO_APAR	entero	Si	Si
Nombre del tipo de aparato	NOMB_TIPO_APAR	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_APAR	Si	No	Si	No	CODI_TIPO_APAR	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_APAR_PACI	CODI_TIPO_APAR	CODI_TIPO_APAR
SIPADU_TIPO_MEDI_APA	CODI_TIPO_APAR	CODI_TIPO_APAR
R		

46. TABLA TIPO DE DOCUMENTO.

Nombre:	Tipo de Documento
Código:	SIAPDU_TIPO_DOCU
Descripción:	Tipos de Documentos que podría tener una persona.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de documento	CODI_TIPO_DOCU	entero	Si	Si
Nombre del tipo de documento	NOMB_TIPO_DOCU	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_DOCU	Si	No	Si	No	CODI_TIPO_DOCU	ASC
U						

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_DOCU_PACI	CODI_TIPO_DOCU	CODI_TIPO_DOCU
SIAPDU_SOLI	CODI_TIPO_DOCU	CODI_TIPO_DOCU

47. TABLA TIPO DE EGRESO.

Nombre:	Tipo de Egreso
Código:	SIAPDU_TIPO_EGRE
Descripción:	Tipos de egresos económicos que podría tener un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de egreso	CODI_TIPO_EGRE	entero	Si	Si
Nombre del tipo de egreso	NOMB_TIPO_EGRE	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_EGRE	Si	No	Si	No	CODI_TIPO_EGRE	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_EGRE_PACI	CODI_TIPO_EGRE	CODI_TIPO_EGRE

48. TABLA TIPO DE INGRESO.

Nombre:	Tipo de Ingreso
----------------	-----------------

Código:	SIAPDU_TIPO_INGR
Descripción:	Tipos de Ingresos económicos que podría tener un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de ingreso	CODI_TIPO_INGR	entero	Si	Si
Nombre del tipo de ingreso	NOMB_TIPO_INGR	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_INGR	Si	No	Si	No	CODI_TIPO_INGR	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_INGR_PACI	CODI_TIPO_INGR	CODI_TIPO_INGR

49. TABLA TIPO DE INTERÉS VOCACIONAL

Nombre:	Tipo de Interés Vocacional
Código:	SIAPDU_TIPO_INTE_VOCA
Descripción:	Tipos de Interés vocacionales que podría tener un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de interés vocacional	CODI_TIPO_INTE_VOCA	entero	Si	Si
Nombre del tipo de enteroeres vocacional	NOMB_TIPO_INTE_VOCA	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_INTE_VOCA	Si	No	Si	No	CODI_TIPO_INTE_VOCA	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_INTE_VOCA_PAC I	CODI_TIPO_INTE_VOCA	CODI_TIPO_INTE_VOCA

50. TABLA TIPO DE MEDIDA DE APARATOS.

Nombre:	Tipo de Medida de Aparatos
Código:	SIPADU_TIPO_MEDI_APAR
Descripción:	Tipos de Medidas para los Aparatos.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de aparato	CODI_TIPO_APAR	entero	Si	Si
Código del tipo de medida	CODI_TIPO_MEDI	entero	Si	Si
Nombre del tipo de medida en inglés	NOMB_TIPO_MEDI_INGL	varchar(250)	No	Si
Nombre del tipo de medida en español	NOMB_TIPO_MEDI_ESPA	varchar(250)	No	No
Unidad del tipo de medida	UNID_TIPO_MEDI	varchar(50)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIPADU_TIPO_MEDI	Si	No	Si	No	CODI_TIPO_APAR	ASC
_APAR					CODI_TIPO_MEDI	ASC
FK_TIPO_APAR_TIPO_MEDI	No	Si	No	No	CODI_TIPO_APAR	ASC
EDI						

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_TIPO_APAR	CODI_TIPO_APAR	CODI_TIPO_APAR

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_MEDI_APAR	CODI_TIPO_APAR	CODI_TIPO_APAR
	CODI_TIPO_MEDI	CODI_TIPO_MEDI

51. TABLA TIPO DE OBSTÁCULO LABORAL.

Nombre:	Tipo de Obstáculo Laboral
Código:	SIAPDU_TIPO_OBST_LABO
Descripción:	Tipos de Obstáculos Laborales que podría tener un paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de obstáculo laboral	CODI_TIPO_OBST_LABO	entero	Si	Si
Nombre del tipo de obstáculo laboral	NOMB_TIPO_OBST_LAB O	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_OBST _LABO	Si	No	Si	No	CODI_TIPO_OBST_LABO	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_OBST_LABO_PA CI	CODI_TIPO_OBST_LABO	CODI_TIPO_OBST_LABO

52. TABLA TIPO DE PAGO DE VIVIENDA.

Nombre:	Tipo de Pago de Vivienda
Código:	SIAPDU_TIPO_PAGO_VIVI
Descripción:	Tipo de Pago de la Vivienda del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de pago de vivienda	CODI_TIPO_PAGO_VIVI	entero	Si	Si
Nombre del tipo de pago de vivienda	NOMB_TIPO_PAGO_VIVI	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_PAGO_VIVI	Si	No	Si	No	CODI_TIPO_PAGO_VIVI	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_VIVI_PACI	CODI_TIPO_PAGO_VIVI	CODI_TIPO_PAGO_VIVI

53. TABLA TIPO DE PARENTESCO.

Nombre:	Tipo de Parentesco
Código:	SIAPDU_TIPO_PARE
Descripción:	Tipos de Tabla Parentesco.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de Tabla Padreesco	CODI_TIPO_PARE	entero	Si	Si
Nombre del tipo de Tabla Padreesco	NOMB_TIPO_PARE	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_PARE	Si	No	Si	No	CODI_TIPO_PARE	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_FAMI_PACI	CODI_TIPO_PARE	CODI_TIPO_PARE

54. TABLA TIPO DE PRUEBA DEL APARATO.

Nombre:	Tipo de Prueba del Aparato
Código:	SIAPDU_TIPO_PRUE_APAR
Descripción:	Tipos de aparatos que se fabrican en FUNTER para la rehabilitación de pacientes.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de prueba de aparato	CODI_TIPO_PRUE_APAR	entero	Si	Si
Nombre del tipo de prueba del aparato	NOMB_TIPO_PRUE_APAR	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_PRUE_APAR	Si	No	Si	No	CODI_TIPO_PRUE_APAR	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_PRUE_APAR	CODI_TIPO_PRUE_APAR	CODI_TIPO_PRUE_APAR

55. TABLA TIPO DE SERVICIO.

Nombre:	Tipo de Servicio
Código:	SIAPDU_TIPO_SERV
Descripción:	Tipos de Servicios que se podrían brindar en una clínica de FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si
Nombre del tipo de servicio	NOMB_TIPO_SERV	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_SERV	Si	No	Si	No	CODI_TIPO_SERV	ASC

Lista de Referencia:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_TIPO_SERV_CLIN	CODI_TIPO_SERV	CODI_TIPO_SERV
SIAPDU_EXPE	CODI_TIPO_SERV	CODI_TIPO_SERV

56. TABLA TIPO DE SERVICIO DE UNA CLÍNICA.

Nombre:	Tipo de Servicio de una Clínica
Código:	SIAPDU_TIPO_SERV_CLIN
Descripción:	Tipos de Servicios que se brindan en una clínica específica de FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la clínica	CODI_CLIN	entero	Si	Si
Código del tipo de servicio	CODI_TIPO_SERV	entero	Si	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_SERV	Si	No	Si	No	CODI_CLIN	ASC
_CLIN					CODI_TIPO_SERV	ASC
FK_CLIN_TIPO_SERV	No	Si	No	No	CODI_CLIN	ASC
FK_TIPO_SERV_CLIN	No	Si	No	No	CODI_TIPO_SERV	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_CLIN	CODI_CLIN	CODI_CLIN
SIAPDU_TIPO_SERV	CODI_TIPO_SERV	CODI_TIPO_SERV

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_SOLI	CODI_CLIN	CODI_CLIN
SIAPDU_EMPL	CODI_TIPO_SERV	CODI_TIPO_SERV
	CODI_CLIN	CODI_CLIN
	CODI_TIPO_SERV	CODI_TIPO_SERV

57. TABLA TIPO DE TERAPIA.

Nombre:	Tipo de Terapia
Código:	SIAPDU_TIPO_TERA
Descripción:	Tipos de terapia que se brindan en FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de terapia	CODI_TIPO_TERA	entero	Si	Si
Nombre del tipo de terapia	NOMB_TIPO_TERA	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_TERA	Si	No	Si	No	CODI_TIPO_TERA	ASC

Lista de Referencia:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_TERA_PACI	CODI_TIPO_TERA	CODI_TIPO_TERA
SIAPDU_CUES_EVAL	CODI_TIPO_TERA	CODI_TIPO_TERA

58. TABLA TIPO DE VIVIENDA

Nombre:	Tipo de Vivienda
Código:	SIAPDU_TIPO_VIVI
Descripción:	Tipo de Vivienda del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de vivienda	CODI_TIPO_VIVI	entero	Si	Si
Nombre del tipo de vivienda	NOMB_TIPO_MOVI	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_TIPO_VIVI	Si	No	Si	No	CODI_TIPO_VIVI	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_VIVI_PACI	CODI_TIPO_VIVI	CODI_TIPO_VIVI

59. TABLA VIVIENDA DEL PACIENTE.

Nombre:	Vivienda del Paciente
----------------	-----------------------

Código:	SIAPDU_VIVI_PACI
Descripción:	Vivienda del Paciente.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del paciente	CODI_PACI	entero	Si	Si
Código de la vivienda	CODI_VIVI_PACI	entero	Si	Si
Código del tipo de vivienda	CODI_TIPO_VIVI	entero	No	Si
Código del tipo de pago de vivienda	CODI_TIPO_PAGO_VIVI	entero	No	Si
Dirección de la vivienda del paciente	DIRE_VIVI_PACI	varchar(250)	No	Si
Area de la vivienda	AREA_VIVI	char(1)	No	Si
Teléfono vivienda del paciente	TELE_VIVI_PACI	char(7)	No	No
Observaciones de la vivienda	OBSE_VIVI_PACI	varchar(250)	No	No

Verificaciones de Columnas (Check):

Columna:	AREA_VIVI			
Valor Mínimo:				
Valor Máximo:				
Valor Default:	U			
Letras Mayúsculas:	Si	Letras Minúsculas:	No	
Lista de Valores:	R	Rural		
	U	Urbana		

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_VIVI_PACI	Si	No	Si	No	CODI_PACI	ASC
					CODI_VIVI_PACI	ASC
FK_PACI_VIVI	No	Si	No	No	CODI_PACI	ASC
FK_TIPO_VIVI	No	Si	No	No	CODI_TIPO_VIVI	ASC
FK_TIPO_PAGO_VIVI	No	Si	No	No	CODI_TIPO_PAGO_VIVI	ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_PACI	CODI_PACI	CODI_PACI
SIAPDU_TIPO_PAGO_VIVI	CODI_TIPO_PAGO_VIVI	CODI_TIPO_PAGO_VIVI
SIAPDU_TIPO_VIVI	CODI_TIPO_VIVI	CODI_TIPO_VIVI

60. TABLA OPCIONES DEL SISTEMA.

Nombre:	Opciones del Sistema
Código:	SIAPDU_OPCI_SIST
Descripción:	Opciones del Sistema SIAPD.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código de la opción del sistema	CODI_OPCI_SIST	entero	Si	Si
Nombre de la opción del sistema	NOMB_OPCI_SIST	varchar(100)	No	Si
Página web de la opción del sistema	WEB_OPCI_SIST	varchar(50)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_OPCI_SIST	Si	No	Si	No	CODI_OPCI_SIST	ASC

Referencias:

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_OPCI_PERF_USU A	CODI_OPCI_SIST	CODI_OPCI_SIST

61. TABLA OPCIONES DEL PERFIL DE USUARIO.

Nombre:	Opciones del Perfil de Usuario
Código:	SIAPDU_OPCI_PERF_USUA
Descripción:	Opciones del Sistema sobre las que tiene derecho un Perfil de Usuario.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Perfil de Usuario	PERF_USUA	varchar(15)	Si	Si
Código de la opción del sistema	CODI_OPCI_SIST	entero	Si	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_OPCI_PERF_USUA	Si	No	Si	No	CODI_OPCI_SIST	ASC
FK_SIAPDU_O_OPCI_PE RF_SIAPD	No	Si	No	No	PERF_USUA CODI_OPCI_SIST	ASC ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_OPCI_SIST	CODI_OPCI_SIST	CODI_OPCI_SIST

62. TABLA SUB TIPO DE TERAPIA.

Nombre:	Sub Tipo de Terapia
Código:	SIAPDU_SUB_TIPO_TERA
Descripción:	Sub - Tipos de terapias que se brindan en FUNTER.

Columnas de la Tabla:

Nombre	Código	Tipo	P	O
Código del tipo de terapia	CODI_TIPO_TERA	entero	Si	Si
Código del sub tipo de terapia	CODI_SUB_TIPO_TERA	entero	Si	Si
Nombre del sub tipo de terapia	NOMB_SUB_TIPO_TERA	varchar(100)	No	Si

Índices:

Código del Índice	P	F	U	C	Columnas	Orden
PK_SIAPDU_SUB_TIPO_TERA	Si	No	Si	No	CODI_TIPO_TERA	ASC
FK_SIAPDU_S_TIPO_TERA	No	Si	No	No	CODI_SUB_TIPO_TERA CODI_TIPO_TERA	ASC ASC

Referencias:

Referencia a	Llave Primaria	Llave Foránea
SIAPDU_TIPO_TERA	CODI_TIPO_TERA	CODI_TIPO_TERA

Referenciada por	Llave Primaria	Llave Foránea
SIAPDU_TERA_PACI	CODI_SUB_TIPO_TERA	CODI_SUB_TIPO_TERA

ANEXO V: SCRIPT DE LA BASE DE DATOS.

```

/*****
/***** ROLES DE LA BASE DE DATOS *****/
*****/

/***** Object: DatabaseRole Doctor   Script Date: 04/09/2002 11:36:40 pm *****/
if not exists (select * from sysusers where name = N'Doctor' and uid > 16399)
    EXEC sp_addrole N'Doctor'
GO

/***** Object: DatabaseRole Terapista   Script Date: 04/09/2002 11:36:40 pm *****/
if not exists (select * from sysusers where name = N'Terapista' and uid > 16399)
    EXEC sp_addrole N'Terapista'
GO

/***** Object: DatabaseRole Protesista   Script Date: 04/09/2002 11:36:40 pm *****/
*****/
if not exists (select * from sysusers where name = N'Protesista' and uid > 16399)
    EXEC sp_addrole N'Protesista'
GO

/***** Object: DatabaseRole TrabSoci   Script Date: 04/09/2002 11:36:40 pm *****/
if not exists (select * from sysusers where name = N'TrabSoci' and uid > 16399)
    EXEC sp_addrole N'TrabSoci'
GO

/***** Object: DatabaseRole SecrClin   Script Date: 04/09/2002 11:36:40 pm *****/
if not exists (select * from sysusers where name = N'SecrClin' and uid > 16399)
    EXEC sp_addrole N'SecrClin'
GO

/***** Object: DatabaseRole SecrAtin   Script Date: 04/09/2002 11:36:40 pm *****/
if not exists (select * from sysusers where name = N'SecrAtin' and uid > 16399)
    EXEC sp_addrole N'SecrAtin'
GO

/***** Object: DatabaseRole Paciente   Script Date: 04/09/2002 11:36:40 pm *****/
if not exists (select * from sysusers where name = N'Paciente' and uid > 16399)
    EXEC sp_addrole N'Paciente'
GO

/***** Object: DatabaseRole Internet   Script Date: 04/09/2002 11:36:40 pm *****/
if not exists (select * from sysusers where name = N'Internet' and uid > 16399)
    EXEC sp_addrole N'Internet'
GO

```

```
/*  
***** TABLAS DE LA BASE DE DATOS  
*/
```

```
/*  
***** Object: Table [dbo].[SIAPDU_ANTE_SALU] Script Date: 04/09/2002  
11:36:47 pm  
*****
```

```
CREATE TABLE [dbo].[SIAPDU_ANTE_SALU] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_ANTE_SALU] [int] NOT NULL ,  
    [DESC_DISC_LIMI] [varchar] (250) NOT NULL ,  
    [CAUS_DISC_LIMI] [varchar] (250) NULL ,  
    [FECH_CAUS_DISC] [datetime] NULL ,  
    [TRAT_MEDI_RECI] [varchar] (250) NULL ,  
    [LUGA_AMPUI] [varchar] (250) NULL ,  
    [MOVI_USUA] [varchar] (250) NULL  
)
```

```
)  
GO
```

```
/*  
***** Object: Table [dbo].[SIAPDU_APAR_PACI] Script Date: 04/09/2002  
11:36:53 pm  
*****
```

```
CREATE TABLE [dbo].[SIAPDU_APAR_PACI] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [CORR_CITA] [int] NOT NULL ,  
    [CORR_APAR_PACI] [int] NOT NULL ,  
    [CODI_TIPO_APAR] [int] NOT NULL ,  
    [CODI_ESTA_APAR] [int] NOT NULL ,  
    [FECH_TOMA_MEDI] [datetime] NOT NULL ,  
    [MEDI_TOMA] [int] NOT NULL ,  
    [OBSE_TOMA_MEDI] [varchar] (250) NULL ,  
    [FECH_ENTR_PROG] [datetime] NULL ,  
    [FECH_ENTR_REAL] [datetime] NULL ,  
    [OBSE_ENTR] [varchar] (250) NULL  
)
```

```
)  
GO
```

```
/*  
***** Object: Table [dbo].[SIAPDU_ASPE_ACAD] Script Date: 04/09/2002  
11:36:55 pm  
*****
```

```
CREATE TABLE [dbo].[SIAPDU_ASPE_ACAD] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_ASPE_ACAD] [int] NOT NULL ,  
    [ESCO_ALCA] [varchar] (250) NOT NULL ,  
    [FECH_ESCO] [datetime] NOT NULL ,  
    [NOMB_CENT_EDUC] [varchar] (250) NOT NULL ,  
    [DIRE_CENT_EDUC] [varchar] (250) NULL ,  
    [OBSE_ASPE_ACAD] [varchar] (250) NULL  
)
```

)
GO

/***** Object: Table [dbo].[SIAPDU_ASPE_LABO] Script Date: 04/09/2002
11:36:57 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_ASPE_LABO] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_ASPE_LABO] [int] NOT NULL ,  
    [LUGA_TRAB] [varchar] (250) NOT NULL ,  
    [DIRE_TRAB] [varchar] (250) NULL ,  
    [PUES_OCUP] [varchar] (250) NOT NULL ,  
    [SALA_MENS_PACI] [money] NOT NULL ,  
    [DESC_TARE_REAL] [varchar] (250) NOT NULL ,  
    [DURA_TRAB_MES] [int] NULL ,  
    [MOTI_RETI] [varchar] (250) NULL ,  
    [ES_TRAB_ACTU] [int] NOT NULL ,  
    [OBSE_ASPE_LABO] [varchar] (250) NULL  
)
```

)
GO

/***** Object: Table [dbo].[SIAPDU_AUDI_TABL] Script Date: 04/09/2002 11:36:59
pm *****/

```
CREATE TABLE [dbo].[SIAPDU_AUDI_TABL] (  
    [CODI_TABL] [varchar] (100) NOT NULL ,  
    [CORR_ACCI] [int] NOT NULL ,  
    [TIPO_ACCI] [char] (1) NOT NULL ,  
    [LLAV_REGI] [varchar] (250) NOT NULL ,  
    [DESC_REGI] [varchar] (250) NOT NULL ,  
    [USUA_ACTU] [char] (20) NOT NULL ,  
    [FECH_ACTU] [datetime] NOT NULL  
)
```

)
GO

/***** Object: Table [dbo].[SIAPDU_CALE_TERA] Script Date: 04/09/2002
11:37:00 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_CALE_TERA] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [CORR_CITA] [int] NOT NULL ,  
    [CORR_TERA_PACI] [int] NOT NULL ,  
    [FECH_HORA_TERA] [datetime] NOT NULL ,  
    [TIPO_ATEN] [char] (1) NOT NULL ,  
    [DIAG_TERA] [varchar] (250) NULL ,  
    [OBSE_TERA] [varchar] (250) NULL ,  
    [ESTA_TERA] [char] (1) NOT NULL  
)
```

)
GO

/***** Object: Table [dbo].[SIAPDU_CITA] Script Date: 04/09/2002 11:37:02 pm
*****/

```
CREATE TABLE [dbo].[SIAPDU_CITA] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [CORR_CITA] [int] NOT NULL ,  
    [FECH_HORA_CITA] [datetime] NOT NULL ,  
    [OBSE_CITA] [varchar] (250) NULL ,  
    [ESTA_CITA] [int] NOT NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_CLAS_RANG_ECON] Script Date: 04/09/2002
11:37:03 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_CLAS_RANG_ECON] (  
    [CODI_RANG_ECON] [int] NOT NULL ,  
    [LIMI_ECON_INIC] [money] NOT NULL ,  
    [LIMI_ECON_FINI] [money] NOT NULL ,  
    [OBSE_RANG_ECON] [varchar] (250) NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_CLIN] Script Date: 04/09/2002 11:37:04 pm
*****/

```
CREATE TABLE [dbo].[SIAPDU_CLIN] (  
    [CODI_CLIN] [int] NOT NULL ,  
    [CODI_PAIS] [int] NOT NULL ,  
    [CODI_DEPA] [int] NOT NULL ,  
    [CODI_MUNI] [int] NOT NULL ,  
    [NOMB_CLIN] [varchar] (250) NOT NULL ,  
    [DIRE_CLIN] [varchar] (250) NOT NULL ,  
    [TELE_CLIN] [char] (7) NOT NULL ,  
    [FAX_CLIN] [char] (7) NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_CONS] Script Date: 04/09/2002 11:37:05 pm
*****/

```
CREATE TABLE [dbo].[SIAPDU_CONS] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [CORR_CITA] [int] NOT NULL ,  
    [CODI_EMPL] [int] NOT NULL ,  
    [PROC_CASO_PACI] [varchar] (250) NULL ,  
    [SINT_PACI] [varchar] (250) NOT NULL ,  
    [DIAG_CONS] [varchar] (250) NOT NULL ,
```

```
    [PRES_CONS] [char] (1) NOT NULL ,
    [RESE_CONS] [varchar] (250) NULL ,
    [OBSE_CONS] [varchar] (250) NULL ,
    [FECH_HORA_PROX_CONS] [datetime] NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_CONT_COMU]   Script Date: 04/09/2002
11:37:06 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_CONT_COMU] (
    [CODI_PACI] [int] NOT NULL ,
    [CODI_CONT_COMU] [int] NOT NULL ,
    [CARA_COMU] [varchar] (250) NULL ,
    [RRHH_REHA] [varchar] (250) NULL ,
    [OTRA_PERS_DISCA] [varchar] (250) NULL ,
    [ACTI_FAMI_COMU] [varchar] (250) NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_CONT_PAGO]   Script Date: 04/09/2002
11:37:08 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_CONT_PAGO] (
    [CODI_PACI] [int] NOT NULL ,
    [CORR_PAGO] [int] NOT NULL ,
    [CODI_SERV_PAGO] [int] NOT NULL ,
    [PRIMA_PAGO] [money] NOT NULL ,
    [ABON_PAGO] [money] NOT NULL ,
    [FORM_PAGO] [int] NOT NULL ,
    [CUOT_PROG_PAGO] [int] NULL ,
    [CUOT_REAL_PAGO] [int] NULL ,
    [ESTA_PAGO] [char] (1) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_CUES_EVAL]   Script Date: 04/09/2002
11:37:09 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_CUES_EVAL] (
    [CODI_TIPO_TERA] [int] NOT NULL ,
    [CODI_PREG] [int] NOT NULL ,
    [PREG] [varchar] (250) NOT NULL ,
    [ESTA_PREG] [int] NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_DEPA]   Script Date: 04/09/2002 11:37:12 pm
*****/
```

```
CREATE TABLE [dbo].[SIAPDU_DEPA] (
```

```
        [CODI_PAIS] [int] NOT NULL ,
        [CODI_DEPA] [int] NOT NULL ,
        [NOMB_DEPA] [varchar] (100) NOT NULL
    )
GO
```

```
/****** Object: Table [dbo].[SIAPDU_DINA_FAMI]   Script Date: 04/09/2002 11:37:13
pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_DINA_FAMI] (
    [CODI_PACI] [int] NOT NULL ,
    [CODI_DINA_FAMI] [int] NOT NULL ,
    [RELA_INTE] [varchar] (250) NULL ,
    [TOMA_DECI] [varchar] (250) NULL ,
    [ACTI_FAMI_ANTE_LIMI] [varchar] (250) NULL ,
    [ES_DINA_ACTU] [int] NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_DOCU_PACI]   Script Date: 04/09/2002
11:37:16 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_DOCU_PACI] (
    [CODI_TIPO_DOCU] [int] NOT NULL ,
    [CODI_PACI] [int] NOT NULL ,
    [NUME_DOCU] [varchar] (50) NOT NULL ,
    [LUGA_EXPE] [varchar] (100) NOT NULL ,
    [FECH_EXPE] [datetime] NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_EGRE_PACI]   Script Date: 04/09/2002
11:37:19 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_EGRE_PACI] (
    [CODI_TIPO_EGRE] [int] NOT NULL ,
    [CODI_PACI] [int] NOT NULL ,
    [CANT_EGRE] [money] NOT NULL ,
    [OBSE_EGRE] [varchar] (250) NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_EMPL]   Script Date: 04/09/2002 11:37:20 pm
*****/
```

```
CREATE TABLE [dbo].[SIAPDU_EMPL] (
    [CODI_EMPL] [int] NOT NULL ,
    [CODI_PUES_TRAB] [int] NOT NULL ,
    [CODI_CLIN] [int] NOT NULL ,
    [CODI_TIPO_SERV] [int] NOT NULL ,
    [NOMB_EMPL] [varchar] (100) NOT NULL ,

```

```
[APEL_EMPL] [varchar] (100) NOT NULL ,
[CODI_USUA] [char] (20) NULL ,
[SEXO_EMPL] [char] (1) NOT NULL ,
[ESTA_CIVI_EMPL] [char] (1) NOT NULL ,
[DIRE_RESI_EMPL] [varchar] (250) NOT NULL ,
[TELE_RESI_EMPL] [char] (7) NULL ,
[TELE_MOVI_EMPL] [char] (7) NULL ,
[ESTA_EMPL] [char] (1) NOT NULL
)
GO
```

```
/***** Object: Table [dbo].[SIAPDU_ESPE] Script Date: 04/09/2002 11:37:22 pm
*****/
```

```
CREATE TABLE [dbo].[SIAPDU_ESPE] (
    [CODI_ESPE] [int] NOT NULL ,
    [NOMB_ESPE] [varchar] (100) NOT NULL
)
GO
```

```
/***** Object: Table [dbo].[SIAPDU_ESPE_EMPL] Script Date: 04/09/2002
11:37:23 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_ESPE_EMPL] (
    [CODI_ESPE] [int] NOT NULL ,
    [CODI_EMPL] [int] NOT NULL
)
GO
```

```
/***** Object: Table [dbo].[SIAPDU_ESTA_APAR] Script Date: 04/09/2002
11:37:25 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_ESTA_APAR] (
    [CODI_ESTA_APAR] [int] NOT NULL ,
    [NOMB_ESTA_APAR] [varchar] (100) NOT NULL
)
GO
```

```
/***** Object: Table [dbo].[SIAPDU_ESTA_RESE_CITA] Script Date: 04/09/2002
11:37:27 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_ESTA_RESE_CITA] (
    [CODI_ESTA_RESE_CITA] [int] NOT NULL ,
    [NOMB_ESTA_RESE_CITA] [varchar] (100) NOT NULL
)
GO
```

```
/***** Object: Table [dbo].[SIAPDU_ESTU_SOC] Script Date: 04/09/2002
11:37:29 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_ESTU_SOC] (
    [CODI_RANG_ECON] [int] NOT NULL ,
```

```
        [CODI_PACI] [int] NOT NULL
    )
GO
```

```
/****** Object: Table [dbo].[SIAPDU_EVAL_PACI]   Script Date: 04/09/2002 11:37:29
pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_EVAL_PACI] (
    [CODI_TIPO_TERA] [int] NOT NULL ,
    [CODI_PREG] [int] NOT NULL ,
    [CODI_PACI] [int] NOT NULL ,
    [CODI_TIPO_SERV] [int] NOT NULL ,
    [CORR_CITA] [int] NOT NULL ,
    [CORR_TERA_PACI] [int] NOT NULL ,
    [RESP] [varchar] (250) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_EXPE]   Script Date: 04/09/2002 11:37:31 pm
*****/
```

```
CREATE TABLE [dbo].[SIAPDU_EXPE] (
    [CODI_PACI] [int] NOT NULL ,
    [CODI_TIPO_SERV] [int] NOT NULL ,
    [NUME_EXPE] [int] NOT NULL ,
    [FECH_APER_EXPE] [datetime] NOT NULL ,
    [ESTA_EXPE] [int] NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_FAMI_PACI]   Script Date: 04/09/2002 11:37:33
pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_FAMI_PACI] (
    [CODI_PACI] [int] NOT NULL ,
    [CODI_FAMI] [int] NOT NULL ,
    [CODI_TIPO_PARE] [int] NOT NULL ,
    [NOMB_FAMI] [varchar] (100) NOT NULL ,
    [APEL_FAMI] [varchar] (100) NOT NULL ,
    [ES_RESP_PACI] [int] NOT NULL ,
    [SEXO_FAMI] [char] (1) NOT NULL ,
    [ESTA_CIVI_FAMI] [char] (1) NOT NULL ,
    [FECH_NACI_FAMI] [datetime] NOT NULL ,
    [DIRE_FAMI] [varchar] (250) NOT NULL ,
    [TELE_RESI_FAMI] [char] (7) NULL ,
    [ESCO_FAMI] [varchar] (250) NULL ,
    [OCUP_FAMI] [varchar] (250) NULL ,
    [SALA_MENS_FAMI] [money] NULL
)
GO
```

/***** Object: Table [dbo].[SIAPDU_HORA_CLIN] Script Date: 04/09/2002
11:37:35 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_HORA_CLIN] (  
    [CODI_CLIN] [int] NOT NULL ,  
    [DIA_HORA] [int] NOT NULL ,  
    [HORA_ENTR] [datetime] NULL ,  
    [HORA_SALI] [datetime] NULL ,  
    [HORA_INIC] [datetime] NULL ,  
    [HORA_FINA] [datetime] NULL ,  
    [DURA_CONS] [int] NULL ,  
    [DURA_TERA] [int] NULL ,  
    [OBSE_HORA] [varchar] (250) NULL ,  
    [ESTA_HORA] [int] NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_HORA_EMPL] Script Date: 04/09/2002
11:37:37 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_HORA_EMPL] (  
    [CODI_EMPL] [int] NOT NULL ,  
    [DIA_HORA_EMPL] [int] NOT NULL ,  
    [HORA_ENTR_EMPL] [datetime] NULL ,  
    [HORA_SALI_EMPL] [datetime] NULL ,  
    [HORA_INIC_RECE] [datetime] NULL ,  
    [HORA_FINA_RECE] [datetime] NULL ,  
    [OBSE_HORA_EMPL] [varchar] (250) NULL ,  
    [ESTA_HORA_EMPL] [int] NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_INGR_PACI] Script Date: 04/09/2002 11:37:39
pm *****/

```
CREATE TABLE [dbo].[SIAPDU_INGR_PACI] (  
    [CODI_TIPO_INGR] [int] NOT NULL ,  
    [CODI_PACI] [int] NOT NULL ,  
    [CANT_INGR] [money] NOT NULL ,  
    [OBSE_INGR] [varchar] (250) NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_INTE_VOCA_PACI] Script Date: 04/09/2002
11:37:43 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_INTE_VOCA_PACI] (  
    [CODI_TIPO_INTE_VOCA] [int] NOT NULL ,  
    [CODI_PACI] [int] NOT NULL ,  
    [EXPE_INTE_VOCA] [varchar] (250) NULL ,
```

```
[MOTI_INTE_VOCA] [varchar] (250) NULL ,
[OPIN_FAMI_INTE_VOCA] [varchar] (250) NULL ,
[OBSE_INTE_VOCA] [varchar] (250) NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_MEDI_APAR]   Script Date: 04/09/2002
11:37:45 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_MEDI_APAR] (
    [CODI_TIPO_APAR] [int] NOT NULL ,
    [CODI_TIPO_MEDI] [int] NOT NULL ,
    [CODI_PACI] [int] NOT NULL ,
    [CODI_TIPO_SERV] [int] NOT NULL ,
    [CORR_CITA] [int] NOT NULL ,
    [CORR_APAR_PACI] [int] NOT NULL ,
    [MEDI_APAR] [varchar] (100) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_MEDI_RESE_CITA]   Script Date: 04/09/2002
11:37:46 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_MEDI_RESE_CITA] (
    [CODI_MEDI_RESE_CITA] [int] NOT NULL ,
    [NOMB_MEDI_RESE_CITA] [varchar] (100) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_MUNI]   Script Date: 04/09/2002 11:37:48 pm
*****/
```

```
CREATE TABLE [dbo].[SIAPDU_MUNI] (
    [CODI_PAIS] [int] NOT NULL ,
    [CODI_DEPA] [int] NOT NULL ,
    [CODI_MUNI] [int] NOT NULL ,
    [NOMB_MUNI] [varchar] (100) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_OBST_LABO_PACI]   Script Date: 04/09/2002
11:37:50 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_OBST_LABO_PACI] (
    [CODI_TIPO_OBST_LABO] [int] NOT NULL ,
    [CODI_PACI] [int] NOT NULL ,
    [OBSE_OBST_LABO] [varchar] (250) NULL
)
GO
```

/****** Object: Table [dbo].[SIAPDU_OPCI_PERF_USUA] Script Date: 04/09/2002
11:37:52 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_OPCI_PERF_USUA] (  
    [PERF_USUA] [varchar] (15) NOT NULL ,  
    [CODI_OPCI_SIST] [int] NOT NULL  
) ON [PRIMARY]  
GO
```

/****** Object: Table [dbo].[SIAPDU_OPCI_SIST] Script Date: 04/09/2002 11:37:55
pm *****/

```
CREATE TABLE [dbo].[SIAPDU_OPCI_SIST] (  
    [CODI_OPCI_SIST] [int] NOT NULL ,  
    [NOMB_OPCI_SIST] [varchar] (100) NOT NULL ,  
    [WEB_OPCI_SIST] [varchar] (50) NOT NULL  
)  
GO
```

/****** Object: Table [dbo].[SIAPDU_OTRA_INST_MEDI] Script Date: 04/09/2002
11:37:57 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_OTRA_INST_MEDI] (  
    [CODI_INST_MEDI] [int] NOT NULL ,  
    [NOMB_INST_MEDI] [varchar] (250) NOT NULL ,  
    [DIRE_INST_MEDI] [varchar] (250) NULL ,  
    [TELE_INST_MEDI] [char] (7) NULL ,  
    [FAX_INST_MEDI] [char] (7) NULL  
)  
GO
```

/****** Object: Table [dbo].[SIAPDU_PACI] Script Date: 04/09/2002 11:37:59 pm
*****/

```
CREATE TABLE [dbo].[SIAPDU_PACI] (  
    [CODI_PACI] [int] NOT NULL ,  
    [NOMB_PACI] [varchar] (100) NOT NULL ,  
    [APEL_PACI] [varchar] (100) NOT NULL ,  
    [SEXO_PACI] [char] (1) NOT NULL ,  
    [ESTA_CIVI_PACI] [char] (1) NOT NULL ,  
    [LUGA_NACI_PACI] [int] NOT NULL ,  
    [FECH_NACI_PACI] [datetime] NOT NULL ,  
    [ESTA_PACI] [int] NOT NULL  
)  
GO
```

/****** Object: Table [dbo].[SIAPDU_PAIS] Script Date: 04/09/2002 11:38:00 pm
*****/

```
CREATE TABLE [dbo].[SIAPDU_PAIS] (  
    [CODI_PAIS] [int] NOT NULL ,  
    [NOMB_PAIS] [varchar] (100) NOT NULL
```

)
GO

/***** Object: Table [dbo].[SIAPDU_PRUE_APAR] Script Date: 04/09/2002
11:38:02 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_PRUE_APAR] (  
    [CODI_TIPO_PRUE_APAR] [int] NOT NULL ,  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [CORR_CITA] [int] NOT NULL ,  
    [CORR_APAR_PACI] [int] NOT NULL ,  
    [CODI_PRUE_APAR] [int] NOT NULL ,  
    [FECH_PRUE_APAR] [datetime] NOT NULL ,  
    [OBSE_PRUE_APAR] [varchar] (250) NULL ,  
    [PRUE_SATI] [int] NOT NULL
```

)
GO

/***** Object: Table [dbo].[SIAPDU_PUES_TRAB] Script Date: 04/09/2002
11:38:03 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_PUES_TRAB] (  
    [CODI_PUES_TRAB] [int] NOT NULL ,  
    [NOMB_PUES_TRAB] [varchar] (100) NOT NULL
```

)
GO

/***** Object: Table [dbo].[SIAPDU_REFE_OTRA_INST] Script Date: 04/09/2002
11:38:05 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_REFE_OTRA_INST] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [CORR_CITA] [int] NOT NULL ,  
    [CORR_REFE] [int] NOT NULL ,  
    [CODI_INST_MEDI] [int] NOT NULL ,  
    [FECH_REFE] [datetime] NOT NULL ,  
    [PERS_REFE_PACI] [varchar] (250) NULL ,  
    [CAUS_REFE] [varchar] (250) NULL ,  
    [PLAN_TRAT_SUGE] [varchar] (250) NULL
```

)
GO

/***** Object: Table [dbo].[SIAPDU_RESE_CITA] Script Date: 04/09/2002
11:38:06 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_RESE_CITA] (  
    [CODI_CLIN] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [CODI_SOLI] [int] NOT NULL ,
```

```
        [CORR_RESE] [int] NOT NULL ,
        [CODI_EMPL] [int] NOT NULL ,
        [CODI_ESTA_RESE_CITA] [int] NOT NULL ,
        [CODI_MEDI_RESE_CITA] [int] NOT NULL ,
        [FECH_HORA_CITA_RESE] [datetime] NOT NULL ,
        [OBSE_RESE_CITA] [varchar] (250) NULL
    )
GO
```

```
/****** Object: Table [dbo].[SIAPDU_SERV_PAGO]   Script Date: 04/09/2002
11:38:08 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_SERV_PAGO] (
    [CODI_SERV_PAGO] [int] NOT NULL ,
    [NOMB_SERV_PAGO] [varchar] (100) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_SOLI]   Script Date: 04/09/2002 11:38:09 pm
*****/
```

```
CREATE TABLE [dbo].[SIAPDU_SOLI] (
    [CODI_CLIN] [int] NOT NULL ,
    [CODI_TIPO_SERV] [int] NOT NULL ,
    [CODI_SOLI] [int] NOT NULL ,
    [CODI_TIPO_DOCU] [int] NOT NULL ,
    [NOMB_SOLI] [varchar] (100) NOT NULL ,
    [APEL_SOLI] [varchar] (100) NOT NULL ,
    [FECH_NACI_SOLI] [datetime] NOT NULL ,
    [NUME_DOCU_IDEN] [varchar] (50) NOT NULL ,
    [DIRE_RESI_SOLI] [varchar] (250) NOT NULL ,
    [TELE_RESI_SOLI] [char] (7) NULL ,
    [DIAG_PREV] [varchar] (250) NULL ,
    [CODI_PACI_SOLI] [int] NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_SUB_TIPO_TERA]   Script Date: 04/09/2002
11:38:12 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_SUB_TIPO_TERA] (
    [CODI_TIPO_TERA] [int] NOT NULL ,
    [CODI_SUB_TIPO_TERA] [int] NOT NULL ,
    [NOMB_SUB_TIPO_TERA] [varchar] (100) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_TEMP_DISP_HORA]   Script Date: 04/09/2002
11:38:13 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_TEMP_DISP_HORA] (
```

```
        [CODI_EMPL] [int] NOT NULL ,
        [HORA] [char] (5) NOT NULL ,
        [DISP] [int] NOT NULL ,
        [CORR_TEMP] [int] NOT NULL
    ) ON [PRIMARY]
GO
```

```
/****** Object: Table [dbo].[SIAPDU_TERA_PACI]   Script Date: 04/09/2002
11:38:15 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_TERA_PACI] (
    [CODI_PACI] [int] NOT NULL ,
    [CODI_TIPO_SERV] [int] NOT NULL ,
    [CORR_CITA] [int] NOT NULL ,
    [CORR_TERA_PACI] [int] NOT NULL ,
    [CODI_TIPO_TERA] [int] NOT NULL ,
    [CODI_SUB_TIPO_TERA] [int] NULL ,
    [CODI_EMPL] [int] NOT NULL ,
    [CANT_TERA] [int] NOT NULL ,
    [MINU_DURA_TERA] [int] NOT NULL ,
    [DIAS_FREQ_TERA] [int] NOT NULL ,
    [FECH_INIC_TERA] [datetime] NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_TIPO_APAR]   Script Date: 04/09/2002
11:38:18 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_TIPO_APAR] (
    [CODI_TIPO_APAR] [int] NOT NULL ,
    [NOMB_TIPO_APAR] [varchar] (100) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_TIPO_DOCU]   Script Date: 04/09/2002
11:38:21 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_TIPO_DOCU] (
    [CODI_TIPO_DOCU] [int] NOT NULL ,
    [NOMB_TIPO_DOCU] [varchar] (100) NOT NULL
)
GO
```

```
/****** Object: Table [dbo].[SIAPDU_TIPO_EGRE]   Script Date: 04/09/2002
11:38:22 pm *****/
```

```
CREATE TABLE [dbo].[SIAPDU_TIPO_EGRE] (
    [CODI_TIPO_EGRE] [int] NOT NULL ,
    [NOMB_TIPO_EGRE] [varchar] (100) NOT NULL
)
GO
```

/***** Object: Table [dbo].[SIAPDU_TIPO_INGR] Script Date: 04/09/2002 11:38:23 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_INGR] (  
    [CODI_TIPO_INGR] [int] NOT NULL ,  
    [NOMB_TIPO_INGR] [varchar] (100) NOT NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_TIPO_INTE_VOCA] Script Date: 04/09/2002 11:38:24 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_INTE_VOCA] (  
    [CODI_TIPO_INTE_VOCA] [int] NOT NULL ,  
    [NOMB_TIPO_INTE_VOCA] [varchar] (100) NOT NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_TIPO_OBST_LABO] Script Date: 04/09/2002 11:38:25 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_OBST_LABO] (  
    [CODI_TIPO_OBST_LABO] [int] NOT NULL ,  
    [NOMB_TIPO_OBST_LABO] [varchar] (100) NOT NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_TIPO_PAGO_VIVI] Script Date: 04/09/2002 11:38:27 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_PAGO_VIVI] (  
    [CODI_TIPO_PAGO_VIVI] [int] NOT NULL ,  
    [NOMB_TIPO_PAGO_VIVI] [varchar] (100) NOT NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_TIPO_PARE] Script Date: 04/09/2002 11:38:28 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_PARE] (  
    [CODI_TIPO_PARE] [int] NOT NULL ,  
    [NOMB_TIPO_PARE] [varchar] (100) NOT NULL  
)  
GO
```

/***** Object: Table [dbo].[SIAPDU_TIPO_PRUE_APAR] Script Date: 04/09/2002 11:38:29 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_PRUE_APAR] (  
    [CODI_TIPO_PRUE_APAR] [int] NOT NULL ,  
    [NOMB_TIPO_PRUE_APAR] [varchar] (100) NOT NULL  
)
```

GO

/***** Object: Table [dbo].[SIAPDU_TIPO_SERV] Script Date: 04/09/2002

11:38:31 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_SERV] (  
    [CODI_TIPO_SERV] [int] NOT NULL ,  
    [NOMB_TIPO_SERV] [varchar] (100) NOT NULL
```

)

GO

/***** Object: Table [dbo].[SIAPDU_TIPO_SERV_CLIN] Script Date: 04/09/2002

11:38:32 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_SERV_CLIN] (  
    [CODI_CLIN] [int] NOT NULL ,  
    [CODI_TIPO_SERV] [int] NOT NULL
```

)

GO

/***** Object: Table [dbo].[SIAPDU_TIPO_TERA] Script Date: 04/09/2002

11:38:33 pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_TERA] (  
    [CODI_TIPO_TERA] [int] NOT NULL ,  
    [NOMB_TIPO_TERA] [varchar] (100) NOT NULL
```

)

GO

/***** Object: Table [dbo].[SIAPDU_TIPO_VIVI] Script Date: 04/09/2002 11:38:35

pm *****/

```
CREATE TABLE [dbo].[SIAPDU_TIPO_VIVI] (  
    [CODI_TIPO_VIVI] [int] NOT NULL ,  
    [NOMB_TIPO_VIVI] [varchar] (100) NOT NULL
```

)

GO

/***** Object: Table [dbo].[SIAPDU_VIVI_PACI] Script Date: 04/09/2002 11:38:36

pm *****/

```
CREATE TABLE [dbo].[SIAPDU_VIVI_PACI] (  
    [CODI_PACI] [int] NOT NULL ,  
    [CODI_VIVI_PACI] [int] NOT NULL ,  
    [CODI_TIPO_VIVI] [int] NOT NULL ,  
    [CODI_TIPO_PAGO_VIVI] [int] NOT NULL ,  
    [DIRE_VIVI_PACI] [varchar] (250) NOT NULL ,  
    [AREA_VIVI] [char] (1) NOT NULL ,  
    [TELE_VIVI_PACI] [char] (7) NULL ,  
    [OBSE_VIVI_PACI] [varchar] (250) NULL
```

)

GO

/***** Object: Table [dbo].[SIPADU_TIPO_MEDI_APAR] Script Date: 04/09/2002
11:38:37 pm *****/

```
CREATE TABLE [dbo].[SIPADU_TIPO_MEDI_APAR] (  
    [CODI_TIPO_APAR] [int] NOT NULL ,  
    [CODI_TIPO_MEDI] [int] NOT NULL ,  
    [NOMB_TIPO_MEDI_INGL] [varchar] (250) NOT NULL ,  
    [NOMB_TIPO_MEDI_ESPA] [varchar] (250) NULL ,  
    [UNID_TIPO_MEDI] [varchar] (50) NOT NULL  
)  
GO
```

```
/*****  
/***** CONSTRAINTS DE LA BASE DE DATOS *****/  
*****/
```

```
ALTER TABLE [dbo].[SIAPDU_ANTE_SALU] WITH NOCHECK ADD  
    CONSTRAINT [PK_SIAPDU_ANTE_SALU] PRIMARY KEY CLUSTERED  
    (  
        [CODI_PACI],  
        [CODI_ANTE_SALU]  
    ) ON [PRIMARY]  
GO
```

```
ALTER TABLE [dbo].[SIAPDU_APAR_PACI] WITH NOCHECK ADD  
    CONSTRAINT [PK_SIAPDU_APAR_PACI] PRIMARY KEY CLUSTERED  
    (  
        [CODI_PACI],  
        [CODI_TIPO_SERV],  
        [CORR_CITA],  
        [CORR_APAR_PACI]  
    ) ON [PRIMARY]  
GO
```

```
ALTER TABLE [dbo].[SIAPDU_ASPE_ACAD] WITH NOCHECK ADD  
    CONSTRAINT [PK_SIAPDU_ASPE_ACAD] PRIMARY KEY CLUSTERED  
    (  
        [CODI_PACI],  
        [CODI_ASPE_ACAD]  
    ) ON [PRIMARY]  
GO
```

```
ALTER TABLE [dbo].[SIAPDU_ASPE_LABO] WITH NOCHECK ADD  
    CONSTRAINT [PK_SIAPDU_ASPE_LABO] PRIMARY KEY CLUSTERED
```

```

        (
            [CODI_PACI],
            [CODI_ASPE_LABO]
        ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_AUDI_TABL] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_AUDI_TABL] PRIMARY KEY CLUSTERED
    (
        [CODI_TABL],
        [CORR_ACCI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_CALE_TERA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CALE_TERA] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_TERA_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_CITA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CITA] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_CLAS_RANG_ECON] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CLAS_RANG_ECON] PRIMARY KEY
    CLUSTERED
    (
        [CODI_RANG_ECON]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_CLIN] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CLIN] PRIMARY KEY CLUSTERED
    (
        [CODI_CLIN]
    ) ON [PRIMARY]
GO

```

```
ALTER TABLE [dbo].[SIAPDU_CONS] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CONS] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CONT_COMU] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CONT_COMU] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_CONT_COMU]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CONT_PAGO] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CONT_PAGO] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CORR_PAGO]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CUES_EVAL] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_CUES_EVAL] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_TERA],
        [CODI_PREG]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_DEPA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_DEPA] PRIMARY KEY CLUSTERED
    (
        [CODI_PAIS],
        [CODI_DEPA]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_DINA_FAMI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_DINA_FAMI] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_DINA_FAMI]
    )
```

```

    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_DOCU_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_DOCU_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_DOCU],
        [CODI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_EGRE_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_EGRE_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_EGRE],
        [CODI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_EMPL] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_EMPL] PRIMARY KEY CLUSTERED
    (
        [CODI_EMPL]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_ESPE] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_ESPE] PRIMARY KEY CLUSTERED
    (
        [CODI_ESPE]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_ESPE_EMPL] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_ESPE_EMPL] PRIMARY KEY CLUSTERED
    (
        [CODI_ESPE],
        [CODI_EMPL]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_ESTA_APAR] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_ESTA_APAR] PRIMARY KEY CLUSTERED
    (
        [CODI_ESTA_APAR]
    ) ON [PRIMARY]
GO

```

```

ALTER TABLE [dbo].[SIAPDU_ESTA_RESE_CITA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_ESTA_RESE_CITA] PRIMARY KEY
    CLUSTERED
    (
        [CODI_ESTA_RESE_CITA]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_ESTU_SOCIO] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_ESTU_SOCIO] PRIMARY KEY CLUSTERED
    (
        [CODI_RANG_ECON],
        [CODI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_EVAL_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_EVAL_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_TERA],
        [CODI_PREG],
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_TERA_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_EXPE] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_EXPE] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_TIPO_SERV]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_FAMI_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_FAMI_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_FAMI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_HORA_CLIN] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_HORA_CLIN] PRIMARY KEY CLUSTERED

```

```

        (
            [CODI_CLIN],
            [DIA_HORA]
        ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_HORA_EMPL] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_HORA_EMPL] PRIMARY KEY CLUSTERED
    (
        [CODI_EMPL],
        [DIA_HORA_EMPL]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_INGR_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_INGR_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_INGR],
        [CODI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_INTE_VOCA_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_INTE_VOCA_PACI] PRIMARY KEY
    CLUSTERED
    (
        [CODI_TIPO_INTE_VOCA],
        [CODI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_MEDI_APAR] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_MEDI_APAR] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_APAR],
        [CODI_TIPO_MEDI],
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_APAR_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_MEDI_RESE_CITA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_MEDI_RESE_CITA] PRIMARY KEY
    CLUSTERED
    (

```

```

        [CODI_MEDI_RESE_CITA]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_MUNI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_MUNI] PRIMARY KEY CLUSTERED
    (
        [CODI_PAIS],
        [CODI_DEPA],
        [CODI_MUNI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_OBST_LABO_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_OBST_LABO_PACI] PRIMARY KEY
    CLUSTERED
    (
        [CODI_TIPO_OBST_LABO],
        [CODI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_OPCI_SIST] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_OPCI_SIST] PRIMARY KEY CLUSTERED
    (
        [CODI_OPCI_SIST]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_OTRA_INST_MEDI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_OTRA_INST_MEDI] PRIMARY KEY
    CLUSTERED
    (
        [CODI_INST_MEDI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_PAIS] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_PAIS] PRIMARY KEY CLUSTERED
    (

```

```

        [CODI_PAIS]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_PRUE_APAR] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_PRUE_APAR] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_PRUE_APAR],
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_APAR_PACI],
        [CODI_PRUE_APAR]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_PUES_TRAB] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_PUES_TRAB] PRIMARY KEY CLUSTERED
    (
        [CODI_PUES_TRAB]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_REFE_OTRA_INST] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_REFE_OTRA_INST] PRIMARY KEY
    CLUSTERED
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_REFE]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_RESE_CITA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_RESE_CITA] PRIMARY KEY CLUSTERED
    (
        [CODI_CLIN],
        [CODI_TIPO_SERV],
        [CODI_SOLI],
        [CORR_RESE]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_SERV_PAGO] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_SERV_PAGO] PRIMARY KEY CLUSTERED
    (

```

```
        [CODI_SERV_PAGO]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_SOLI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_SOLI] PRIMARY KEY CLUSTERED
    (
        [CODI_CLIN],
        [CODI_TIPO_SERV],
        [CODI_SOLI]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_SUB_TIPO_TERA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_SUB_TIPO_TERA] PRIMARY KEY
    CLUSTERED
    (
        [CODI_TIPO_TERA],
        [CODI_SUB_TIPO_TERA]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TERA_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_TERA_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_TERA_PACI]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_APAR] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_TIPO_APAR] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_APAR]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_DOCU] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_TIPO_DOCU] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_DOCU]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_EGRE] WITH NOCHECK ADD
```

```
CONSTRAINT [PK_SIAPDU_TIPO_EGRE] PRIMARY KEY CLUSTERED
(
    [CODI_TIPO_EGRE]
) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_INGR] WITH NOCHECK ADD
CONSTRAINT [PK_SIAPDU_TIPO_INGR] PRIMARY KEY CLUSTERED
(
    [CODI_TIPO_INGR]
) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_INTE_VOCA] WITH NOCHECK ADD
CONSTRAINT [PK_SIAPDU_TIPO_INTE_VOCA] PRIMARY KEY
CLUSTERED
(
    [CODI_TIPO_INTE_VOCA]
) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_OBST_LABO] WITH NOCHECK ADD
CONSTRAINT [PK_SIAPDU_TIPO_OBST_LABO] PRIMARY KEY
CLUSTERED
(
    [CODI_TIPO_OBST_LABO]
) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_PAGO_VIVI] WITH NOCHECK ADD
CONSTRAINT [PK_SIAPDU_TIPO_PAGO_VIVI] PRIMARY KEY
CLUSTERED
(
    [CODI_TIPO_PAGO_VIVI]
) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_PARE] WITH NOCHECK ADD
CONSTRAINT [PK_SIAPDU_TIPO_PARE] PRIMARY KEY CLUSTERED
(
    [CODI_TIPO_PARE]
) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_TIPO_PRUE_APAR] WITH NOCHECK ADD
CONSTRAINT [PK_SIAPDU_TIPO_PRUE_APAR] PRIMARY KEY
CLUSTERED
```

```

        (
            [CODI_TIPO_PRUE_APAR]
        ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_TIPO_SERV] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_TIPO_SERV] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_SERV]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_TIPO_SERV_CLIN] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_TIPO_SERV_CLIN] PRIMARY KEY
    CLUSTERED
    (
        [CODI_CLIN],
        [CODI_TIPO_SERV]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_TIPO_TERA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_TIPO_TERA] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_TERA]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_TIPO_VIVI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_TIPO_VIVI] PRIMARY KEY CLUSTERED
    (
        [CODI_TIPO_VIVI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIAPDU_VIVI_PACI] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_VIVI_PACI] PRIMARY KEY CLUSTERED
    (
        [CODI_PACI],
        [CODI_VIVI_PACI]
    ) ON [PRIMARY]
GO

ALTER TABLE [dbo].[SIPADU_TIPO_MEDI_APAR] WITH NOCHECK ADD
    CONSTRAINT [PK_SIPADU_TIPO_MEDI_APAR] PRIMARY KEY
    CLUSTERED
    (

```

```
        [CODI_TIPO_APAR],
        [CODI_TIPO_MEDI]
    ) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_APAR_PACI] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_AP__MEDI___5165187F] DEFAULT (0) FOR
[MEDI_TOMA],
    CONSTRAINT [CKC_MEDI_TOMA_SIAPDU_A] CHECK ([MEDI_TOMA] = 1
or [MEDI_TOMA] = 0)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_ASPE_LABO] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_AS__ES_TR__787EE5A0] DEFAULT (1) FOR
[ES_TRAB_ACTU],
    CONSTRAINT [CKC_ES_TRAB_ACTU_SIAPDU_A] CHECK
([ES_TRAB_ACTU] = 1 or [ES_TRAB_ACTU] = 0)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_AUDI_TABL] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_AU__TIPO___08EA5793] DEFAULT ('I') FOR
[TIPO_ACCI],
    CONSTRAINT [CKC_TIPO_ACCI_SIAPDU_A] CHECK ([TIPO_ACCI] = 'U' or
[TIPO_ACCI] = 'D' or [TIPO_ACCI] = 'I')
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CALE_TERA] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_CA__TIPO___0B91BA14] DEFAULT ('E') FOR
[TIPO_ATEN],
    CONSTRAINT [DF__SIAPDU_CA__ESTA___0D7A0286] DEFAULT ('A') FOR
[ESTA_TERA],
    CONSTRAINT [CKC_ESTA_TERA_SIAPDU_C] CHECK ([ESTA_TERA] = 'I'
or [ESTA_TERA] = 'P' or [ESTA_TERA] = 'A'),
    CONSTRAINT [CKC_TIPO_ATEN_SIAPDU_C] CHECK ([TIPO_ATEN] = 'S' or
[TIPO_ATEN] = 'E')
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CITA] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_CI__ESTA___59FA5E80] DEFAULT (1) FOR
[ESTA_CITA],
    CONSTRAINT [CKC_ESTA_CITA_SIAPDU_C] CHECK ([ESTA_CITA] = 2 or
[ESTA_CITA] = 1 or [ESTA_CITA] = 0)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CONS] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_CO__PRES___4BAC3F29] DEFAULT ('T')
FOR [PRES_CONS],
```

```
        CONSTRAINT [CKC_PRES_CONS_SIAPDU_C] CHECK ([PRES_CONS] =
'R' or [PRES_CONS] = 'T' or [PRES_CONS] = 'A')
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CONT_PAGO] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_CO__PRIMA__114A936A] DEFAULT (0.00)
FOR [PRIMA_PAGO],
    CONSTRAINT [DF__SIAPDU_CO__ABON__123EB7A3] DEFAULT (0.00)
FOR [ABON_PAGO],
    CONSTRAINT [DF__SIAPDU_CO__FORM__1332DBDC] DEFAULT (1)
FOR [FORM_PAGO],
    CONSTRAINT [DF__SIAPDU_CO__ESTA__151B244E] DEFAULT ('C') FOR
[ESTA_PAGO],
    CONSTRAINT [CKC_ESTA_PAGO_SIAPDU_C] CHECK ([ESTA_PAGO] = 'P'
or [ESTA_PAGO] = 'C'),
    CONSTRAINT [CKC_FORM_PAGO_SIAPDU_C] CHECK ([FORM_PAGO] = 2
or [FORM_PAGO] = 1)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_CUES_EVAL] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_CU__ESTA__5FB337D6] DEFAULT (1) FOR
[ESTA_PREG],
    CONSTRAINT [CKC_ESTA_PREG_SIAPDU_C] CHECK ([ESTA_PREG] = 0
or [ESTA_PREG] = 1)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_DINA_FAMI] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_DI__ES_DI__1CBC4616] DEFAULT (0) FOR
[ES_DINA_ACTU],
    CONSTRAINT [CKC_ES_DINA_ACTU_SIAPDU_D] CHECK
([ES_DINA_ACTU] = 1 or [ES_DINA_ACTU] = 0)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_EMPL] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_EM__SEXO__440B1D61] DEFAULT ('M')
FOR [SEXO_EMPL],
    CONSTRAINT [DF__SIAPDU_EM__ESTA__45F365D3] DEFAULT ('S') FOR
[ESTA_CIVI_EMPL],
    CONSTRAINT [DF__SIAPDU_EM__ESTA__47DBAE45] DEFAULT ('A')
FOR [ESTA_EMPL],
    CONSTRAINT [CKC_ESTA_CIVI_EMPL_SIAPDU_E] CHECK
([ESTA_CIVI_EMPL] = 'V' or [ESTA_CIVI_EMPL] = 'S' or [ESTA_CIVI_EMPL] = 'D' or
[ESTA_CIVI_EMPL] = 'C' or [ESTA_CIVI_EMPL] = 'A'),
    CONSTRAINT [CKC_ESTA_EMPL_SIAPDU_E] CHECK ([ESTA_EMPL] = 'R'
or [ESTA_EMPL] = 'D' or [ESTA_EMPL] = 'A'),
    CONSTRAINT [CKC_SEXO_EMPL_SIAPDU_E] CHECK ([SEXO_EMPL] = 'M'
or [SEXO_EMPL] = 'F')
```

GO

```
ALTER TABLE [dbo].[SIAPDU_EXPE] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_EX_FECH___5535A963] DEFAULT
(getdate()) FOR [FECH_APER_EXPE],
    CONSTRAINT [DF__SIAPDU_EX_ESTA___5629CD9C] DEFAULT (1) FOR
[ESTA_EXPE],
    CONSTRAINT [CKC_ESTA_EXPE_SIAPDU_E] CHECK ([ESTA_EXPE] = 0 or
[ESTA_EXPE] = 1)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_FAMI_PACI] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_FA_ES_RE___6D0D32F4] DEFAULT (1) FOR
[ES_RESP_PACI],
    CONSTRAINT [DF__SIAPDU_FA_SEXO___6EF57B66] DEFAULT ('M') FOR
[SEXO_FAMI],
    CONSTRAINT [DF__SIAPDU_FA_ESTA___70DDC3D8] DEFAULT ('S')
FOR [ESTA_CIVI_FAMI],
    CONSTRAINT [CKC_ES_RESP_PACI_SIAPDU_F] CHECK
([ES_RESP_PACI] = 1 or [ES_RESP_PACI] = 0),
    CONSTRAINT [CKC_ESTA_CIVI_FAMI_SIAPDU_F] CHECK
([ESTA_CIVI_FAMI] = 'V' or [ESTA_CIVI_FAMI] = 'S' or [ESTA_CIVI_FAMI] = 'D' or
[ESTA_CIVI_FAMI] = 'C' or [ESTA_CIVI_FAMI] = 'A'),
    CONSTRAINT [CKC_SEXO_FAMI_SIAPDU_F] CHECK ([SEXO_FAMI] = 'F'
or [SEXO_FAMI] = 'M')
GO
```

```
ALTER TABLE [dbo].[SIAPDU_HORA_CLIN] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_HO_DIA_H___282DF8C2] DEFAULT (1) FOR
[DIA_HORA],
    CONSTRAINT [DF__SIAPDU_HO_ESTA___2A164134] DEFAULT (1) FOR
[ESTA_HORA],
    CONSTRAINT [CKC_DIA_HORA_SIAPDU_H] CHECK ([DIA_HORA] >= 1 and
[DIA_HORA] <= 7 and ([DIA_HORA] = 7 or [DIA_HORA] = 6 or [DIA_HORA] = 5 or
[DIA_HORA] = 4 or [DIA_HORA] = 3 or [DIA_HORA] = 2 or [DIA_HORA] = 1)),
    CONSTRAINT [CKC_ESTA_HORA_SIAPDU_H] CHECK ([ESTA_HORA] = 1
or [ESTA_HORA] = 0)
GO
```

```
ALTER TABLE [dbo].[SIAPDU_HORA_EMPL] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_HO_DIA_H___22751F6C] DEFAULT (1) FOR
[DIA_HORA_EMPL],
    CONSTRAINT [DF__SIAPDU_HO_ESTA___245D67DE] DEFAULT (1) FOR
[ESTA_HORA_EMPL],
    CONSTRAINT [CKC_DIA_HORA_EMPL_SIAPDU_H] CHECK
([DIA_HORA_EMPL] >= 1 and [DIA_HORA_EMPL] <= 7 and ([DIA_HORA_EMPL] =
```

```

7 or [DIA_HORA_EMPL] = 6 or [DIA_HORA_EMPL] = 5 or [DIA_HORA_EMPL] = 4 or
[DIA_HORA_EMPL] = 3 or [DIA_HORA_EMPL] = 2 or [DIA_HORA_EMPL] = 1)),
    CONSTRAINT [CKC_ESTA_HORA_EMPL_SIAPDU_H] CHECK
([ESTA_HORA_EMPL] = 1 or [ESTA_HORA_EMPL] = 0)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_OPCI_PERF_USUA] WITH NOCHECK ADD
    CONSTRAINT [PK_SIAPDU_OPCI_PERF_USUA] PRIMARY KEY
NONCLUSTERED
(
    [PERF_USUA],
    [CODI_OPCI_SIST]
) ON [PRIMARY]
GO

```

```

ALTER TABLE [dbo].[SIAPDU_PACI] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_PA__SEXO___145C0A3F] DEFAULT ('M')
FOR [SEXO_PACI],
    CONSTRAINT [DF__SIAPDU_PA__ESTA___164452B1] DEFAULT ('S') FOR
[ESTA_CIVI_PACI],
    CONSTRAINT [DF_SIAPDU_PACI_ESTA_PACI] DEFAULT (1) FOR
[ESTA_PACI],
    CONSTRAINT [CKC_ESTA_CIVI_PACI_SIAPDU_P] CHECK
([ESTA_CIVI_PACI] = 'V' or [ESTA_CIVI_PACI] = 'S' or [ESTA_CIVI_PACI] = 'D' or
[ESTA_CIVI_PACI] = 'C' or [ESTA_CIVI_PACI] = 'A'),
    CONSTRAINT [CKC_ESTA_PACI_SIAPDU_P] CHECK ([ESTA_PACI] = 1 or
[ESTA_PACI] = 0),
    CONSTRAINT [CKC_SEXO_PACI_SIAPDU_P] CHECK ([SEXO_PACI] = 'M'
or [SEXO_PACI] = 'F')
GO

```

```

ALTER TABLE [dbo].[SIAPDU_PRUE_APAR] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_PR__PRUE___6754599E] DEFAULT (0) FOR
[PRUE_SATI],
    CONSTRAINT [CKC_PRUE_SATI_SIAPDU_P] CHECK ([PRUE_SATI] = 0 or
[PRUE_SATI] = 1)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_RESE_CITA] WITH NOCHECK ADD
    CONSTRAINT [DF_SIAPDU_RESE_CITA_CORR_RESE] DEFAULT (1) FOR
[CORR_RESE]
GO

```

```

ALTER TABLE [dbo].[SIAPDU_SOLI] WITH NOCHECK ADD
    CONSTRAINT [DF_SIAPDU_SOLI_FECH_NACI_SOLI] DEFAULT (getdate())
FOR [FECH_NACI_SOLI]
GO

```

```

ALTER TABLE [dbo].[SIAPDU_TEMP_DISP_HORA] WITH NOCHECK ADD
    CONSTRAINT [DF_SIAPDU_TEMP_DISP_HORA_DISP] DEFAULT (0) FOR
[DISP],
    CONSTRAINT [PK_SIAPDU_TEMP_DISP_HORA] PRIMARY KEY
NONCLUSTERED
    (
        [CORR_TEMP]
    ) ON [PRIMARY]
GO

```

```

ALTER TABLE [dbo].[SIAPDU_VIVI_PACI] WITH NOCHECK ADD
    CONSTRAINT [DF__SIAPDU_VI__AREA__74AE54BC] DEFAULT ('U') FOR
[AREA_VIVI],
    CONSTRAINT [CKC_AREA_VIVI_SIAPDU_V] CHECK ([AREA_VIVI] = 'U' or
[AREA_VIVI] = 'R')
GO

```

```

/*****
/***** INDICES DE LA BASE DE DATOS *****/
/*****

```

```

CREATE INDEX [FK_PACI_ANTE_SALU] ON [dbo].[SIAPDU_ANTE_SALU]
([CODI_PACI]) ON [PRIMARY]
GO

```

```

CREATE INDEX [FK_TIPO_APAR] ON [dbo].[SIAPDU_APAR_PACI]
([CODI_TIPO_APAR]) ON [PRIMARY]
GO

```

```

CREATE INDEX [FK_CONSI_APAR] ON [dbo].[SIAPDU_APAR_PACI]
([CODI_PACI], [CODI_TIPO_SERV], [CORR_CITA]) ON [PRIMARY]
GO

```

```

CREATE INDEX [FK_ESTA_APAR] ON [dbo].[SIAPDU_APAR_PACI]
([CODI_ESTA_APAR]) ON [PRIMARY]
GO

```

```

CREATE INDEX [FK_PACI_ACAD] ON [dbo].[SIAPDU_ASPE_ACAD]
([CODI_PACI]) ON [PRIMARY]
GO

```

```

CREATE INDEX [FK_PACI_LABO] ON [dbo].[SIAPDU_ASPE_LABO]
([CODI_PACI]) ON [PRIMARY]

```

GO

```
CREATE INDEX [FK_EXPE_CITA] ON [dbo].[SIAPDU_CITA]([CODI_PACI],  
[CODI_TIPO_SERV]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_MUNI_CLIN] ON [dbo].[SIAPDU_CLIN]([CODI_PAIS],  
[CODI_DEPA], [CODI_MUNI]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_EMPL_CONS] ON [dbo].[SIAPDU_CONS]([CODI_EMPL]) ON  
[PRIMARY]  
GO
```

```
CREATE INDEX [FK_PACI_CONT_COMU] ON [dbo].[SIAPDU_CONT_COMU]  
([CODI_PACI]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_PACI_PAGO] ON [dbo].[SIAPDU_CONT_PAGO]  
([CODI_PACI]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_PAGO_SERV] ON [dbo].[SIAPDU_CONT_PAGO]  
([CODI_SERV_PAGO]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_TIPO_TERA_CUES] ON [dbo].[SIAPDU_CUES_EVAL]  
([CODI_TIPO_TERA]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_PAIS_DEPA] ON [dbo].[SIAPDU_DEPA]([CODI_PAIS]) ON  
[PRIMARY]  
GO
```

```
CREATE INDEX [FK_PACI_DINA_FAMI] ON [dbo].[SIAPDU_DINA_FAMI]  
([CODI_PACI]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_PACI_DOCU] ON [dbo].[SIAPDU_DOCU_PACI]  
([CODI_PACI]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_TIPO_DOCU] ON [dbo].[SIAPDU_DOCU_PACI]  
([CODI_TIPO_DOCU]) ON [PRIMARY]  
GO
```

```
CREATE INDEX [FK_TIPO_EGRE] ON [dbo].[SIAPDU_EGRE_PACI]
([CODI_TIPO_EGRE]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_EGRE] ON [dbo].[SIAPDU_EGRE_PACI]
([CODI_PACI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PUES_TRAB_EMPL] ON [dbo].[SIAPDU_EMPL]
([CODI_PUES_TRAB]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_SERV_CLIN_EMPL] ON [dbo].[SIAPDU_EMPL]
([CODI_CLIN], [CODI_TIPO_SERV]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_EMPL_ESPE_EMPL] ON [dbo].[SIAPDU_ESPE_EMPL]
([CODI_EMPL]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_ESPE_ESPE_EMPL] ON [dbo].[SIAPDU_ESPE_EMPL]
([CODI_ESPE]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_ESTU_SOC] ON [dbo].[SIAPDU_ESTU_SOC]
([CODI_PACI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_RANG_ECON_ESTU] ON [dbo].[SIAPDU_ESTU_SOC]
([CODI_RANG_ECON]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_CUEST_EVAL] ON [dbo].[SIAPDU_EVAL_PACI]
([CODI_TIPO_TERA], [CODI_PREG]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TERA_EVAL] ON [dbo].[SIAPDU_EVAL_PACI]([CODI_PACI],
[CODI_TIPO_SERV], [CORR_CITA], [CORR_TERA_PACI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_SERV_EXPE] ON [dbo].[SIAPDU_EXPE]
([CODI_TIPO_SERV]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_EXPE] ON [dbo].[SIAPDU_EXPE]([CODI_PACI]) ON
[PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_FAMI] ON [dbo].[SIAPDU_FAMI_PACI]([CODI_PACI])
ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_PARE] ON [dbo].[SIAPDU_FAMI_PACI]
([CODI_TIPO_PARE]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_HORA_CLIN] ON [dbo].[SIAPDU_HORA_CLIN]
([CODI_CLIN]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_HORA_EMPL] ON [dbo].[SIAPDU_HORA_EMPL]
([CODI_EMPL]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_INGR] ON [dbo].[SIAPDU_INGR_PACI]([CODI_PACI])
ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_INGR] ON [dbo].[SIAPDU_INGR_PACI]
([CODI_TIPO_INGR]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_INTE_VOCA] ON [dbo].[SIAPDU_INTE_VOCA_PACI]
([CODI_PACI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_INTE_VOCA] ON [dbo].[SIAPDU_INTE_VOCA_PACI]
([CODI_TIPO_INTE_VOCA]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_APAR_MEDI_APAR] ON [dbo].[SIAPDU_MEDI_APAR]
([CODI_PACI], [CODI_TIPO_SERV], [CORR_CITA], [CORR_APAR_PACI]) ON
[PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_MEDI_MEDI_APAR] ON [dbo].[SIAPDU_MEDI_APAR]
([CODI_TIPO_APAR], [CODI_TIPO_MEDI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_DEPA_MUNI] ON [dbo].[SIAPDU_MUNI]([CODI_PAIS],
[CODI_DEPA]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_OBST_LABO] ON [dbo].[SIAPDU_OBST_LABO_PACI]
([CODI_PACI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_OBST_LABO] ON [dbo].[SIAPDU_OBST_LABO_PACI]
([CODI_TIPO_OBST_LABO]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_APAR_PRUE_APAR] ON [dbo].[SIAPDU_PRUE_APAR]
([CODI_PACI], [CODI_TIPO_SERV], [CORR_CITA], [CORR_APAR_PACI]) ON
[PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_PRUE_APAR] ON [dbo].[SIAPDU_PRUE_APAR]
([CODI_TIPO_PRUE_APAR]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_CONS_MEDI_REFE] ON [dbo].
[SIAPDU_REFE_OTRA_INST]([CODI_PACI], [CODI_TIPO_SERV], [CORR_CITA])
ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_OTRA_INST_MEDI_REFE] ON [dbo].
[SIAPDU_REFE_OTRA_INST]([CODI_INST_MEDI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_EMPL_RELA_CITA] ON [dbo].[SIAPDU_RESE_CITA]
([CODI_EMPL]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_ESTA_RESE_CITA] ON [dbo].[SIAPDU_RESE_CITA]
([CODI_ESTA_RESE_CITA]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_MEDI_RESE_CITA] ON [dbo].[SIAPDU_RESE_CITA]
([CODI_MEDI_RESE_CITA]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_SERV_SOLI] ON [dbo].[SIAPDU_SOLI]([CODI_CLIN],
[CODI_TIPO_SERV]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_DOCU_SOLI] ON [dbo].[SIAPDU_SOLI]
([CODI_TIPO_DOCU]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_TERA] ON [dbo].[SIAPDU_TERA_PACI]
([CODI_TIPO_TERA]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_CONS_MEDI_TERA] ON [dbo].[SIAPDU_TERA_PACI]
([CODI_PACI], [CODI_TIPO_SERV], [CORR_CITA]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_EMPL_TERA] ON [dbo].[SIAPDU_TERA_PACI]
([CODI_EMPL]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_CLIN_TIPO_SERV] ON [dbo].[SIAPDU_TIPO_SERV_CLIN]
([CODI_CLIN]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_SERV_CLIN] ON [dbo].[SIAPDU_TIPO_SERV_CLIN]
([CODI_TIPO_SERV]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_PACI_VIVI] ON [dbo].[SIAPDU_VIVI_PACI]([CODI_PACI]) ON
[PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_VIVI] ON [dbo].[SIAPDU_VIVI_PACI]
([CODI_TIPO_VIVI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_PAGO_VIVI] ON [dbo].[SIAPDU_VIVI_PACI]
([CODI_TIPO_PAGO_VIVI]) ON [PRIMARY]
GO
```

```
CREATE INDEX [FK_TIPO_APAR_TIPO_MEDI] ON [dbo].
[SIAPDU_TIPO_MEDI_APAR]([CODI_TIPO_APAR]) ON [PRIMARY]
GO
```

```
ALTER TABLE [dbo].[SIAPDU_ANTE_SALU] ADD
    CONSTRAINT [FK_SIAPDU_A_PACI_ANTE_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    )
GO
```

```
ALTER TABLE [dbo].[SIAPDU_APAR_PACI] ADD
    CONSTRAINT [FK_SIAPDU_A_CONSI_APA_SIAPDU_C] FOREIGN KEY
```

```

(
    [CODI_PACI],
    [CODI_TIPO_SERV],
    [CORR_CITA]
) REFERENCES [dbo].[SIAPDU_CONS] (
    [CODI_PACI],
    [CODI_TIPO_SERV],
    [CORR_CITA]
),
CONSTRAINT [FK_SIAPDU_A_ESTA_APAR_SIAPDU_E] FOREIGN KEY
(
    [CODI_ESTA_APAR]
) REFERENCES [dbo].[SIAPDU_ESTA_APAR] (
    [CODI_ESTA_APAR]
),
CONSTRAINT [FK_SIAPDU_A_TIPO_APAR_SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_APAR]
) REFERENCES [dbo].[SIAPDU_TIPO_APAR] (
    [CODI_TIPO_APAR]
)
)
GO

ALTER TABLE [dbo].[SIAPDU_ASPE_ACAD] ADD
CONSTRAINT [FK_SIAPDU_A_PACI_ACAD_SIAPDU_P] FOREIGN KEY
(
    [CODI_PACI]
) REFERENCES [dbo].[SIAPDU_PACI] (
    [CODI_PACI]
)
)
GO

ALTER TABLE [dbo].[SIAPDU_ASPE_LABO] ADD
CONSTRAINT [FK_SIAPDU_A_PACI_LABO_SIAPDU_P] FOREIGN KEY
(
    [CODI_PACI]
) REFERENCES [dbo].[SIAPDU_PACI] (
    [CODI_PACI]
)
)
GO

ALTER TABLE [dbo].[SIAPDU_CALE_TERA] ADD
CONSTRAINT [FK_SIAPDU_C_TERA_CONT_SIAPDU_T] FOREIGN KEY
(
    [CODI_PACI],
    [CODI_TIPO_SERV],
    [CORR_CITA],

```

```

        [CORR_TERA_PACI]
    ) REFERENCES [dbo].[SIAPDU_TERA_PACI] (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_TERA_PACI]
    )
GO

ALTER TABLE [dbo].[SIAPDU_CITA] ADD
    CONSTRAINT [FK_SIAPDU_C_EXPE_CITA_SIAPDU_E] FOREIGN KEY
    (
        [CODI_PACI],
        [CODI_TIPO_SERV]
    ) REFERENCES [dbo].[SIAPDU_EXPE] (
        [CODI_PACI],
        [CODI_TIPO_SERV]
    )
GO

ALTER TABLE [dbo].[SIAPDU_CLIN] ADD
    CONSTRAINT [FK_SIAPDU_C_MUNI_CLIN_SIAPDU_M] FOREIGN KEY
    (
        [CODI_PAIS],
        [CODI_DEPA],
        [CODI_MUNI]
    ) REFERENCES [dbo].[SIAPDU_MUNI] (
        [CODI_PAIS],
        [CODI_DEPA],
        [CODI_MUNI]
    )
GO

ALTER TABLE [dbo].[SIAPDU_CONS] ADD
    CONSTRAINT [FK_SIAPDU_C_CITA_CONS_SIAPDU_C] FOREIGN KEY
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ) REFERENCES [dbo].[SIAPDU_CITA] (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ),
    CONSTRAINT [FK_SIAPDU_C_EMPL_CONS_SIAPDU_E] FOREIGN KEY
    (
        [CODI_EMPL]

```

```

        ) REFERENCES [dbo].[SIAPDU_EMPL] (
            [CODI_EMPL]
        )
GO

ALTER TABLE [dbo].[SIAPDU_CONT_COMU] ADD
    CONSTRAINT [FK_SIAPDU_C_PACI_CONT_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    )
GO

ALTER TABLE [dbo].[SIAPDU_CONT_PAGO] ADD
    CONSTRAINT [FK_SIAPDU_C_PACI_PAGO_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    ),
    CONSTRAINT [FK_SIAPDU_C_PAGO_SERV_SIAPDU_S] FOREIGN KEY
    (
        [CODI_SERV_PAGO]
    ) REFERENCES [dbo].[SIAPDU_SERV_PAGO] (
        [CODI_SERV_PAGO]
    )
GO

ALTER TABLE [dbo].[SIAPDU_CUES_EVAL] ADD
    CONSTRAINT [FK_SIAPDU_C_TIPO_TERA_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_TERA]
    ) REFERENCES [dbo].[SIAPDU_TIPO_TERA] (
        [CODI_TIPO_TERA]
    )
GO

ALTER TABLE [dbo].[SIAPDU_DEPA] ADD
    CONSTRAINT [FK_SIAPDU_D_PAIS_DEPA_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PAIS]
    ) REFERENCES [dbo].[SIAPDU_PAIS] (
        [CODI_PAIS]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_DINA_FAMI] ADD
    CONSTRAINT [FK_SIAPDU_D_PACI_DINA_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_DOCU_PACI] ADD
    CONSTRAINT [FK_SIAPDU_D_PACI_DOCU_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    ),
    CONSTRAINT [FK_SIAPDU_D_TIPO_DOCU_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_DOCU]
    ) REFERENCES [dbo].[SIAPDU_TIPO_DOCU] (
        [CODI_TIPO_DOCU]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_EGRE_PACI] ADD
    CONSTRAINT [FK_SIAPDU_E_PACI_EGRE_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    ),
    CONSTRAINT [FK_SIAPDU_E_TIPO_EGRE_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_EGRE]
    ) REFERENCES [dbo].[SIAPDU_TIPO_EGRE] (
        [CODI_TIPO_EGRE]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_EMPL] ADD
    CONSTRAINT [FK_SIAPDU_E_PUES_TRAB_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PUES_TRAB]
    ) REFERENCES [dbo].[SIAPDU_PUES_TRAB] (
        [CODI_PUES_TRAB]
    ),
    CONSTRAINT [FK_SIAPDU_E_TIPO_SERV_SIAPDU_T] FOREIGN KEY

```

```

(
    [CODI_CLIN],
    [CODI_TIPO_SERV]
) REFERENCES [dbo].[SIAPDU_TIPO_SERV_CLIN] (
    [CODI_CLIN],
    [CODI_TIPO_SERV]
)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_ESPE_EMPL] ADD
    CONSTRAINT [FK_SIAPDU_E_EMPL_ESPE_SIAPDU_E] FOREIGN KEY
    (
        [CODI_EMPL]
    ) REFERENCES [dbo].[SIAPDU_EMPL] (
        [CODI_EMPL]
    ),
    CONSTRAINT [FK_SIAPDU_E_ESPE_ESPE_SIAPDU_E] FOREIGN KEY
    (
        [CODI_ESPE]
    ) REFERENCES [dbo].[SIAPDU_ESPE] (
        [CODI_ESPE]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_ESTU_SOC] ADD
    CONSTRAINT [FK_SIAPDU_E_PACI_ESTU_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    ),
    CONSTRAINT [FK_SIAPDU_E_RANG_ECON_SIAPDU_C] FOREIGN KEY
    (
        [CODI_RANG_ECON]
    ) REFERENCES [dbo].[SIAPDU_CLAS_RANG_ECON] (
        [CODI_RANG_ECON]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_EVAL_PACI] ADD
    CONSTRAINT [FK_SIAPDU_E_CUEST_EVA_SIAPDU_C] FOREIGN KEY
    (
        [CODI_TIPO_TERA],
        [CODI_PREG]
    ) REFERENCES [dbo].[SIAPDU_CUES_EVAL] (
        [CODI_TIPO_TERA],
        [CODI_PREG]
    )

```

```

),
CONSTRAINT [FK_SIAPDU_E_TERA_EVAL_SIAPDU_T] FOREIGN KEY
(
    [CODI_PACI],
    [CODI_TIPO_SERV],
    [CORR_CITA],
    [CORR_TERA_PACI]
) REFERENCES [dbo].[SIAPDU_TERA_PACI] (
    [CODI_PACI],
    [CODI_TIPO_SERV],
    [CORR_CITA],
    [CORR_TERA_PACI]
)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_EXPE] ADD
CONSTRAINT [FK_SIAPDU_E_PACI_EXPE_SIAPDU_P] FOREIGN KEY
(
    [CODI_PACI]
) REFERENCES [dbo].[SIAPDU_PACI] (
    [CODI_PACI]
),
CONSTRAINT [FK_SIAPDU_E_TIPO_SER__SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_SERV]
) REFERENCES [dbo].[SIAPDU_TIPO_SERV] (
    [CODI_TIPO_SERV]
)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_FAMI_PACI] ADD
CONSTRAINT [FK_SIAPDU_F_PACI_FAMI_SIAPDU_P] FOREIGN KEY
(
    [CODI_PACI]
) REFERENCES [dbo].[SIAPDU_PACI] (
    [CODI_PACI]
),
CONSTRAINT [FK_SIAPDU_F_TIPO_PARE_SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_PARE]
) REFERENCES [dbo].[SIAPDU_TIPO_PARE] (
    [CODI_TIPO_PARE]
)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_HORA_CLIN] ADD
CONSTRAINT [FK_SIAPDU_H_HORA_CLIN_SIAPDU_C] FOREIGN KEY

```

```

        (
            [CODI_CLIN]
        ) REFERENCES [dbo].[SIAPDU_CLIN] (
            [CODI_CLIN]
        )
GO

ALTER TABLE [dbo].[SIAPDU_HORA_EMPL] ADD
    CONSTRAINT [FK_SIAPDU_H_HORA_EMPL_SIAPDU_E] FOREIGN KEY
    (
        [CODI_EMPL]
    ) REFERENCES [dbo].[SIAPDU_EMPL] (
        [CODI_EMPL]
    )
GO

ALTER TABLE [dbo].[SIAPDU_INGR_PACI] ADD
    CONSTRAINT [FK_SIAPDU_I_PACI_INGR_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    ),
    CONSTRAINT [FK_SIAPDU_I_TIPO_INGR_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_INGR]
    ) REFERENCES [dbo].[SIAPDU_TIPO_INGR] (
        [CODI_TIPO_INGR]
    )
GO

ALTER TABLE [dbo].[SIAPDU_INTE_VOCA_PACI] ADD
    CONSTRAINT [FK_SIAPDU_I_PACI_INTE_SIAPDU_P] FOREIGN KEY
    (
        [CODI_PACI]
    ) REFERENCES [dbo].[SIAPDU_PACI] (
        [CODI_PACI]
    ),
    CONSTRAINT [FK_SIAPDU_I_TIPO_INTE_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_INTE_VOCA]
    ) REFERENCES [dbo].[SIAPDU_TIPO_INTE_VOCA] (
        [CODI_TIPO_INTE_VOCA]
    )
GO

ALTER TABLE [dbo].[SIAPDU_MEDI_APAR] ADD

```

```

CONSTRAINT [FK_SIAPDU_M_APAR_MEDI_SIAPDU_A] FOREIGN KEY
(
    [CODI_PACI],
    [CODI_TIPO_SERV],
    [CORR_CITA],
    [CORR_APAR_PACI]
) REFERENCES [dbo].[SIAPDU_APAR_PACI] (
    [CODI_PACI],
    [CODI_TIPO_SERV],
    [CORR_CITA],
    [CORR_APAR_PACI]
),
CONSTRAINT [FK_SIAPDU_M_TIPO_MEDI_SIPADU_T] FOREIGN KEY
(
    [CODI_TIPO_APAR],
    [CODI_TIPO_MEDI]
) REFERENCES [dbo].[SIPADU_TIPO_MEDI_APAR] (
    [CODI_TIPO_APAR],
    [CODI_TIPO_MEDI]
)
GO

ALTER TABLE [dbo].[SIAPDU_MUNI] ADD
CONSTRAINT [FK_SIAPDU_M_DEPA_MUNI_SIAPDU_D] FOREIGN KEY
(
    [CODI_PAIS],
    [CODI_DEPA]
) REFERENCES [dbo].[SIAPDU_DEPA] (
    [CODI_PAIS],
    [CODI_DEPA]
)
GO

ALTER TABLE [dbo].[SIAPDU_OBST_LABO_PACI] ADD
CONSTRAINT [FK_SIAPDU_O_PACI_OBST_SIAPDU_P] FOREIGN KEY
(
    [CODI_PACI]
) REFERENCES [dbo].[SIAPDU_PACI] (
    [CODI_PACI]
),
CONSTRAINT [FK_SIAPDU_O_TIPO_OBST_SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_OBST_LABO]
) REFERENCES [dbo].[SIAPDU_TIPO_OBST_LABO] (
    [CODI_TIPO_OBST_LABO]
)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_OPCI_PERF_USUA] ADD
    CONSTRAINT [FK_SIAPDU_O_OPCI_PERF_SIAPDU_O] FOREIGN KEY
    (
        [CODI_OPCI_SIST]
    ) REFERENCES [dbo].[SIAPDU_OPCI_SIST] (
        [CODI_OPCI_SIST]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_PRUE_APAR] ADD
    CONSTRAINT [FK_SIAPDU_P_APAR_PRUE_SIAPDU_A] FOREIGN KEY
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_APAR_PACI]
    ) REFERENCES [dbo].[SIAPDU_APAR_PACI] (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA],
        [CORR_APAR_PACI]
    ),
    CONSTRAINT [FK_SIAPDU_P_TIPO_PRUE_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_PRUE_APAR]
    ) REFERENCES [dbo].[SIAPDU_TIPO_PRUE_APAR] (
        [CODI_TIPO_PRUE_APAR]
    )
GO

```

```

ALTER TABLE [dbo].[SIAPDU_REFE_OTRA_INST] ADD
    CONSTRAINT [FK_SIAPDU_R_CONS_MEDI_SIAPDU_C] FOREIGN KEY
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ) REFERENCES [dbo].[SIAPDU_CONS] (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ),
    CONSTRAINT [FK_SIAPDU_R_OTRA_INST_SIAPDU_O] FOREIGN KEY
    (
        [CODI_INST_MEDI]
    ) REFERENCES [dbo].[SIAPDU_OTRA_INST_MEDI] (
        [CODI_INST_MEDI]
    )

```

```

)
GO

ALTER TABLE [dbo].[SIAPDU_SOLI] ADD
    CONSTRAINT [FK_SIAPDU_S_TIPO_DOCU_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_DOCU]
    ) REFERENCES [dbo].[SIAPDU_TIPO_DOCU] (
        [CODI_TIPO_DOCU]
    ),
    CONSTRAINT [FK_SIAPDU_S_TIPO_SERV_SIAPDU_T] FOREIGN KEY
    (
        [CODI_CLIN],
        [CODI_TIPO_SERV]
    ) REFERENCES [dbo].[SIAPDU_TIPO_SERV_CLIN] (
        [CODI_CLIN],
        [CODI_TIPO_SERV]
    )
)
GO

ALTER TABLE [dbo].[SIAPDU_SUB_TIPO_TERA] ADD
    CONSTRAINT [FK_SIAPDU_S_TIPO_TERA_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_TERA]
    ) REFERENCES [dbo].[SIAPDU_TIPO_TERA] (
        [CODI_TIPO_TERA]
    )
)
GO

ALTER TABLE [dbo].[SIAPDU_TERA_PACI] ADD
    CONSTRAINT [FK_SIAPDU_T_CONS_MEDI_SIAPDU_C] FOREIGN KEY
    (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ) REFERENCES [dbo].[SIAPDU_CONS] (
        [CODI_PACI],
        [CODI_TIPO_SERV],
        [CORR_CITA]
    ),
    CONSTRAINT [FK_SIAPDU_T_EMPL_TERA_SIAPDU_E] FOREIGN KEY
    (
        [CODI_EMPL]
    ) REFERENCES [dbo].[SIAPDU_EMPL] (
        [CODI_EMPL]
    ),
    CONSTRAINT [FK_SIAPDU_T_SUB_TIPO_SIAPDU_S] FOREIGN KEY

```

```

(
    [CODI_TIPO_TERA],
    [CODI_SUB_TIPO_TERA]
) REFERENCES [dbo].[SIAPDU_SUB_TIPO_TERA] (
    [CODI_TIPO_TERA],
    [CODI_SUB_TIPO_TERA]
),
CONSTRAINT [FK_SIAPDU_T_TIPO_TERA_SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_TERA]
) REFERENCES [dbo].[SIAPDU_TIPO_TERA] (
    [CODI_TIPO_TERA]
)
)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_TIPO_SERV_CLIN] ADD
CONSTRAINT [FK_SIAPDU_T_CLIN_TIPO_SIAPDU_C] FOREIGN KEY
(
    [CODI_CLIN]
) REFERENCES [dbo].[SIAPDU_CLIN] (
    [CODI_CLIN]
),
CONSTRAINT [FK_SIAPDU_T_TIPO_SERV_SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_SERV]
) REFERENCES [dbo].[SIAPDU_TIPO_SERV] (
    [CODI_TIPO_SERV]
)
)
GO

```

```

ALTER TABLE [dbo].[SIAPDU_VIVI_PACI] ADD
CONSTRAINT [FK_SIAPDU_V_PACI_VIVI_SIAPDU_P] FOREIGN KEY
(
    [CODI_PACI]
) REFERENCES [dbo].[SIAPDU_PACI] (
    [CODI_PACI]
),
CONSTRAINT [FK_SIAPDU_V_TIPO_PAGO_SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_PAGO_VIVI]
) REFERENCES [dbo].[SIAPDU_TIPO_PAGO_VIVI] (
    [CODI_TIPO_PAGO_VIVI]
),
CONSTRAINT [FK_SIAPDU_V_TIPO_VIVI_SIAPDU_T] FOREIGN KEY
(
    [CODI_TIPO_VIVI]
) REFERENCES [dbo].[SIAPDU_TIPO_VIVI] (

```

```

        [CODI_TIPO_VIVI]
    )
GO

ALTER TABLE [dbo].[SIPADU_TIPO_MEDI_APAR] ADD
    CONSTRAINT [FK_SIPADU_T_TIPO_APAR_SIAPDU_T] FOREIGN KEY
    (
        [CODI_TIPO_APAR]
    ) REFERENCES [dbo].[SIAPDU_TIPO_APAR] (
        [CODI_TIPO_APAR]
    )
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS OFF
GO

```

```

/*****
/***** PROCEDIMIENTOS ALMACENADOS DE LA BASE DE DATOS *****/
/*****

```

```

/***** Object: Stored Procedure dbo.SIAPDP_CREA_EXPE_CONF_CITA Script
Date: 24/07/2002 07:44:44 pm *****/

```

```

CREATE PROCEDURE SIAPDP_CREA_EXPE_CONF_CITA

```

```

-- Parametros:

```

```

    @p_codi_clin          int,
    @p_codi_tipo_serv     int,
    @p_codi_soli          int,
    @p_fech_cita          datetime,
    @resp                 int output

```

```

AS

```

```

-- Procedimiento para Registrar los datos de un Nuevo Paciente y su Expediente
-- a partir de los datos de Solicitante (parametros @p_codi_clin, @p_codi_tipo_serv,
@p_codi_soli).

```

```

-- este procedimiento ademas registra la cita de un paciente.

```

```

-- Creacion: Giovanni Ramirez 24/07/2002 (Ultima modificacion 09/09/2002).

```

```

-- Declaracion de Variables y Tablas Temporales:

```

```

declare @codi_tipo_docu    int,
        @nomb_soli         varchar(100),
        @apel_soli         varchar(100),
        @fech_naci_soli    datetime,
        @nume_docu_iden    varchar(50),
        @dire_resi_soli    varchar(250),
        @tele_resi_soli    char(7),

```

```
@codi_paci_soli    int,  
@codi_nuev_paci   int,  
@nume_expe        int,  
@corr_cita        int
```

-- PARTE I: CREACION DEL EXPEDIENTE DE UN PACIENTE:

-- Toma los Datos del Solicitante:

```
select @codi_tipo_docu = CODI_TIPO_DOCU,  
       @nomb_soli     = NOMB_SOLI,  
       @apel_soli    = APEL_SOLI,  
       @fech_naci_soli = FECH_NACI_SOLI,  
       @nume_docu_iden = NUME_DOCU_IDEN,  
       @dire_resi_soli = DIRE_RESI_SOLI,  
       @tele_resi_soli = TELE_RESI_SOLI,  
       @codi_paci_soli = CODI_PACI_SOLI  
from SIAPDU_SOLI  
where CODI_CLIN = @p_codi_clin  
and CODI_TIPO_SERV = @p_codi_tipo_serv  
and CODI_SOLI = @p_codi_soli
```

-- Verifica si el solicitante ya esta registrado como paciente

IF (@codi_paci_soli is not null) and (@codi_paci_soli > 0)

-- Si el solicitante ya es paciente se tomara el codigo de paciente:

```
Begin  
  select @codi_nuev_paci = @codi_paci_soli  
End
```

ELSE

-- Si el solicitante no es paciente:

Begin

-- Registro de los datos del solicitante como nuevo paciente:

-- en el trigger de insercion se generara el codigo del nuevo paciente

```
insert into SIAPDU_PACI (CODI_PACI, NOMB_PACI, APEL_PACI, SEXO_PACI,  
ESTA_CIVI_PACI, LUGA_NACI_PACI, FECH_NACI_PACI, ESTA_PACI)  
values(0, @nomb_soli, @apel_soli, 'M', 'S', 1, @fech_naci_soli, 1)
```

-- Busqueda del codigo del nuevo paciente:

```
select @codi_nuev_paci = CODI_PACI  
from SIAPDU_PACI  
where NOMB_PACI = @nomb_soli  
and APEL_PACI = @apel_soli  
and convert(char(8), FECH_NACI_PACI, 112) = convert(char(8),  
@fech_naci_soli, 112)
```

-- Registro del Expediente de un Paciente:

-- en el trigger de insercion se generara el codigo del nuevo expediente

```

insert into SIAPDU_EXPE (CODI_PACI, CODI_TIPO_SERV, NUME_EXPE,
FECH_APER_EXPE, ESTA_EXPE)
values(@codi_nuev_paci, @p_codi_tipo_serv, 0, getdate(), 1)

-- Insercion los datos de la Vivienda del Paciente:
insert into SIAPDU_VIVI_PACI (CODI_PACI, CODI_VIVI_PACI,
CODI_TIPO_VIVI, CODI_TIPO_PAGO_VIVI,
DIRE_VIVI_PACI, AREA_VIVI, TELE_VIVI_PACI,
OBSE_VIVI_PACI)
values(@codi_nuev_paci, 1, 1, 1, @dire_resi_soli, 'U', @tele_resi_soli, null)

-- Insercion los datos del Documento del Paciente:
insert into SIAPDU_DOCU_PACI (CODI_PACI, CODI_TIPO_DOCU,
NUME_DOCU, LUGA_EXPE, FECH_EXPE)
values(@codi_nuev_paci, @codi_tipo_docu, @nume_docu_iden, " ", getdate())

-- Actualizacion de la Tabla Solicitante con elCodigo del Paciente:
update SIAPDU_SOLI
set CODI_PACI_SOLI = @codi_nuev_paci
where CODI_CLIN = @p_codi_clin
and CODI_TIPO_SERV = @p_codi_tipo_serv
and CODI_SOLI = @p_codi_soli
End

-- Parte II: CONFIRMACION DE LA CITA DE UN PACIENTE:

-- Si se creo el expediente o si este ya existe, se confirmara la cita, y se devolvera
1:
-- si fallo la creacion de expediente, se enviara un cero, indicando error
IF ((select count(*) from SIAPDU_PACI where CODI_PACI = @codi_nuev_paci) = 0
or
(select count(*) from SIAPDU_EXPE where CODI_PACI = @codi_nuev_paci and
CODI_TIPO_SERV = @p_codi_tipo_serv) = 0)

select @resp = 0
ELSE
Begin
select @corr_cita = isnull(max(CORR_CITA),0) + 1
from SIAPDU_CITA
where CODI_PACI = @codi_nuev_paci
and CODI_TIPO_SERV = @p_codi_tipo_serv

insert into SIAPDU_CITA (CODI_PACI, CODI_TIPO_SERV, CORR_CITA,
FECH_HORA_CITA, OBSE_CITA, ESTA_CITA)
values(@codi_nuev_paci, @p_codi_tipo_serv, @corr_cita, @p_fech_cita, null, 1)

-- Actualizacion del Estatus de la Reservacion de Cita:

```

```
update SIAPDU_RESE_CITA
  set CODI_ESTA_RESE_CITA = 2
where CODI_CLIN = @p_codi_clin
  and CODI_TIPO_SERV = @p_codi_tipo_serv
  and CODI_SOLI = @p_codi_soli
  and FECH_HORA_CITA_RESE = @p_fech_cita
```

```
select @resp = 1
End
```

RETURN

GO

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

/***** Object: Stored Procedure dbo.SIAPDP_DISP_HORA_CITA Script Date:
10/07/2002 07:44:45 pm *****/

```
CREATE PROCEDURE SIAPDP_DISP_HORA_CITA
```

```
-- Parametros:
```

```
@codi_clin      int,  
@codi_tipo_serv int,  
@fech_cita      datetime
```

AS

```
-- Procedimiento para Buscar la Disponibilidad de Horarios para Citas de Consultas  
medicas,
```

```
-- segun los horarios de los doctores y los parametros recibidos: Clinica, Tipo de  
Servicio
```

```
-- y la fecha deseada para la cita (Las disponibilidades se guardaran temporalmente  
en SIADPU_TEMP_SIP_HORA)
```

```
-- Si el procedimiento se ejecuto con exito devolvera 1.
```

```
-- Creacion: Giovanni Ramirez 10/07/2002 (Ultima modificacion 09/09/2002).
```

```
-- Declaracion de Variables y Tablas Temporales:
```

```
declare @nomb_dia      varchar(20),  
        @nume_dia      int,  
        @dura_cons     int,  
        @hora_disp     datetime,  
        @c_codi_empl   int,  
        @c_hora_entr   datetime,  
        @c_hora_sali   datetime,  
        @c_hora_inic_rece datetime,
```

```

@c_hora_fina_rece datetime,
  @conf_exec          int,
@corr_temp           int

-- Limpia la Tabla SIAPDU_TEMP_DISP_HORA
-- en esta tabla se guardaran las disponibilidades con solo los
-- registros generados en cada ejecucion del procedimiento:
delete SIAPDU_TEMP_DISP_HORA

-- Verifica que se haya borrado la tabla temporal, sino ya no se ejecuta el resto:
If ((select count(*) from SIAPDU_TEMP_DISP_HORA) > 0)
begin
  select @conf_exec = 0
  select @conf_exec
  return
end

-- Formato con el que se manejara la fecha de la cita (sin horas):
select @fech_cita = convert(char(8), @fech_cita, 112)

-- Calcula el Dia Actual de la Semana:
-- en la funcion datepart lunes = 1
select @nume_dia = datepart(weekday, @fech_cita)

-- Duracion de una Consulta Medica:
select @dura_cons = DURA_CONS
  from SIAPDU_HORA_CLIN
 where ESTA_HORA = 1
  and CODI_CLIN = @codi_clin
  and DIA_HORA = @nume_dia

-- Cursor para crear la Tabla de Horarios de un Empleado:
Declare C_HORA_EMPL cursor for
select e.CODI_EMPL,
       h.HORA_ENTR_EMPL,
       HORA_SALI_EMPL,
       HORA_INIC_RECE,
       HORA_FINA_RECE
  from SIAPDU_EMPL e,
       SIAPDU_HORA_EMPL h
 where e.CODI_EMPL = h.CODI_EMPL
  and h.ESTA_HORA_EMPL = 1
  and e.CODI_CLIN = @codi_clin
  and e.CODI_TIPO_SERV = @codi_tipo_serv
  and h.DIA_HORA_EMPL = @nume_dia

```

```

Open C_HORA_EMPL
Fetch Next From C_HORA_EMPL
Into @c_codi_empl, @c_hora_entr, @c_hora_sali, @c_hora_inic_rece,
@c_hora_fina_rece

select @corr_temp = 1

WHILE @@FETCH_STATUS = 0
BEGIN
-- Primera Hora a Insertar en la Tabla de Horarios de un Empleado:
select @hora_disp = @c_hora_entr

-- Insercion de Registros en la Tabla Temporal:
While convert(char(5), @c_hora_sali, 108) > convert(char(5), @hora_disp, 108)
Begin
-- Si la Hora no es parte del horario de receso del empleado => insertar como
hora disponible:
if (convert(char(5), @hora_disp, 108) < convert(char(5), @c_hora_inic_rece,
108))
or (convert(char(5), @hora_disp, 108) >= convert(char(5),
@c_hora_fina_rece, 108))
insert into SIAPDU_TEMP_DISP_HORA (CORR_TEMP, CODI_EMPL,
HORA, DISP)
values(@corr_temp, @c_codi_empl, convert(char(5), @hora_disp, 108), 0)

-- Nueva Hora a ser evaluada: Sumar a la hora insertada la duracion de la
consulta:
select @hora_disp = (dateadd(minute, @dura_cons, convert(char(5),
@hora_disp, 108)))

-- Nuevo Correlativo:
select @corr_temp = @corr_temp + 1
End

Fetch Next From C_HORA_EMPL
Into @c_codi_empl, @c_hora_entr, @c_hora_sali, @c_hora_inic_rece,
@c_hora_fina_rece
END
-- Eliminacion del cursor:
Close C_HORA_EMPL
Deallocate C_HORA_EMPL

-- Actualizacion de la Tabla de Disponibilidades de Citas:
-- Se colocará un 1 a las citas reservadas
Update SIAPDU_TEMP_DISP_HORA
set DISP = 1
from SIAPDU_TEMP_DISP_HORA d, SIAPDU_RESE_CITA r

```

```

where r.CODI_CLIN = @codi_clin
and r.CODI_TIPO_SERV = @codi_tipo_serv
and d.CODI_EMPL = r.CODI_EMPL
and convert(char(8), r.FECH_HORA_CITA_RESE, 112) = @fech_cita
and d.HORA = convert(char(5), r.FECH_HORA_CITA_RESE, 108)

-- Si el procedimiento se ejecuto con exito devolvera 1:
If ((select count(*) from SIAPDU_TEMP_DISP_HORA) > 0)
begin
select @conf_exec = 1
select @conf_exec
end
Else
begin
select @conf_exec = 0
select @conf_exec
end

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Stored Procedure dbo.SIAPDP_GENE_CALE_TERA Script Date:
10/09/2002 07:44:45 pm *****/
CREATE PROCEDURE SIAPDP_GENE_CALE_TERA
-- Parametros:
@codi_paci int,
@codi_tipo_serv int,
@corr_cita int,
@corr_tera int

AS
-- Procedimiento para generar el calendario de terapias de un paciente en base a la
-- prescripcion del doctor dada en la consulta medica
-- La llamada de este procedimiento se hace en el trigger de insercion de la tabla
SIAPDU_TERA_PACI
-- Es decir que cuando el doctor prescribe una terapia en la consulta medica,
entonces se genera el calendario.
-- Creacion: Giovanni Ramirez 10/09/2002.

-- Declaracion de Variables y Tablas Temporales:

```

```

declare @cant_tera          int,
        @dias_frec_tera    int,
        @fech_tera         datetime,
        @dia                int,
        @corr_cale         int

-- Seleccion de los datos generales de la terapia prescrita por el medico:
select @cant_tera = CANT_TERA,
       @dias_frec_tera = DIAS_FREC_TERA,
       @fech_tera    = FECH_INIC_TERA
from SIAPDU_TERA_PACI
where CODI_PACI = @codi_paci
and CODI_TIPO_SERV = @codi_tipo_serv
and CORR_CITA = @corr_cita
and CORR_TERA_PACI = @corr_tera

-- Formato con el que se manejara la fecha de las terapias (yyyymmdd):
select @fech_tera = convert(char(8), @fech_tera, 112)

-- Inicializa la variable que contendra la llave de la calendarizacion:
select @corr_cale = 1

-- Generacion de la Calendarizacion:
WHILE @cant_tera > 0
BEGIN
    -- Calcula el Dia de la Semana de la fecha de terapia:
    -- en la funcion datepart lunes = 1
    select @dia = datepart(weekday, @fech_tera)

    -- Solamente se podran dar terapias de lunes a viernes:
    If @dia = 6
        select dateadd(day, 2, @fech_tera)
    Else if @dia = 7
        select dateadd(day, 1, @fech_tera)

    -- Insercion de la Calendarizacion en la tabla SIAPDU_CALE_TERA
    -- Si es la primera terapia el tipo de atencion sera: Evaluacion (E)
    -- Si no es la primera terapia el tipo de atencion sera: Subsecuente (S)
    If @corr_cale = 1
        insert into SIAPDU_CALE_TERA (CODI_PACI, CODI_TIPO_SERV,
CORR_CITA, CORR_TERA_PACI, CORR_CALE, FECH_HORA_TERA,
TIPO_ATEN, DIAG_TERA, OBSE_TERA, ESTA_TERA)
            values(@codi_paci, @codi_tipo_serv, @corr_cita, @corr_tera, @corr_cale,
@fech_tera, 'E', '', '', 'E')
    Else

```

```

insert into SIAPDU_CALE_TERA (CODI_PACI, CODI_TIPO_SERV,
CORR_CITA, CORR_TERA_PACI, CORR_CALE, FECH_HORA_TERA,
TIPO_ATEN, DIAG_TERA, OBSE_TERA, ESTA_TERA)
values(@codi_paci, @codi_tipo_serv, @corr_cita, @corr_tera, @corr_cale,
@fech_tera, 'S', ", ", 'E')

```

```

-- Calcula la Fecha de la Proxima Terapia:
select @fech_tera = @fech_tera + @dias_frec_tera

```

```

-- Calcula la llave de la calendarizacion:
select @corr_cale = @corr_cale + 1

```

```

-- Actualiza el contador de cantidades de terapias:
select @cant_tera = @cant_tera - 1
END

```

```

RETURN

```

```

GO

```

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

```

/*****
/***** TRIGGERS DE LA BASE DE DATOS *****/
/*****

```

```

/***** Object: Trigger dbo.SIAPDT_ADIC_CLAS_RANG_ECON Script Date:
18/09/2002 07:44:45 pm *****/

```

```

create trigger SIAPDT_ADIC_CLAS_RANG_ECON
on SIAPDU_CLAS_RANG_ECON

```

```

for insert

```

```

as

```

```

-- Trigger de Adicion de la Tabla SIAPDU_CLAS_RANG_ECON

```

```

declare @nume_regi smallint,
@llave smallint

```

```

select @nume_regi = @@rowcount

```

```

if @nume_regi = 0

```

```

return

```

```

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_CLAS_RANG_ECON" - No es permitido
adicionar varios registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_RANG_ECON),0) + 1 from
SIAPDU_CLAS_RANG_ECON holdlock)
update SIAPDU_CLAS_RANG_ECON set CODI_RANG_ECON = @llave where
CODI_RANG_ECON = 0

```

RETURN

GO

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

/***** Object: Trigger dbo.SIAPDT_ADIC_CLIN Script Date: 18/09/2002 07:44:45
pm *****/

```

create trigger SIAPDT_ADIC_CLIN
on SIAPDU_CLIN

```

for insert

as

-- Trigger de Adicion de la Tabla SIAPDU_CLIN

```

declare @nume_regi    smallint,
        @llave        smallint

```

```

select @nume_regi = @@rowcount

```

```

if @nume_regi = 0
return

```

```

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_CLIN" - No es permitido adicionar varios
registros a la vez'
    rollback transaction
    return
end

```

```

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_CLIN),0) + 1 from SIAPDU_CLIN holdlock)
update SIAPDU_CLIN set CODI_CLIN = @llave where CODI_CLIN = 0

```

RETURN

GO

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

/***** Object: Trigger dbo.SIAPDT_ADIC_EMPL Script Date: 18/09/2002 07:44:45

pm *****/

```

create trigger SIAPDT_ADIC_EMPL
on SIAPDU_EMPL

```

```

for insert

```

```

as

```

```

-- Trigger de Adicion de la Tabla SIAPDU_EMPL

```

```

declare @nume_regi smallint,
        @llave smallint

```

```

select @nume_regi = @@rowcount

```

```

if @nume_regi = 0
return

```

```

if @nume_regi > 1
begin

```

```

    raiserror 30001 ' "SIAPDT_ADIC_EMPL" - No es permitido adicionar varios
registros a la vez'

```

```

    rollback transaction

```

```

    return

```

```

end

```

```

-- Genera la llave de la Tabla al momento de insertar el registro

```

```

-- esto con el fin de evitar problemas al tener usuarios concurrentes

```

```

select @llave = (select isnull(max(CODI_EMPL),0) + 1 from SIAPDU_EMPL
holdlock)

```

```

update SIAPDU_EMPL set CODI_EMPL = @llave where CODI_EMPL = 0

```

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/****** Object: Trigger dbo.SIAPDT_ADIC_ESPE Script Date: 18/09/2002 07:44:45
pm *****/

create trigger SIAPDT_ADIC_ESPE
on SIAPDU_ESPE

for insert

as

-- Trigger de Adicion de la Tabla SIAPDU_ESPE

declare @nume_regi smallint,
@llave smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
raiserror 30001 ' "SIAPDT_ADIC_ESPE" - No es permitido adicionar varios
registros a la vez'
rollback transaction
return
end

-- Genera la llave de la Tabla al momento de insertar el registro

-- esto con el fin de evitar problemas al tener usuarios concurrentes

select @llave = (select isnull(max(CODI_ESPE),0) + 1 from SIAPDU_ESPE
holdlock)

update SIAPDU_ESPE set CODI_ESPE = @llave where CODI_ESPE = 0

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```
/****** Object: Trigger dbo.SIAPDT_ADIC_ESTA_APAR   Script Date: 18/09/2002
07:44:45 pm *****/
```

```
create trigger SIAPDT_ADIC_ESTA_APAR
    on SIAPDU_ESTA_APAR
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_ESTA_APAR
declare @nume_regi    smallint,
        @llave        smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_ESTA_APAR" - No es permitido adicionar varios
registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_ESTA_APAR),0) + 1 from
SIAPDU_ESTA_APAR holdlock)

update SIAPDU_ESTA_APAR set CODI_ESTA_APAR = @llave where
CODI_ESTA_APAR = 0
```

```
RETURN
```

```
GO
```

```
SET QUOTED_IDENTIFIER OFF   SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER OFF   SET ANSI_NULLS ON
GO
```

```
/****** Object: Trigger dbo.SIAPDT_ADIC_ESTA_RESE_CITA   Script Date:
18/09/2002 07:44:45 pm *****/
```

```
create trigger SIAPDT_ADIC_ESTA_RESE_CITA
    on SIAPDU_ESTA_RESE_CITA
for insert
as
```

```

-- Trigger de Adicion de la Tabla SIAPDU_ESTA_RESE_CITA
declare @nume_regi    smallint,
        @llave        smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_ESTA_RESE_CITA" - No es permitido adicionar
varios registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_ESTA_RESE_CITA),0) + 1 from
SIAPDU_ESTA_RESE_CITA holdlock)
update SIAPDU_ESTA_RESE_CITA set CODI_ESTA_RESE_CITA = @llave where
CODI_ESTA_RESE_CITA = 0

RETURN

GO

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_EXPE  Script Date: 18/09/2002 07:44:46
pm *****/
create trigger SIAPDT_ADIC_EXPE
    on SIAPDU_EXPE
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_EXPE
declare @nume_regi    smallint,
        @llave        smallint,
        @codi_paci    int,
        @codi_tipo_serv int

select @nume_regi = @@rowcount

```

```

if @nume_regi = 0
return

if @nume_regi > 1
begin
raiserror 30001 ' "SIAPDT_ADIC_EXPE" - No es permitido adicionar varios
registros a la vez'
rollback transaction
return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @codi_paci = CODI_PACI,
       @codi_tipo_serv = CODI_TIPO_SERV
from inserted

select @llave = (select isnull(max(NUMEXPE),0) + 1 from SIAPDU_EXPE where
CODI_TIPO_SERV = @codi_tipo_serv)
update SIAPDU_EXPE set NUMEXPE = @llave where NUMEXPE = 0 and
CODI_PACI = @codi_paci and CODI_TIPO_SERV = @codi_tipo_serv

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_MEDI_RESE_CITA Script Date:
18/09/2002 07:44:46 pm *****/
create trigger SIAPDT_ADIC_MEDI_RESE_CITA
on SIAPDU_MEDI_RESE_CITA
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_MEDI_RESE_CITA
declare @nume_regi smallint,
       @llave smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

```

```

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_MEDI_RESE_CITA" - No es permitido adicionar
varios registros a la vez'
    rollback transaction
    return
end

```

```

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_MEDI_RESE_CITA),0) + 1 from
SIAPDU_MEDI_RESE_CITA holdlock)
update SIAPDU_MEDI_RESE_CITA set CODI_MEDI_RESE_CITA = @llave where
CODI_MEDI_RESE_CITA = 0

```

RETURN

GO

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

/***** Object: Trigger dbo.SIAPDT_ADIC_OTRA_INST_MEDI Script Date:
18/09/2002 07:44:46 pm *****/

```

create trigger SIAPDT_ADIC_OTRA_INST_MEDI
on SIAPDU_OTRA_INST_MEDI

```

```

for insert

```

```

as

```

```

-- Trigger de Adicion de la Tabla SIAPDU_OTRA_INST_MEDI

```

```

declare @nume_regi    smallint,
        @llave        smallint

```

```

select @nume_regi = @@rowcount

```

```

if @nume_regi = 0
return

```

```

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_OTRA_INST_MEDI" - No es permitido adicionar
varios registros a la vez'
    rollback transaction
    return
end

```

```

end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_INST_MEDI),0) + 1 from
SIAPDU_OTRA_INST_MEDI holdlock)
update SIAPDU_OTRA_INST_MEDI set CODI_INST_MEDI = @llave where
CODI_INST_MEDI = 0

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_PACI Script Date: 18/09/2002 07:44:46
pm *****/
create trigger SIAPDT_ADIC_PACI
on SIAPDU_PACI
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_PACI
declare @nume_regi smallint,
@llave smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
raiserror 30001 ' "SIAPDT_ADIC_PACI" - No es permitido adicionar varios
registros a la vez'
rollback transaction
return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_PACI),0) + 1 from SIAPDU_PACI holdlock)
update SIAPDU_PACI set CODI_PACI = @llave where CODI_PACI = 0

```

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_PAIS Script Date: 18/09/2002 07:44:46

pm *****/

CREATE trigger SIAPDT_ADIC_PAIS
on SIAPDU_PAIS

for insert

as

-- Trigger de Adicion de la Tabla SIAPDU_PAIS

declare @nume_regi smallint,
@llave smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin

raiserror 30001 ' "SIAPDT_ADIC_PAIS" - No es permitido adicionar varios
registros a la vez'

rollback transaction
return
end

-- Genera la llave de la Tabla al momento de insertar el registro

-- esto con el fin de evitar problemas al tener usuarios concurrentes

select @llave = (select isnull(max(CODI_PAIS),0) + 1 from SIAPDU_PAIS holdlock)

update SIAPDU_PAIS set CODI_PAIS = @llave where CODI_PAIS = 0

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

/***** Object: Trigger dbo.SIAPDT_ADIC_PUES_TRAB   Script Date: 18/09/2002
07:44:46 pm *****/
create trigger SIAPDT_ADIC_PUES_TRAB
    on SIAPDU_PUES_TRAB
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_PUES_TRAB
declare @nume_regi    smallint,
        @llave       smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_PUES_TRAB" - No es permitido adicionar varios
registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_PUES_TRAB),0) + 1 from
SIAPDU_PUES_TRAB holdlock)
update SIAPDU_PUES_TRAB set CODI_PUES_TRAB = @llave where
CODI_PUES_TRAB = 0

```

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

/***** Object: Trigger dbo.SIAPDT_ADIC_SERV_PAGO   Script Date: 18/09/2002
07:44:46 pm *****/
create trigger SIAPDT_ADIC_SERV_PAGO
    on SIAPDU_SERV_PAGO
for insert
as

```

```

-- Trigger de Adicion de la Tabla SIAPDU_SERV_PAGO
declare @nume_regi      smallint,
        @llave          smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_SERV_PAGO" - No es permitido adicionar
varios registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_SERV_PAGO),0) + 1 from
SIAPDU_SERV_PAGO holdlock)
update SIAPDU_SERV_PAGO set CODI_SERV_PAGO = @llave where
CODI_SERV_PAGO = 0

```

RETURN

GO

```

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

```

/***** Object: Trigger dbo.SIAPDT_ADIC_TERA_PACI Script Date: 18/09/2002
07:44:46 pm *****/

```

create trigger SIAPDT_ADIC_TERA_PACI
on SIAPDU_TERA_PACI

```

```

for insert
as

```

```

-- Trigger de Adicion de la Tabla SIAPDU_TERA_PACI

```

```

declare @nume_regi      smallint,
        @codi_paci      int,
        @codi_tipo_serv int,
        @corr_cita      int,
        @corr_tera      int

```

```

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_TERA_PACI" - No es permitido adicionar varios
registros a la vez'
    rollback transaction
    return
end

-- Toma la llave del registro ser insertado
select @codi_paci = CODI_PACI,
        @codi_tipo_serv = CODI_TIPO_SERV,
        @corr_cita = CORR_CITA,
        @corr_tera = CORR_TERA_PACI
from inserted

-- Genera la calendarizacion de la Terapia:
execute SIAPDP_GENE_CALE_TERA @codi_paci, @codi_tipo_serv, @corr_cita,
@corr_tera

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_APAR Script Date: 18/09/2002
07:44:46 pm *****/
create trigger SIAPDT_ADIC_TIPO_APAR
on SIAPDU_TIPO_APAR
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_APAR
declare @nume_regi smallint,
        @llave smallint

select @nume_regi = @@rowcount

if @nume_regi = 0

```

```

return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_TIPO_APAR" - No es permitido adicionar varios
registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_APAR),0) + 1 from
SIAPDU_TIPO_APAR holdlock)
update SIAPDU_TIPO_APAR set CODI_TIPO_APAR = @llave where
CODI_TIPO_APAR = 0

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_DOCU Script Date: 18/09/2002
07:44:46 pm *****/
create trigger SIAPDT_ADIC_TIPO_DOCU
on SIAPDU_TIPO_DOCU
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_DOCU
declare @nume_regi smallint,
        @llave smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_TIPO_DOCU" - No es permitido adicionar varios
registros a la vez'
    rollback transaction

```

```

    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_DOCU),0) + 1 from
SIAPDU_TIPO_DOCU holdlock)
update SIAPDU_TIPO_DOCU set CODI_TIPO_DOCU = @llave where
CODI_TIPO_DOCU = 0

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_EGRE Script Date: 18/09/2002
07:44:46 pm *****/
create trigger SIAPDT_ADIC_TIPO_EGRE
    on SIAPDU_TIPO_EGRE
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_EGRE
declare @nume_regi    smallint,
        @llave        smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_TIPO_EGRE" - No es permitido adicionar varios
registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_EGRE),0) + 1 from
SIAPDU_TIPO_EGRE holdlock)

```

```
update SIAPDU_TIPO_EGRE set CODI_TIPO_EGRE = @llave where  
CODI_TIPO_EGRE = 0
```

```
RETURN
```

```
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON  
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON  
GO
```

```
/****** Object: Trigger dbo.SIAPDT_ADIC_TIPO_INGR Script Date: 18/09/2002  
07:44:47 pm *****/
```

```
create trigger SIAPDT_ADIC_TIPO_INGR  
on SIAPDU_TIPO_INGR
```

```
for insert
```

```
as
```

```
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_INGR
```

```
declare @nume_regi smallint,  
        @llave smallint
```

```
select @nume_regi = @@rowcount
```

```
if @nume_regi = 0  
return
```

```
if @nume_regi > 1  
begin
```

```
raiserror 30001 ' "SIAPDT_ADIC_TIPO_INGR" - No es permitido adicionar varios  
registros a la vez'
```

```
rollback transaction  
return  
end
```

```
-- Genera la llave de la Tabla al momento de insertar el registro  
-- esto con el fin de evitar problemas al tener usuarios concurrentes
```

```
select @llave = (select isnull(max(CODI_TIPO_INGR),0) + 1 from  
SIAPDU_TIPO_INGR holdlock)  
update SIAPDU_TIPO_INGR set CODI_TIPO_INGR = @llave where  
CODI_TIPO_INGR = 0
```

```
RETURN
```

```
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
/****** Object: Trigger dbo.SIAPDT_ADIC_TIPO_INTE_VOCA Script Date:
18/09/2002 07:44:47 pm *****/
```

```
create trigger SIAPDT_ADIC_TIPO_INTE_VOCA
on SIAPDU_TIPO_INTE_VOCA
```

```
for insert
```

```
as
```

```
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_INTE_VOCA
```

```
declare @nume_regi smallint,
        @llave smallint
```

```
select @nume_regi = @@rowcount
```

```
if @nume_regi = 0
return
```

```
if @nume_regi > 1
begin
```

```
raiserror 30001 ' "SIAPDT_ADIC_TIPO_INTE_VOCA" - No es permitido adicionar
varios registros a la vez'
```

```
rollback transaction
```

```
return
```

```
end
```

```
-- Genera la llave de la Tabla al momento de insertar el registro
```

```
-- esto con el fin de evitar problemas al tener usuarios concurrentes
```

```
select @llave = (select isnull(max(CODI_TIPO_INTE_VOCA),0) + 1 from
SIAPDU_TIPO_INTE_VOCA holdlock)
```

```
update SIAPDU_TIPO_INTE_VOCA set CODI_TIPO_INTE_VOCA = @llave where
CODI_TIPO_INTE_VOCA = 0
```

```
RETURN
```

```
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_OBST_LABO   Script Date:
18/09/2002 07:44:47 pm *****/
create trigger SIAPDT_ADIC_TIPO_OBST_LABO
    on SIAPDU_TIPO_OBST_LABO
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_OBST_LABO
declare @nume_regi    smallint,
        @llave        smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_TIPO_OBST_LABO" - No es permitido adicionar
varios registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_OBST_LABO),0) + 1 from
SIAPDU_TIPO_OBST_LABO holdlock)
update SIAPDU_TIPO_OBST_LABO set CODI_TIPO_OBST_LABO = @llave where
CODI_TIPO_OBST_LABO = 0

RETURN

```

GO

```

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

```

```

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_PAGO_VIVI   Script Date:
18/09/2002 07:44:47 pm *****/
create trigger SIAPDT_ADIC_TIPO_PAGO_VIVI
    on SIAPDU_TIPO_PAGO_VIVI
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_PAGO_VIVI

```

```

declare @nume_regi    smallint,
        @llave        smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_TIPO_PAGO_VIVI" - No es permitido adicionar
varios registros a la vez'
    rollback transaction
    return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_PAGO_VIVI),0) + 1 from
SIAPDU_TIPO_PAGO_VIVI holdlock)
update SIAPDU_TIPO_PAGO_VIVI set CODI_TIPO_PAGO_VIVI = @llave where
CODI_TIPO_PAGO_VIVI = 0

```

RETURN

GO

```

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF  SET ANSI_NULLS ON
GO

```

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_PARE Script Date: 18/09/2002
07:44:47 pm *****/

```

create trigger SIAPDT_ADIC_TIPO_PARE
    on SIAPDU_TIPO_PARE
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_PARE
declare @nume_regi    smallint,
        @llave        smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

```

```

if @nume_regi > 1
begin
    raiserror 30001 ' "SIAPDT_ADIC_TIPO_PARE" - No es permitido adicionar varios
registros a la vez'
    rollback transaction
    return
end

```

```

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_PARE),0) + 1 from
SIAPDU_TIPO_PARE holdlock)
update SIAPDU_TIPO_PARE set CODI_TIPO_PARE = @llave where
CODI_TIPO_PARE = 0

```

RETURN

GO

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

```

```

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_PRUE_APAR Script Date:
18/09/2002 07:44:47 pm *****/

```

```

create trigger SIAPDT_ADIC_TIPO_PRUE_APAR
on SIAPDU_TIPO_PRUE_APAR

```

```

for insert

```

```

as

```

```

-- Trigger de Adicion de la Tabla SIAPDU_TIPO_PRUE_APAR

```

```

declare @nume_regi    smallint,
        @llave        smallint

```

```

select @nume_regi = @@rowcount

```

```

if @nume_regi = 0
return

```

```

if @nume_regi > 1
begin

```

```

    raiserror 30001 ' "SIAPDT_ADIC_TIPO_PRUE_APAR" - No es permitido adicionar
varios registros a la vez'
    rollback transaction
    return

```

```

end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_PRUE_APAR),0) + 1 from
SIAPDU_TIPO_PRUE_APAR holdlock)
update SIAPDU_TIPO_PRUE_APAR set CODI_TIPO_PRUE_APAR = @llave where
CODI_TIPO_PRUE_APAR = 0

RETURN

GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO

/***** Object: Trigger dbo.SIAPDT_ADIC_TIPO_SERV Script Date: 18/09/2002
07:44:47 pm *****/
create trigger SIAPDT_ADIC_TIPO_SERV
on SIAPDU_TIPO_SERV
for insert
as
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_SERV
declare @nume_regi smallint,
@llave smallint

select @nume_regi = @@rowcount

if @nume_regi = 0
return

if @nume_regi > 1
begin
raiserror 30001 ' "SIAPDT_ADIC_TIPO_SERV" - No es permitido adicionar varios
registros a la vez'
rollback transaction
return
end

-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
select @llave = (select isnull(max(CODI_TIPO_SERV),0) + 1 from
SIAPDU_TIPO_SERV holdlock)

```

```
update SIAPDU_TIPO_SERV set CODI_TIPO_SERV = @llave where  
CODI_TIPO_SERV = 0
```

```
RETURN
```

```
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON  
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON  
GO
```

```
/****** Object: Trigger dbo.SIAPDT_ADIC_TIPO_TERA Script Date: 18/09/2002  
07:44:47 pm *****/
```

```
create trigger SIAPDT_ADIC_TIPO_TERA  
on SIAPDU_TIPO_TERA
```

```
for insert
```

```
as
```

```
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_TERA
```

```
declare @nume_regi smallint,  
        @llave smallint
```

```
select @nume_regi = @@rowcount
```

```
if @nume_regi = 0  
return
```

```
if @nume_regi > 1  
begin  
    raiserror 30001 ' "SIAPDT_ADIC_TIPO_TERA" - No es permitido adicionar varios  
registros a la vez'  
    rollback transaction  
    return  
end
```

```
-- Genera la llave de la Tabla al momento de insertar el registro  
-- esto con el fin de evitar problemas al tener usuarios concurrentes  
select @llave = (select isnull(max(CODI_TIPO_TERA),0) + 1 from  
SIAPDU_TIPO_TERA holdlock)  
update SIAPDU_TIPO_TERA set CODI_TIPO_TERA = @llave where  
CODI_TIPO_TERA = 0
```

```
RETURN
```

```
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```

```
/****** Object: Trigger dbo.SIAPDT_ADIC_TIPO_VIVI Script Date: 18/09/2002
07:44:47 pm *****/
```

```
create trigger SIAPDT_ADIC_TIPO_VIVI
on SIAPDU_TIPO_VIVI
```

```
for insert
```

```
as
```

```
-- Trigger de Adicion de la Tabla SIAPDU_TIPO_VIVI
```

```
declare @nume_regi smallint,
        @llave smallint
```

```
select @nume_regi = @@rowcount
```

```
if @nume_regi = 0
return
```

```
if @nume_regi > 1
begin
```

```
raiserror 30001 ' "SIAPDT_ADIC_TIPO_VIVI" - No es permitido adicionar varios
registros a la vez'
```

```
rollback transaction
return
end
```

```
-- Genera la llave de la Tabla al momento de insertar el registro
-- esto con el fin de evitar problemas al tener usuarios concurrentes
```

```
select @llave = (select isnull(max(CODI_TIPO_VIVI),0) + 1 from
SIAPDU_TIPO_VIVI holdlock)
```

```
update SIAPDU_TIPO_VIVI set CODI_TIPO_VIVI = @llave where CODI_TIPO_VIVI
= 0
```

```
RETURN
```

```
GO
```

```
SET QUOTED_IDENTIFIER OFF SET ANSI_NULLS ON
GO
```