

UNIVERSIDAD DON BOSCO  
FACULTAD DE INGENIERÍA



MANUAL DEL PROGRAMADOR

# Sistema de Tutorías Estudiantiles (SITE)

PRESENTADO POR:  
DIEGO JOSÉ ESPINOZA CARRANZA  
HEISEL PATRICIA CALDERÓN GONZÁLEZ

Versión 1.0

# INDICE

<b>INTRODUCCIÓN</b>	<b>1</b>
<b>ESTRUCTURA DEL CÓDIGO</b>	<b>2</b>
CARPETA ADMINISTRACION	2
CARPETA AYUDA	4
CARPETA FXC	4
CARPETA SCRIPTS	7
CARPETA SRC/JPGRAPH_CACHE	7
CARPETA ARCHIVOS_CSV	7
CARPETA IMAGENES	8
CARPETA RAÍZ	8
<b>Función minitablas</b>	<b>12</b>
<b>Función preio</b>	<b>18</b>
<b>Función opciones</b>	<b>22</b>
<b>Función carnets</b>	<b>23</b>
<b>Funciones cambiar_carnet y cambiar_Usu</b>	<b>27</b>
<b>Función nuevoCiclo_Anio</b>	<b>30</b>
<b>Funciones relacionadas con los ciclos y meses</b>	<b>33</b>
<b>Función obtenersegundospasados</b>	<b>36</b>
<b>Función verificar_para_dar_BAJA</b>	<b>37</b>
<b>Función verificar_para_dar_ALTA</b>	<b>40</b>
<b>Función verificar_para_ASIG_TUTOR</b>	<b>42</b>
<b>Función verificar_tareas_coord</b>	<b>45</b>
<b>Función ZAVEZEZIONG</b>	<b>46</b>
<b>Función ZAVEZEZIONN</b>	<b>50</b>
<b>Función esta_completa</b>	<b>54</b>

# INTRODUCCIÓN

En este anexo vamos a tratar la documentación técnica de programación. En este manual vamos a tratar sobre el código generado para la aplicación.

El apartado tratará sobre el código fuente generada, su estructura de archivos, las convenciones que se han utilizado.

El propósito de este manual del programador es dar a conocer al lector todos los listados del programa realizado para ellos se tratará de forma amena y concisa un repaso de todas las unidades, ficheros Include, etc.; con el fin de que el usuario del conjunto puede modificar por necesidad algunos de los valores y parámetros de las funciones expuestas.

Para darle mantenimiento al sistema se necesita tener conocimientos sobre php 5, javascript y mysql 5. Pues esa es la plataforma sobre la que se ha creado el programa. Los programas necesarios, para poder modificar algún elemento del sistema SITE, no son muy complicados; en el caso de modificaciones al código fuente del programa no se necesita más que un editor de texto, por otro lado para realizar modificaciones a la base de datos, se necesita tener una interfaz de manejo y administración de MySQL.

# ESTRUCTURA DEL CÓDIGO

Vamos a explicar el contenido de la carpeta “SITE” que corresponde a la carpeta raíz de la aplicación y a otros recursos necesarios para ejecutar la aplicación.

El código fuente de la aplicación se encuentra dividida en cinco carpetas principales: “Administración”, “Ayuda”, “FXC”, “scripts” y “src/jpgraph\_cache”. Además tenemos dos carpetas: “archivos\_csv”, “imagenes” con la utilidad de almacenar archivos con extensión csv e imágenes de los Tutorados y usuarios.

## Carpeta “Administración”

Archivo	Función
AuxHor.php	Interfaz para el mantenimiento de los días disponibles para asistir a clases
AuxHor2.php	Interfaz para modificar los días disponibles para asistir a clases
AuxHorT.php	Interfaz para el mantenimiento de los días del horario de trabajo
AuxHorT2.php	Interfaz para modificar los días del horario de trabajo
Auxiliares.php	Página que contiene una lista de tablas auxiliares que se utilizan en la ficha de registro
Cambiar_Estado.php?var_q=1 <u>Actualizar Pensum</u>	Consulta las tablas “notas” y “nuke_retirados” en udb_porta_i_aaaa según el ciclo y el año seleccionado, para conocer el estado de las asignaturas cursadas por cada estudiante. El resultado de cada materia se almacena en la tabla pensum_alumno en la base de datos sitebd
Cambiar_Estado.php?var_q=2 <u>Actualizar Estado de Tutorados</u>	Se debe realizar al inicio de ciclo. Consulta la tabla “nuke_inscritos” en udb_porta_i_aaaa según el ciclo y el año seleccionado, para conocer si los tutorados están activos o inactivos. El estado se actualiza en la tabla tutorado en sitebd siempre y cuando no se han egresados
cargos.php	Interfaz para el mantenimiento de los cargos que desempeña en una empresa
cargos2.php	Interfaz para modificar los cargos que desempeña en una empresa
carrerasopc.php	Interfaz para el mantenimiento de las carreras que no se encuentran en la tabla “nuke_carrera”, es decir, no son impartidas en la universidad.
carrerasopc2.php	Interfaz para modificar las carreras que no se encuentran en la tabla “nuke_carrera”, es decir, no son impartidas en la universidad.
cargar_notas.php <u>Cargar Archivo de Egresados</u>	Interfaz para cargar archivo de egresados. Se debe realizar esta acción al final de cada ciclo universitario
change_estado.php	Actualiza el estado de todos los tutorados egresados en la tabla “tutorado” según el archivo cargado en la página <i>cargar_notas.php</i>
CostEst.php	Interfaz para el mantenimiento del encargado/a de pagar los

CostEst2.php	estudios académicos
crearciclo.php	Interfaz para modificar el encargado/a de pagar los estudios académicos
<u>Crear Tablas para Nuevo Ciclo Universitario</u>	Interfaz que crea las siguientes tablas en sitebd: - obs_ciclo_i_aaaa - reg_sesion_i_aaaa - resp_area_x_ciclo_i_aaaa
	Con el objetivo de almacenar las sesiones por ciclo y año y también para un fácil respaldo de información
DISCAPACIDADES.php	Interfaz para el mantenimiento de las discapacidades
DISCAPACIDADES2.php	Interfaz para modificar las discapacidades
egresados.csv	Archivo ejemplo para cambiar estado de tutorados en la herramienta <b>Cargar archivo de egresados</b>
empresas.php	Interfaz para el mantenimiento de las empresas
empresas2.php	Interfaz para modificar las empresas
enteroCarr.php	Interfaz para el mantenimiento de las formas de enterarse de la carrera seleccionada
enteroCarr2.php	Interfaz para modificar las formas de enterarse de la carrera seleccionada
entretenimiento.php	Interfaz para el mantenimiento de los entretenimientos
entretenimiento2.php	Interfaz para modificar los entretenimientos
espbachillerato.php	Interfaz para el mantenimiento de las especialidades de bachillerato
espbachillerato2.php	Interfaz para modificar las especialidades de bachillerato
ExpLab.php	Interfaz para el mantenimiento de las expectativas laborales
ExpLab2.php	Interfaz para modificar las expectativas laborales
ExpSE.php	Interfaz para el mantenimiento de las expectativas del servicio educativo
ExpSE2.php	Interfaz para modificar las expectativas del servicio educativo
GradAcad.php	Interfaz para el mantenimiento de los grados académicos
GradAcad2.php	Interfaz para modificar los grados académicos
GrupoFam.php	Interfaz para el mantenimiento del grupo familiar
GrupoFam2.php	Interfaz para modificar el grupo familiar
habilidades.php	Interfaz para el mantenimiento de las habilidades
habilidades2.php	Interfaz para modificar las habilidades
index.php	Página principal del <b>módulo de Mantenimiento</b>
instbachillerato.php	Interfaz para el mantenimiento de las instituciones educación media
instbachillerato2.php	Interfaz para modificar las instituciones educación media
instbecas.php	Interfaz para el mantenimiento de las instituciones que brindan becas
instbecas2.php	Interfaz para modificar las instituciones que brindan becas
MotInsCarr.php	Interfaz para el mantenimiento de los motivos para inscribir la carrera seleccionada
MotInsCarr2.php	Interfaz para modificar los motivos para inscribir la carrera seleccionada
MoTrab.php	Interfaz para el mantenimiento de los motivos para trabajar
MoTrab2.php	Interfaz para modificar los motivos para trabajar
municipios.php	Interfaz para el mantenimiento de los municipios
municipios2.php	Interfaz para modificar los municipios
noticias.php	Interfaz para el mantenimiento de las noticias
noticias2.php	Interfaz para modificar las noticias
países.php	Interfaz para el mantenimiento de los países

países2.php	Interfaz para modificar los países
personas1.php	Interfaz para el mantenimiento de los usuarios
personas2.php	Interfaz para modificar la información general de los usuarios
premios.php	Interfaz para el mantenimiento de los premios recibidos
premios2.php	Interfaz para modificar los premios recibidos
principales1.php	Página que contiene una lista de tablas principales
RazEstU.php	Interfaz para el mantenimiento de las razones para decidir estudiar en la Universidad Don Bosco
RazEstU2.php	Interfaz para modificar las razones para decidir estudiar en la Universidad Don Bosco
servicio.php	Interfaz para el mantenimiento de las instituciones de servicio
servicio2.php	Interfaz para modificar las instituciones de servicio
universidad.php	Interfaz para el mantenimiento de otras universidades
universidad2.php	Interfaz para modificar otras universidades

## Carpeta “Ayuda”

Archivo	Función
ayuda_imagen.php	Donde se cargan los archivos de ayuda

## Carpeta “FXC”

Archivo	Función
BDacceso.php	Se encuentran diferentes funciones que se llaman desde el archivo BDacceso.php
minitablas	Para actualizar las tablas auxiliares donde se relaciona el Carnet_Tut con otra tabla principal. Por ejemplo: Tabla Principal: conoc_carrera Tabla auxiliar: conoc_carrera_alum Esta función sirve para ingresar elementos en la Tabla Principal si no estaba, y para actualizar la tabla auxiliar, ya sea eliminando, insertando registros o dejando los registros tal como se encuentran.
preio	Función creada para ingresar las carreras opcionales de lo tutorados. Si alguna no está en la base de datos la agrega.
correos_f.php	Se encuentran diferentes funciones conteniendo formatos de envío de notificaciones a través de correos.
enviarcorreo_dar_baja	En esta función se crea el correo que se le enviara a un coordinador de X facultad cuando el coordinador de la facultad A solicita que se le de baja a un estudiante de la facultad X.
enviarcorreo_dio_baja2	En esta función se crea el correo que se le envía a un tutor cuando se le quita uno o varios tutorados cuando se le da baja a los tutorados en su facultad.
enviarcorreo_dio_alta	En esta función se crea el correo que se le envía a un tutor cuando se le asigna uno o varios tutorados cuando se le da alta a los tutorados en su facultad o cuando se le asignan tutorados.
enviarcorreo_quito_tutorado	En esta función se crea el correo que se envía cuando se le quita(n) tutorado(s) al tutor, ya sea por cambio de carrera o porque se dio baja
enviarcorreo_d_tutorado	En esta función se crea el correo que se le envía a los tutorados cuando se le asigna un tutor
funciones.php	En este archivo están la mayoría de las funciones empleadas en el

## Sistema

validate_email	Valida el correo electrónico no institucional
validate_email2	Valida el correo electrónico institucional
valida_usuario	Función que verifica que en el usuario ingresado solo hayan caracteres alfanuméricos y los siguientes símbolos: "_", "%", ".", "y" y "-".
validate_tele	Función que verifica los caracteres válidos para un numero telefónico fijo
validate_cel	Función que verifica los caracteres válidos para un numero telefónico celular.
validate_names	Función que verifica los caracteres ingresados en los campos donde se ingresa un nombre
validate_last	Función que verifica los caracteres ingresados en los campos donde se ingresa un apellido
validate_direc	Función que verifica que al menos se haya digitado dos caracteres en un campo de dirección.
validate_duit	Función que verifica si el formato del DUI ingresado es valido.
validate_minimo	Función que verifica que al menos se haya digitado dos caracteres en un campo.
numerohoras	Función que recibe dos horas y dos parámetros de AM/PM; con esto realiza los cálculos para obtener el número de horas pasadas entre ambos límites de tiempo.
llamar_a	De acuerdo a un parámetro, indica cual es la siguiente página.
llamar_a2	
opciones	Función donde se encuentran diferentes conjuntos de opciones que aparecen en algunos campos de selección y que se tiene que no cambiaran mucho a lo largo del tiempo. Un ejemplo podría ser el tipo de sangre: al enviar el parámetro gruposanguineos, la función devuelve un arreglo con la información "A RH+"; "B RH+"; "O RH+"; "AB RH+"; "A RH-"; "B RH-"; "O RH-"; "AB RH-"; "PENDIENTE";
carnets	Función que genera el carné provisional de acuerdo a las iniciales de su nombre, apellidos y año actual.
pasara24horas	Función que obtiene como parámetros una hora en formato de 12 horas y un parámetro indicando si es a.m. o pm., y devuelve el valor de la hora en formato de 24 horas.
anio_s	Función que obtiene como parámetro la fecha en formato "aaaa-mm-dd" y cada una devuelve el valor respectivo: anio_s, devuelve el año;
mes_s	mes_s devuelve el mes y dia_s devuelve el día.
dia_s	
bd_a_hora	Función que devuelve la hora al obtener un valor de hora con formato hh:mm, devuelve solo la porción hh.
Seguridad	Función que valida que la sesión no haya caducado y que la pagina que se esta accesando sea permitida para el nivel de usuario que se encuentra navegando.
verif_carrera	Función que se utiliza para verificar que una carrera que está a punto de ingresada en la base de datos, no exista y que tenga valores validos.
cambiar_Usu	Función que actualiza todas las tablas necesarias cuando se cambia el usuario de una persona.
cambiar_carnet	Función que actualiza todas las tablas relacionadas con la ficha de registro al momento de cambiar el carnet de provisional a oficial de los tutorados.
enviarcorreo	Función donde se crea el correo que se les envía a los tutores cuando son ingresados al sistema.
nuevoCiclo_Anio	Función donde se crean 3 tablas al crear ciclo nuevo, estas tablas son: `reg_sesion_ii_aaaa`, `resp_area_x_ciclo_ii_aaaa`, `obs_ciclo_ii_aaaa`. ii indica el ciclo a crear y aaaa indica el año en el

	cual se creara el ciclo.
ta,tb,tc,td,te,tf,tg,th,ti,tj	Estas funciones se encaran de verificar que cada parte de la ficha de registro este completa. Por ejemplo la función ta, verifica que la parte A este completa.
tiene_ficha_al_cien	Función que verifica que una ficha de registro de un estudiante este completa (todas las partes llenas)
storefile	Función que se utiliza para verificar la extensión (jpg) y tipo de archivo a subir al servidor y si esta todo correcto copia el archivo al servidor. En esta función se suben las imágenes de los usuarios.
dias_mes	Función que calcula cuantos días tiene un mes específico.
nombremes	Función que devuelve el nombre de un mes cuando se le envía como parámetro numérico
convertCICLO2let	Función que recibe el numero de ciclo en numero y lo devuelve en formato i,ii,iii
getcilclo	Función que recibe un numero de mes y se devuelve a que ciclo pertenece en formato i,ii,iii
getcilclo_num	Función que recibe un numero de mes y se devuelve a que ciclo pertenece en formato numérico, 1,2,3
meses_de_ciclo	Función que recibe el numero de ciclo y devuelve los números de los meses que pertenecen a ese ciclo
convertciclo2num	Función que recibe el numero de ciclo en formato I, II ó III y lo devuelve en formato numérico ya sea 1, 2 ó 3.
enviarcorreo2	Función que crea el correo que se le enviara al tutor cuando se actualiza información de su perfil por el coordinador de su facultad.
nombre_nivel	Esta función devuelve la descripción del nivel numérico que se le envía
href_function	Esta función sirve para indicar que pagina de inicio le corresponde a cada quien de acuerdo al nivel de usuario.
<b>obtenersegundospasados</b>	Esta función es utilizada en la ficha de registro cuando es llenada por primera vez por el tutorado, su función principal es calcular el tiempo que ha transcurrido desde que el estudiante comenzó a completar la ficha de registro.
storefile_csv	Función que se utiliza para verificar la extensión (csv) y tipo de archivo a subir al servidor y si esta todo correcto copia el archivo al servidor. En esta función se suben los archivos que se utilizan para importar datos.
Estado_Tut2letra	Función que devuelve el Estado de un Tutorado en letras(Activo, inactivo y egresado) al recibir el estado en convención numérica (1,2,3)
bib_datediff	Devuelve el tiempo transcurrido entre dos fechas específicas. Se utiliza más que todo para calcular la edad.
cumplea	Verifica los cumpleaños del día actual. Si el tutor ha configurado que se le recuerde mas tarde, se muestra una ventana para que se le pueda enviar un mensaje de felicitación, sino no se muestra esta ventana.
createMonth	Funciones que se utilizan para la visualización del calendario
appendEvent	Y también se utiliza al programar la próxima cita de una sesión.
addToday	
displayMonth	
displayEvent	
editar_event	
eliminar_event	
login	
validar_evento	
addEvent	



saveEvents	
makeDate	
fac_carnet	
verificar_para_dar_BAJA	Esta función se emplea para verificar si no hay Tutorados para dar de baja actualmente.
verificar_para_dar_ALTA	Esta función se emplea para verificar si no hay Tutorados para dar de alta actualmente.
verificar_para_ASIG_TUTOR	Esta función se emplea para verificar si no hay tutorados que se han cambiado de carrera (de una facultad a la del coordinador usando el sistema). Esta verificación se realiza con el fin de Recordarle al Coordinador que debe asignarle un Tutor
verificar_tareas_coord	Esta función verifica las tareas pendientes de dar baja, dar alta ó asignar tutor a los tutorados que han ingresado a la facultad a través del cambio de carrera
ultimo_plan	Esta función ve en la tabla nuke_asignatura para ver cual es el ultimo pensum actualizado para X materia.
SESSIONsave.php	Las funciones que se encuentran en este archivos se utilizan para almacenar sesiones
ZAVEZEZIONG	Guarda información general (observaciones y recomendaciones generales) de la sesión que realiza el Tutor con el Tutorado
estadodesesion	Actualmente no se usa en ninguna parte del sistema, lo que hace es ver si están todas las partes de la sesión llenas.
ZAVEZEZIONN	Guarda la información cada área de Sesión con sus respectivas observaciones.
esta_completa	Verifica que se haya ingresado al menos una observación ya sea general ó en alguna de las Áreas.

## Carpeta “scripts”

Archivo	Función
Format.js	Sirve para colocar un guión después de 4 números en el teléfono y también valida que solo se ingresen números en el teléfono y la extensión
script.js	Archivo que almacena diferentes funciones
cambiar_p	Con esta función se cambia de nivel
color_c	Esta función sirve para cambiar de color a los botones
open_window	Función para abrir una nueva ventana con ninguna de las barras
Help_go	Función para abrir la página <b>ayuda_p.php</b>
onlyNum	Función que obliga ingresar solo números

## Carpeta “src/jpgraph\_cache”

Archivo	Función
graficar_program.php	Programa para crear la gráfica de pastel
graficar_program_barra.php	Programa para crear la gráfica de barras

## Carpeta “archivos\_csv”

Carpeta que almacena los archivos csv que son cargados al servidor desde el sistema.

## Carpeta “imagenes”

Carpeta que almacena las imágenes del sistema, esta dividida en otras carpetas que tiene como nombre la extensión de las imágenes que contienen. Por ejemplo: imagenes/gif/

## Carpeta raíz

Archivo	Función
autentico.php	Programa para autenticar al usuario
ayuda_p.php	Archivo que se utiliza para visualizar las ayudas del sistema
clase_Encuesta.php	Programa que contiene Conjunto de clases empleadas para actualizar y guardar la ficha de registro en la tabla “tutorado”
Conexion.php	Archivo de configuración de conexión a la base de datos sitebd
conexionporta.php	Archivo de configuración de conexión a la base de datos udbporta
	Sirve para destruir las matrices creadas en la sesión al generar reportes. Esto se lleva a cabo al salir de la página donde esta el reporte. Además se utiliza para regresar los valores que se utilizaron para crear el reporte
destruyes.php	Sirve para destruir las variables creadas en la sesión al llenar la ficha de registro. Esto se lleva a cabo al salir de la página donde esta la ficha
destruyes_ficha.php	Página de inicio de la aplicación. Aquí se ingresa el usuario y la contraseña.
inicio.php	Con esta página se cambia de nivel de usuario y se eliminan las variables de sesión
Niveles_usuario.php	Archivo para visualizar las noticias
noticias_detalle.php	Archivo que contiene diferentes estilos que hacen el formato y las posiciones de las imágenes de las paginas
SITE.css	Interfaz para el mantenimiento del perfil de los usuarios
PerfilUsuario.php	

## Comunes

	A está pagina tienen acceso los niveles de Coordinador y Tutor. Se utiliza para visualizar los pensum de acuerdo al nivel, es decir, para el nivel de Coordinador se visualizan los pensum de todas las facultades y para el nivel de Tutor se visualizan los pensum solamente de la facultad del Tutor
pensum.php	Interfaz donde se seleccionan varias opciones según las preferencias o necesidades del usuario para crear un reporte. A está pagina tienen acceso el nivel de Coordinador y Tutor
reporte_dinamico.php	Programa donde se presenta el resultado de los parámetros seleccionados en el <b>reporte_dinamico.php</b>
reporte_dinamico_vista.php	Programas que conforman la ficha de registro. Se utilizan para actualizar la ficha de registro. A estas paginas tienen acceso el nivel de Tutor y Alumno
parteaT1.php	
partebT1.php	
partecdT1.php	
parteefT1.php	
partegT1.php	

partehiT1.php  
partejT1.php  
toexcel\_din.php

Archivo donde se genera el archivo Excel a partir del reporte presentado en **reporte\_dinamico\_vista.php**

## Alumno

endE.php  
indexE.php  
partea1.php  
parteb1.php  
partecd1.php  
parteef1.php  
parteg1.php  
partehi1.php

Página que sirve para cerrar la aplicación después de haber terminado de llenar la ficha de registro  
Página de inicio donde se muestra un introducción para los alumnos que están a punto de ingresar a la ficha de registro por primera vez  
Programas que conforman la ficha de registro. Se utilizan para llenar la ficha de registro por primera vez

## Evaluador-Consultor

consolasps.php  
Evaluadorindex.php  
  
reportes.php  
reportesejec.php  
toexcel.php  
tosps.php  
tosps\_process.php

Interfaz para seleccionar la facultad, el ciclo y el año de ingreso  
Página principal para el nivel Evaluador-Consultor  
Interfaz para seleccionar la facultad, el ciclo y el año de ingreso y el tipo de reporte a generar. Los tipos de reporte que se pueden seleccionar son todas aquellas preguntas que contiene la ficha de registro  
Programa donde se presenta el resultado de los parámetros seleccionados en el **reportes.php**  
Archivo donde se genera el archivo Excel a partir del reporte presentado en **reportesejec.php**  
Archivo donde se genera el archivo Excel a partir del reporte presentado en **tosps\_process.php**  
Programa donde se presenta el resultado de los parámetros seleccionados en el **consolasps.php**

## Tutor

AREASESIONES.php  
AREASESIONESG.PHP  
correosT.php  
  
ctrlasistencia.php  
Cumpleaneros.php  
Cumpleaneros\_mensaje.php  
DetaPens.php

Interfaz que presenta las áreas de sesión para que puedan ser completadas por el usuario  
Interfaz que permite ingresar una próxima cita y se presentan las notas del ciclo actual del Tutorado  
Página que se utiliza para enviar correo a varios Tutorados de una sola vez  
Página que muestra información académica y de la sesiones del Tutorado por ciclo académico o por mes calendario. Se utiliza para visualizar el numero de sesiones que se lleva en el ciclo y poder observar el seguimiento de la Tutoría para cada sesión  
Página que muestra los cumpleaños del día  
Página que se utiliza para enviar un mensaje de felicitación a los cumpleaños que aparecen en la página **Cumpleaneros.php**  
Página que muestra el historial completo de las materias cursadas

DetaPensA.php	Página que muestra información del estado de una materia cursada
DetaPensElectivas.php	Página que muestra las electivas de una materia
parteaTR1.php	Programas que conforman la ficha de registro. Se utilizan únicamente para visualizar la ficha de registro
partebTR1.php	
partecdTR1.php	
parteefTR1.php	
partegTR1.php	
partehiTR1.php	
partejTR1.php	
pensumA.php	Página para visualizar el avance del pensum de un Tutorado específico
Sesiones.php	Página para seleccionar un Tutorado
Sesiones.php?g=3	Página para cambiar el carnet provisional a un Tutorado
	Página donde se visualiza el avance en la ficha de registro por Tutorado. Además se puede enviar un correo electrónico personalizado
Tutor_correospartes.php	Sirve para consultar las notas del ciclo actual
TutorConsNota.php	Página que contiene diferentes consultas en un menú
TutorConsulta.php	Página principal para el nivel de acceso Tutor
TutorIndex.php	Página para visualizar las próximas citas que ha creado el Tutor. Además en esta página se pueden eliminar las citas
TutorMostarEventos.php	Página donde se seleccionan los reportes predefinidos
TutorReportes.php	Programa donde se presenta el resultado de los parámetros seleccionados en el <b>TutorReportes.php</b>
TutorReportes_vista.php	

## Coordinador

CoorActualizarEst.php	Programa que se utiliza para cambiarle el estado a los Tutorados
CoordinadorAdmin.php	Página para el mantenimiento de los Tutores
CoordinadorAsignar.php	Página para asignar Tutor a uno o más Tutorados
	Página para cambiar la carrera a uno o más Tutorados. Si es de la misma Facultad el Tutorado también le cambia el Tutor
CoordinadorCHcarrera.php	Página donde se encuentra el menú de las consultas que el usuario Coordinador puede realizar
CoordinadorConsulta.php	Buscar el Tutor asignado al Tutorado ingresado. También se muestra información general del Tutorado
CoordinadorConsulta.php?cons=1	
<u><i>Consulta de Tutor asignado por estudiante</i></u>	
	Página que muestra el porcentaje de Tutorados que tiene la ficha completa por cada Tutor en la facultad del usuario Coordinador
CoordinadorConsulta1.php	Página que presenta el número de sesiones por cada Tutor en la facultad del usuario Coordinador en un período específico
CoordinadorConsulta2.php	Interfaz para el mantenimiento de Tutores
CoordinadorCrear.php	Interfaz para modificar la información general de los Tutores
CoordinadorCrear2.php	Página donde se presentan los Tutorados que tiene pendientes para dar de alta
CoordinadorDarAlta.php	Página donde se presentan los Tutorados que tiene pendientes para dar de alta
CoordinadorDarBaja.php	Página para eliminar la asignación de Tutores
CoordinadorEliminarA.php	Página principal para el nivel de acceso Coordinador
Coordinadorindex.php	Programa para solicitar dar de baja a un Tutorado de cualquier facultad
CoordinadorSol_Baja.php	

CoordinadorTareas.php  
CoordinadorTutPartes.php  
CoordinadorTutPartes\_P.php  
coordinadorVerTut.php  
CoorFichasTutorados.php  
CoorListaTutorados.php

Interfaz que muestra los Tutorados que se necesita que se le den de baja o alta o que se les asigne un Tutor  
Página que muestra el estado de las ficha de registro como un detalle  
Página que muestra el estado de las ficha de registro en resumen  
Página que sirve para seleccionar uno o varios Tutores  
Página que muestra el resultado **CoordinadorTutPartes.php**  
Página que muestra los Tutorados por Tutor

## Función minitablas

Para almacenar la ficha de registro se utilizan clases. Las clases utilizadas son las siguientes parteA, parteB, parteCD, parteEF, parteG, parteHI, parteJ. Cada una de ellas almacena todas las respuestas de cada parte y para almacenar a la base de datos tienen una función especial para ello **Guardar\_a\_BD**.

```
function minitablas ($IdMaestraV, $IdMaestraN, $tablamaestraN, $desc_maestraN,
$desc_maestraV, $tablalejanaN, $carnet,$link, &$this->Id_Ins_Serv, &$this-
>Desc_Servicio){
    ....definición de la función...
}
```

En la función **Guardar\_a\_BD**, se utiliza la función **minitablas**.

**Tabla inst de servicio**, maestro de las instituciones de servicio.

**Estructura:**

	_____

**Datos**

Id_Ins_Serv	Desc_Servicio	Direc_servicio
1	Club de Leones	
2	Club Rotario	
3	Clubes Deportivos	
4	Asociaciones de jóvenes cristianos	
5	Instituciones de emergencia (Cruz Roja, Cruz Verde, etc)	
6	Club de jóvenes exploradores	
7	Voces de El Salvador	

**Tabla inst de servicio alum**.

En esta tabla se relacionan las instituciones de servicio seleccionadas por el tutorado. Por ejemplo, si el tutorado con carnet CH060031 ha seleccionado el *Club de Leones* y *Voces de El Salvador*, este sería el contenido de esta tabla:

**Estructura:**

	_____
	_____

**Datos**

Carnet_Tut	Id_Ins_Serv
CH060031	1
CH060031	7

Lo que se realiza con la función **minitablas** es que si el Estudiante ya no pertenece a una de las instituciones que había seleccionado anteriormente, se elimina ese registro; también si pertenece a más instituciones, se agregan a la tabla **inst\_de\_servicio\_alum**; si

**inst\_de\_servicio**

**PK Id\_Ins\_Serv**

**Desc\_Servicio**

es necesario se ingresan instituciones que no estaban en la tabla maestra. Y si las instituciones para este Estudiante siguen siendo las mismas no se realiza ningún cambio.

Para usar esta función los **parámetros** que se envían son los siguientes:

IdMaestraV: vector que contiene las llaves primarias de una tabla maestra que han sido seleccionadas.

IdMaestraN: Nombre del campo llave primaria de la tabla maestra

tablamaestraN: Nombre de la tabla maestra

desc\_maestraN: Nombre del campo que es la descripción en la tabla maestra

desc\_maestraV: Descripción de un nuevo elemento en la tabla primaria.

tablalejanaN: Nombre de la tabla auxiliar, donde se relaciona el Carnet del Estudiante con la tabla maestra.

carnet: El carnet que se afectara.

link: vinculo de conexión a la base de datos.

&\$this->Id Ins Serv: variable que contiene el vector de las llaves primarias de una tabla maestra que han sido seleccionadas

&\$this->Desc Servicio: variable que contiene la descripción de un nuevo elemento en la tabla primaria.

El código de la función:

```
function minitablas($EXT_IdMaestraV, $EXT_IdMaestraN, $EXT_tablamaestraN,
$EXT_desc_maestraN, $EXT_desc_maestraV, $EXT_tablalejanaN, $EXT_carnet,
$EXT_link, $EXT_matrizo, $EXT_descripciono){
    $EXT_antes=array();//del tutorado
    $EXT_consMS="SELECT ".$EXT_IdMaestraN." FROM ".$EXT_tablalejanaN."
WHERE `Carnet_Tut`='".$EXT_carnet.'";
    $EXT_ResConsMS=mysqli_query($EXT_link,$EXT_consMS);
    if(!(mysqli_num_rows($EXT_ResConsMS)==0)){
        while($EXT_rowCons=mysqli_fetch_array($EXT_ResConsMS)){
            $EXT_antes[]=$EXT_rowCons[$EXT_IdMaestraN];
        }
    }
    if(sizeof($EXT_IdMaestraV)==0){
        $EXT_diferencia=$EXT_antes;
    }else{
```





```

$EXT_ResConsMS=mysqli_query($EXT_link,$EXT_consMS);
if(!(mysqli_num_rows($EXT_ResConsMS)==0)){
    while($EXT_rowCons=mysqli_fetch_array($EXT_ResConsMS)){
        $EXT_antes[]=$EXT_rowCons[$EXT_IdMaestraN];
    }
}

```

En la variable **\$EXT\_antes** se almacena el vector de las llaves primarias que anteriormente estaban asignadas al tutorado en cuestión.

En la variable **\$EXT\_diferencia** se colocan todas las llaves primarias que ya no tienen relación con el tutorado pero que antes sí, por lo tanto se eliminarán de la tabla auxiliar.

```

if(sizeof($EXT_IdMaestraV)==0){
    $EXT_diferencia=$EXT_antes;
}else{
    $EXT_diferencia=array_diff($EXT_antes,$EXT_IdMaestraV);
}

```

Si el número de elementos seleccionados esta vez es igual a cero, en el vector **\$EXT\_diferencia** se colocan todas las que tenía asignadas el tutorado, pues todas se deben eliminar, pues no se ha seleccionado nada. Caso contrario (que el número de elementos seleccionados sea mayor a cero), se saca la diferencia de las que tenía asignadas antes y las que selecciono ahora, es decir se obtendrán las que ya no están seleccionadas ahora que antes sí estaban seleccionadas, y dicho resultado se almacenara en **\$EXT\_diferencia**.

```

$EXT_DD=sizeof($EXT_diferencia);
if ($EXT_DD>0){
    $EXT_diferencia=implode(',',$EXT_diferencia);
    $EXT_Del="DELETE FROM ".$EXT_tablalejanaN." WHERE
    ".$EXT_IdMaestraN." IN ( ".$EXT_diferencia." )";
    mysqli_query($EXT_link,$EXT_Del);
}

```

Si hay elementos que YA NO están relacionados con el tutorado, entonces se procede a eliminarlos de la tabla auxiliar [`$EXT_tablalejanaN`].

```

if(!empty($EXT_desc_maestraV)){
    $EXT_Sql1="INSERT INTO ".$EXT_tablamaestraN."
        (".$EXT_IdMaestraN.",`".$EXT_desc_maestraN."`)
        values(0,`".$EXT_desc_maestraV."`);
    mysqli_query($EXT_link,$EXT_Sql1);
    $EXT_lastSql1=mysqli_insert_id($EXT_link);
    $EXT_Sql_1="INSERT INTO ".$EXT_tablalejanaN."
        (Carnet_Tut`,`".$EXT_IdMaestraN."`)
        values(`".$EXT_carnet."`,`".$EXT_lastSql1."`);
    mysqli_query($EXT_link,$EXT_Sql_1);
    array_push($EXT_matrizo,$EXT_lastSql1);// INSERTAR AL FINAL
    $EXT_descripcion="";
}

```

Si hay un nuevo elemento que no existía antes, por ejemplo una nueva institución de servicio, se ingresa en la tabla maestra y se relaciona con el tutorado ingresándolo también en la tabla auxiliar. Una vez hecho esto, la llave primaria del nuevo elemento se adiciona al final del vector de los elementos seleccionados [`$EXT_matrizo ≡ &$this->Id_Ins_Serv`] y se limpia la variable donde estaba la descripción del nuevo elemento. [`$EXT_descripcion ≡ &$this->Desc_Servicio`]

```

if(sizeof($EXT_IdMaestraV)!=0){
    foreach ($EXT_IdMaestraV as $EXT_Id1_Ins_Serv){
        $EXT_Sql_1="INSERT INTO ".$EXT_tablalejanaN."
            (`Carnet_Tut`,`".$EXT_IdMaestraN."`)
            values(`".$EXT_carnet."`,`".$EXT_Id1_Ins_Serv."`);

        $EXT_query=mysqli_query($EXT_link,$EXT_Sql_1);
    }
}

```

Si hay elementos seleccionados [vector de llaves primarias], entonces se relacionan con el tutorado, ingresándolos en la tabla auxiliar.

Esta función no retorna resultado.

Para mostrar el como opera esta función, se puede tomar como ejemplo la clase parteA, la cual entre sus miembros posee:

Id\_Ins\_Serv, que es el vector donde se almacenan las llaves primarias de las instituciones de servicio seleccionadas,

Desc\_Servicio que es donde se almacena la descripción de una nueva institución de servicio.

Carnet\_Tut variable que contiene el carnet del tutorado que se encuentra llenando la ficha de registro

. La forma de llamar la función minitablas es de la siguiente manera:

```
$IdMaestraV=$this->Id_Ins_Serv;  
$IdMaestraN="Id_Ins_Serv";  
$tablamaestraN="inst_de_servicio";  
$desc_maestraN="Desc_Servicio";  
$desc_maestraV=$this->Desc_Servicio;  
$tablalejanaN="inst_de_servicio_alum";  
$carnet=$this->Carnet_Tut;  
minitablas($IdMaestraV,$IdMaestraN,$tablamaestraN,$desc_maestraN,$desc_maestraV,  
$tablalejanaN,$carnet,$link,&$this->Id_Ins_Serv,&$this->Desc_Servicio);
```

Retomando el ejemplo del carnet CH060031 se tendría:

```
$IdMaestraV=array(1,7);  
$IdMaestraN="Id_Ins_Serv";  
$tablamaestraN="inst_de_servicio";  
$desc_maestraN="Desc_Servicio";  
$desc_maestraV="Libras de Amor";  
$tablalejanaN="inst_de_servicio_alum";  
$carnet="CH060031";
```

Después de la llamada a la función el resultado para estas variables sería

```
$this->Id_Ins_Serv= array(1,7,8);  
$this->Desc_Servicio ="";
```

## Función preio

Esta función se utiliza para almacenar las carreras seleccionadas en algún momento por el tutorado.

```
function preio ($idcarreraopcX, $descarreraopcX, $link, $num, $carnet){  
    ....definición de la función...  
}
```

### **Parámetros:**

idcarreraopcX: Identificador (llave primaria) de la carrera seleccionada

descarreraopcX: Descripción ó nombre de la carrera que se desea seleccionar, pero que no se encuentra en las opciones disponibles de carreras.

link: vinculo de conexión a la base de datos.

num: para indicar que es el registro a insertar, puede ser 1=primera carrera opcional; 2=segunda carrera opcional; 3=tercera carrera opcional; 4=carrera que estudiaba antes.

carnet: Carnet del tutorado

Esta función no retorna resultado.

### Código de la función

```
function preio($EXT_idcarreraopcX, $EXT_descarreraopcX, $EXT_link,  
$EXT_num, $EXT_carnet){  
    if($EXT_idcarreraopcX==-1){//selecciono otra carrera  
        $EXT_Sql1="INSERT INTO carreras_m  
        (`Id_Carrera`,`Desc_Carrera` ) values  
        (0, '$EXT_descarreraopcX.' )";  
        mysqli_query($EXT_link, $EXT_Sql1);  
        $EXT_idcarreraopcX=mysqli_insert_id($EXT_link);  
        $EXT_descarreraopcX="";  
        $EXT_letra="E";  
    }else{  
        $EXT_end =strlen($EXT_idcarreraopcX);  
        $EXT_start= strrpos($EXT_idcarreraopcX,"E");  
        if($EXT_start===false)  
            $EXT_letra="U";  
        else{  
            $EXT_letra="E";  
            $EXT_start++;  
            $EXT_idcarreraopcX2=$EXT_idcarreraopcX;
```

```

        $EXT_idcarreraopcX=substr($EXT_idcarreraopcX2,
        $EXT_start, $EXT_end - $EXT_start);
    }
}
$EXT_sel2="select * from carreraopcional where
Carnet_Tut='".$EXT_carnet.'" AND Num_Opcion='".$EXT_num;
$EXT_sel2resul=mysqli_query($EXT_link, $EXT_sel2);
if(mysqli_num_rows($EXT_sel2resul)!=0)
    $EXT_Sql1="UPDATE `carreraopcional` set
    `Id_Carrera_m_o_nuk`='".$EXT_idcarreraopcX."',
    `porta`='".$EXT_letra.'" WHERE
    `Carnet_Tut`='".$EXT_carnet.'" AND
    `Num_Opcion`='".$EXT_num;
else
    $EXT_Sql1="INSERT INTO carreraopcional
    values('".$EXT_carnet."', '".$EXT_idcarreraopcX."', ".$EXT_
    T_num.", '".$EXT_letra.'")";
mysqli_query($EXT_link, $EXT_Sql1);
if(isset($EXT_idcarreraopcX2))
    $EXT_idcarreraopcX=$EXT_idcarreraopcX2;
} //de la función preio

```

Para la base de datos sitebd, se tiene la tabla **carreraopcional**, donde se relacionan las carreras con los Estudiantes:

Estructura	Datos																								
<table border="1"> <tr> <td style="background-color: #cccccc;"></td> <td></td> </tr> <tr> <td></td> <td>_____</td> </tr> <tr> <td></td> <td>_____</td> </tr> <tr> <td></td> <td></td> </tr> </table>				_____		_____			<table border="1"> <thead> <tr> <th>Carnet_Tut</th> <th>Id_Carrera_m_o_nuk</th> <th>Num_Opcion</th> <th>porta</th> </tr> </thead> <tbody> <tr> <td>CH060031</td> <td>407</td> <td>1</td> <td>U</td> </tr> <tr> <td>CH060031</td> <td>108</td> <td>2</td> <td>U</td> </tr> <tr> <td>CH060031</td> <td>2</td> <td>3</td> <td>E</td> </tr> </tbody> </table>	Carnet_Tut	Id_Carrera_m_o_nuk	Num_Opcion	porta	CH060031	407	1	U	CH060031	108	2	U	CH060031	2	3	E
	_____																								
	_____																								
Carnet_Tut	Id_Carrera_m_o_nuk	Num_Opcion	porta																						
CH060031	407	1	U																						
CH060031	108	2	U																						
CH060031	2	3	E																						

En esta tabla Num\_Opcion indica el número de opción de la carrera (primera, segunda ó tercera). Porta puede tomar dos valores 'U' (indica que es una carrera de UDBPorta. nuke\_carrera) ó 'E'(indica que es una carrera que no es de UDBPorta. nuke\_carrera, sino que pertenece a la tabla sitebd.carreras\_m).

Explicación del funcionamiento de la función en detalle:

Primero verifica si ha seleccionado otra carrera, cuando idcarreraopcX es igual a "-1" es porque se ha seleccionado "OTRA CARRERA" que no esta en ninguna base de datos. Si es así, se ingresa en la tabla carreras\_m en la base de datos 'sitebd'. Si no selecciono 'otra carrera', se verifica si es de UdbPorta o de SiteBd; si tiene la "E" al inicio del valor es porque no es de UdbPorta.

```

if($EXT_idcarreraopcX==-1){//selecciono otra carrera
    $EXT_Sql1="INSERT INTO carreras_m
    (`Id_Carrera`,`Desc_Carrera` ) values
    (0, ' ".$EXT_desccarreraopcX." ' )";
    mysqli_query($EXT_link, $EXT_Sql1);
    $EXT_idcarreraopcX=mysqli_insert_id($EXT_link);
    $EXT_desccarreraopcX=" ";
    $EXT_letra="E";
}
else{
    $EXT_end =strlen($EXT_idcarreraopcX);
    $EXT_start= strrpos($EXT_idcarreraopcX,"E");
    if($EXT_start===false)
        $EXT_letra="U";
    else{
        $EXT_letra="E";
        $EXT_start++;
        $EXT_idcarreraopcX2=$EXT_idcarreraopcX;
        $EXT_idcarreraopcX=substr($EXT_idcarreraopcX2,
        $EXT_start, $EXT_end - $EXT_start);
    }
}
}

```

Luego se consulta la base de datos para saber si este numero de opción ya había sido ingresado para el Tutorado, de esta forma se realiza ya sea una inserción en la base de datos ó una actualización.

```

$EXT_sel2="select * from carreraopcional where
Carnet_Tut=' ".$EXT_carnet."' AND Num_Opcion=' ".$EXT_num;
$EXT_sel2resul=mysqli_query($EXT_link, $EXT_sel2);
if(mysqli_num_rows($EXT_sel2resul)!=0)
    $EXT_Sql1="UPDATE `carreraopcional` set
`Id_Carrera_m_o_nuk`=' ".$EXT_idcarreraopcX." ,
`porta`=' ".$EXT_letra."' WHERE

```

```

        `Carnet_Tut`='".$EXT_carnet."' AND
        `Num_Opcion`=".$EXT_num;
else
    $EXT_Sql1="INSERT INTO carreraopcional
    values('".$EXT_carnet."', '".$EXT_idcarreraopcX."', ".$EXT_
    T_num.", '".$EXT_letra."')";
mysqli_query($EXT_link, $EXT_Sql1);
if(isset($EXT_idcarreraopcX2))
    $EXT_idcarreraopcX=$EXT_idcarreraopcX2;

```

Esta función se utiliza en la parteB y en la parteJ.

En la parteB se utiliza para las opciones de carrera que hubieran seleccionado los tutorados. Como ejemplo, se muestra la llamada de la función cuando se debe insertar la primera opción del tutorado.

```

preio(&$this->Id_Carrera_Opcional1, &$this->Desc_Carrera_Opcional1, $link,1, $this-
>Carnet_Tut);

```

## Función opciones

La función opciones tiene una utilidad vital en el Sistema de Tutorías SITE. Debido a que en esta función se encuentran definidas las posibles opciones que aparecen en los combo-box de la aplicación. Los valores de combo-box definidos en esta función son: departamentos, grupo sanguíneo, nivel académico de ingles, parentesco, tipo de residencia, sexo, estado civil, niveles de estudios, opciones de cuando empezó a trabajar, las diferentes condiciones de inscripción, tipos de centro educativo, zona de ubicación de centros educativos, opciones de reportes, opciones de ciclos, estados de tutorado, estados de una materia en el pensum.

Para obtener las opciones disponibles, se llama a la función de la siguiente manera

```
$opcion_deseada="departamentos";
```

```
$resulopc=opciones($opcion_deseada).
```

De esta manera en resulopc se obtiene un vector con las diferentes opciones para el parámetro enviado. En la siguiente tabla se muestran los posibles parámetros a enviar:

<b>Parámetro</b>	<b>Valores que retornara la función</b>
departamento	departamentos
gruposanguineos	Grupos sanguíneos
nivelingles	nivel académico de ingles
parentesco	parentesco
residencia	tipo de residencia
sexo	sexo
estciv	estado civil
mayniv	niveles de estudios
comtrab	opciones de cuando empezó a trabajar
CondicionesInsc	las diferentes condiciones de inscripción
tipocentro	tipos de centro educativo
zonaubicacion	zona de ubicación de centros educativos
reportes	opciones de reportes
ciclo_r	opciones de ciclos, incluyendo la opción "Todos"
Estado_Tut	estados de tutorado
ciclo_detapensA	Opciones de ciclo académico
Estado_detapensA	estados de una materia en el pensum
Tutor_Reportes	Reportes disponibles para tutores

Esta función retorna como resultado un vector con las posibles opciones.



## Función carnets

```
function carnets ($nombre, $apellido, $link){  
    ....definición de la función...  
}
```

Esta función es la que se encarga de crear un carnet provisional cuando el estudiante llena por primera vez la ficha de registro. El formato del carnet provisional es:

Iniciales de nombre y apellidos                      +                      año actual                      +                      numero correlativo de acuerdo a las iniciales

Para las iniciales, el máximo número de iniciales que se obtendrán serán 4. Es decir que aunque se tenga el caso de 3 nombres y 3 apellidos, esta función tomara las primeras dos palabras del nombre y las primeras dos palabras del apellido. Si al contrario, se tienen menos de dos nombres ó apellidos, se obtienen las iniciales de las palabras que se tengan.

El numero correlativo se calcula de acuerdo a las iniciales y de acuerdo al año, es decir si hay tutorados con las mismas iniciales en el mismo año, le pone un numero mas que el ultimo; si no hay ninguno con las mismas iniciales le asigna el correlativo =1. Para controlar los correlativos se auxilia en esta tabla

	_____
	_____

En la que se va almacenando las iniciales del nombre en el campo carnet1, el año en el campo anioi y el correlativo en el campo correlativo.

Los **parámetros** que se envían a la función son:

nombre: nombre del tutorado (ingresado en la sección A)

apellido: apellido del tutorado (ingresado en la sección A)

link: vinculo de conexión a la base de datos.

El código de la función:

```
function carnets($EXT_nombre, $EXT_apellido, $EXT_link ){
    $EXT_nombre=strtoupper(trim($EXT_nombre));
    $EXT_apellido=strtoupper(trim($EXT_apellido));
    $EXT_elanio=date('Y');//el año
    $EXT_caracteres1 = preg_split("/[\\s,]+/", $EXT_nombre,2);
    $EXT_caracteres2 = preg_split("/[\\s,]+/", $EXT_apellido,2);
    $EXT_car1="";
    foreach ($EXT_caracteres1 as $EXT_momento)
        $EXT_car1=$EXT_car1.strtoupper(substr($EXT_momento,0,1));
    foreach ($EXT_caracteres2 as $EXT_momento)
        $EXT_car1=$EXT_car1.strtoupper(substr($EXT_momento,0,1));

    $EXT_selectC="SELECT * FROM auxcarnet WHERE carnet1='".$EXT_car1."'
                AND anioi=".$EXT_elanio;
    $EXT_resultadoC=mysqli_query($EXT_link,$EXT_selectC);
    if(mysqli_num_rows($EXT_resultadoC)==1){
        while($EXT_row=mysqli_fetch_array($EXT_resultadoC)){
            $EXT_siguie=$EXT_row['correlativo']+1;
        }
    }elseif(mysqli_num_rows($EXT_resultadoC)==0){
        $EXT_siguie=1;
    }
    $EXT_siguiente= sprintf('%04d', $EXT_siguie);
    if($EXT_siguie==1){
        $EXT_ejecutar="insert into `sitebd`.`auxcarnet`
                    (carnet1, correlativo,
                    anioi)
                    values
                    ('".$EXT_car1."', ".$EXT_siguiente."',
                    ".$EXT_elanio."));";
    }else{
```

```

        $EXT_ejecutar="update `sitebd`.`auxcarnet` set
        correlativo=".$EXT_siguiente." WHERE carnet1='".$EXT_car1."'
        AND anioi=".$EXT_elanio;
    }
    $EXT_quier=mysqli_query($EXT_link,$EXT_ejecutar);
    if (!$EXT_quier){
        die('<span> Error al capturar en la Base de
        datos:'.mysqli_error($EXT_link).'

```

#### Descripción de su funcionamiento:

En la variable \$EXT\_caracteres1 se almacenan las primeras dos palabras del nombre, y en la variable \$EXT\_caracteres2 las primeras dos palabras del apellido.

```

$EXT_nombre=strtoupper(trim($EXT_nombre));
$EXT_apellido=strtoupper(trim($EXT_apellido));
$EXT_elanio=date('Y');//el año
$EXT_caracteres1 = preg_split("/[\\s,]+/", $EXT_nombre,2);
$EXT_caracteres2 = preg_split("/[\\s,]+/", $EXT_apellido,2);

```

En el foreach se recorren los vectores \$EXT\_caracteres1 y \$EXT\_caracteres2, que contienen como máximo dos palabras cada uno, en este recorrido se obtienen las iniciales de cada palabra y se almacena en la variable \$EXT\_car1.

```

$EXT_car1="";
foreach ($EXT_caracteres1 as $EXT_momento)
    $EXT_car1=$EXT_car1.strtoupper(substr($EXT_momento,0,1));
foreach ($EXT_caracteres2 as $EXT_momento)
    $EXT_car1=$EXT_car1.strtoupper(substr($EXT_momento,0,1));

```

Después de haber obtenido las iniciales que le corresponden a esta persona, se consulta en la base de datos la tabla auxcarnet para obtener el último correlativo otorgado a una persona con las mismas iniciales. Si no se encuentra ningún registro, se **insertara** en la tabla con el

correlativo de valor uno. Se utiliza la función `sprintf`, para asignarle un formato de cuatro dígitos a cada correlativo, es decir se rellena con ceros a la izquierda hasta lograr tener cuatro dígitos. Si las iniciales ya se encontraban en la tabla `auxcarnet` en el mismo año, se procede a **actualizar** la tabla con el nuevo último correlativo.

```

$EXT_selectC="SELECT * FROM auxcarnet WHERE carnet1='".$EXT_car1."'
                AND anioi=".$EXT_elanio;
$EXT_resultadoC=mysqli_query($EXT_link,$EXT_selectC);
if(mysqli_num_rows($EXT_resultadoC)==1){
    while($EXT_row=mysqli_fetch_array($EXT_resultadoC)){
        $EXT_siguie=$EXT_row['correlativo']+1;
    }
}elseif(mysqli_num_rows($EXT_resultadoC)==0){
    $EXT_siguie=1;
}
$EXT_siguiente= sprintf('%04d', $EXT_siguie);
if($EXT_siguie==1){
    $EXT_ejecutar="insert into `sitebd`.`auxcarnet`
                (carnet1, correlativo,
                anioi)
                values
                ('".$EXT_car1."', ".$EXT_siguiente.",
                ".$EXT_elanio.")";
}else{
    $EXT_ejecutar="update `sitebd`.`auxcarnet` set
                correlativo=".$EXT_siguiente." WHERE carnet1='".$EXT_car1."' AND
                anioi=".$EXT_elanio;
}
$EXT_quier=mysqli_query($EXT_link,$EXT_ejecutar);
if (!$EXT_quier){
    die('<span> Error al capturar en la Base de
        datos:'.mysqli_error($EXT_link).'

```

Esta función retorna el valor del Carnet provisional generado.

## Funciones cambiar\_carnet y cambiar\_Usu

Estas funciones sirven cuando se cambia carnet a los tutorados ó cuando se cambia el usuario (correo electrónico institucional) de una persona.

Con estas funciones se actualizan los campos relacionados (con el Usuario ó Carnet) en otras tablas de la base de datos sitebd.

Para el caso del cambio de carnet, son todas las tablas auxiliares las que se deben modificar.

```
function cambiar_carnet($EXT_antiguocarnet,$EXT_nuevocarnet,$EXT_link){
    ....definición de la función...
}
```

Los **parámetros** de esta función son

**antiguocarnet**: El carnet provisional que se le asigno por medio de SITE

**nuevocarnet**: El nuevo carnet, asignado por la universidad.

**link**: vinculo de conexión a la base de datos.

La definición de la función es la siguiente:

```
function cambiar_carnet($EXT_antiguocarnet,$EXT_nuevocarnet,$EXT_link){
    $EXT_tbls__a_cambiar=array(`carreraopcional`,`causas_repro_alum`,`conoc_carrera_alum`,`direccion_1`,`direccion_2`,`discap_tut`,`enf_medi_seguro`,`expectativas_alum_laborales`,`expectativas_alum_ser`,`grados_academicos_alum`,`habilidades_alum`,`horarios_clases_alumn`,`horarios_trabajo_alumn`,`inst_de_servicio_alum`,`motivo_insc_alum`,`motivo_trab_alum`,`pasatiempo_alum`,`pensum_alumno`,`premios_alumn`,`razon_udb_alum`,`tutorado`);
    foreach($EXT_tbls__a_cambiar as $EXT_tablaact){
        $EXT_oracion="update `sitebd`.`$EXT_tablaact.` set
Carnet_Tut='".$EXT_nuevocarnet.'" where Carnet_Tut='".$EXT_antiguocarnet.'"";
        $EXT_quer=mysqli_query($EXT_link,$EXT_oracion);
        if (!$EXT_quer){
            die('<span> Error al capturar en la Base de datos: -
link'.mysqli_error($EXT_link).$EXT_link.</msgerror>');
        }
    }
    $EXT_oracion="update `sitebd`.`tutorado` set Car_ActSN='S' where
Carnet_Tut='".$EXT_nuevocarnet.'"";
    $EXT_query=mysqli_query($EXT_link,$EXT_oracion);
    if (!$EXT_query){
        die('<span> Error al capturar en la Base de datos: -
link'.mysqli_error($EXT_link).$EXT_link.</msgerror>');
```

```
}  
}
```

Explicación:

En un vector se almacenan todos los nombres de las tablas que contienen el campo `Carnet_Tut`, con el fin de actualizar todas esas tablas. El vector se llama `$EXT_tbls__a_cambiar`. Con un `foreach` se recorre dicho vector, en este recorrido se van actualizando todas las tablas y al final se actualiza la tabla tutorado donde además del campo `Carnet_Tut`, se actualiza el campo `Car_ActSN='S'`, para indicar que ya se realizó cambio de carnet, de esta forma ya no saldrá en la consola donde se selecciona a quien se le cambiara el carnet.

No se devuelve ningún valor.

```
function cambiar_Usu($antiguoUsu,$nuevoUsu,$link){  
    ....definición de la función...  
}
```

Los parámetros de esta función son los siguientes

antiguoUsu: Usuario de la persona que se modificara

nuevoUsu: El nuevo usuario que se le asignara a la persona que se modificara.

link: vinculo de conexión a la base de datos.

La definición de la función es la siguiente:

```
function cambiar_Usu($EXT_antiguoUsu,$EXT_nuevoUsu,$EXT_link){ //HP  
    $anio_tope=date("Y");  
    $anio_inicio=2006;  
    for($anio_momento=$anio_inicio;$anio_momento<=$anio_tope;$anio_momento++)  
    ){  
        for($ciclo_momento=1;$ciclo_momento<=3;$ciclo_momento++){  
            $EXT_complemento=convertCICLO2let($ciclo_momento)."_".  
            $anio_momento;  
            $EXT_oracion="update  
            `sitebd`.`reg_sesion_".$EXT_complemento."`  
            set Responsable='".$EXT_nuevoUsu."' "  
            where Responsable='".$EXT_antiguoUsu."'";  
            mysqli_query($EXT_link,$EXT_oracion);  
        }  
    }  
    //del for de los años  
    $EXT_oracion1="update `sitebd`.`tutorado` set Usu='".$EXT_nuevoUsu."' "  
    where Usu='".$EXT_antiguoUsu."'";  
    mysqli_query($EXT_link,$EXT_oracion1);
```

```
$EXT_oracion2="update `sitebd`.`historial_tutorado` set
responsable='".$EXT_nuevoUsu."' where
responsable='".$EXT_antiguoUsu."';
mysqli_query($EXT_link,$EXT_oracion2);
$EXT_oracion3="update `sitebd`.`personas` set
Usu='".$EXT_nuevoUsu."',Correo_Persona='".$EXT_nuevoUsu."@udb.edu.sv'
where Usu='".$EXT_antiguoUsu."';
mysqli_query($EXT_link,$EXT_oracion3);
} //Fin de función cambiar Usu
```

**Explicación:** reg\_sesion\_, tutorado, historial\_tutorado, personas

En esta función se actualizan las tablas reg\_sesion\_ii\_aaaa (de todos los ciclos), tutorado, historial\_tutorado y personas. Pues en todas ellas tienen relación con el campo Usu de la tabla personas.

No se devuelve ningún valor.

## Función nuevoCiclo\_Anio

Esta función se utiliza para crear las tablas necesarias para crear las tablas necesarias para almacenar las sesiones.

```
function nuevoCiclo_Anio($anio,$ciclo,$link){  
    ....definición de la función...  
}
```

Los parámetros:

anio : Año al que corresponde la creación del nuevo ciclo.

ciclo: ciclo que se desea crear en formato i, ii ó iii.

link: vinculo de conexión a la base de datos.

Las tablas que se crean son:

`reg\_sesion\_"\$ciclo."\_"\$anio."`

`resp\_area\_x\_ciclo\_"\$ciclo."\_"\$anio."`

`obs\_ciclo\_"\$ciclo."\_"\$anio."`

Donde \$ciclo está en formato i, ii ó iii

Y el \$anio esta en formato numérico, por ejemplo 2006.

La definición de la función es la siguiente:

```
function nuevoCiclo_Anio($EXT_anio,$EXT_ciclo,$EXT_link){  
$EXT_select_reg_sesion="  
CREATE TABLE `reg_sesion_"$EXT_ciclo."_"$EXT_anio."` (  
    `Id_Sesion` int(5) unsigned NOT NULL auto_increment,  
    `Carnet_Tut` varchar(12) NOT NULL default " COMMENT 'Identificador de  
Tutorado',  
    `fecha_Sesion` date NOT NULL default '0000-00-00' COMMENT 'Fecha de la  
sesión',  
    `Obser_Sesion` longtext COMMENT 'Observaciones generales de la sesión',  
    `Recom_Sesion` longtext COMMENT 'Recomendaciones generales de la  
sesión',  
    `Estado` int(1) unsigned NOT NULL default '0' COMMENT 'Estado de la  
sesión',  
    `Responsable` varchar(32) default NULL COMMENT 'Tutor que ha ingresado la  
información de la sesión.'
```



```

`fecha_registro` date NOT NULL,
PRIMARY KEY (`Id_Sesion`),
KEY `FK_Reg_Sesion_".$EXT_ciclo."_".$EXT_anio."_1` (`Carnet_Tut`),
KEY `Responsable` (`Responsable`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Maestro de Registro de
Sesiones para el ciclo ".$EXT_ciclo." del ".$EXT_anio."';";
mysqli_query($EXT_link,$EXT_select_reg_sesion);

```

\$EXT\_select\_resp\_area="

```

CREATE TABLE `resp_area_x_ciclo_".$EXT_ciclo."_".$EXT_anio.` (
  `Id_Sesion` int(5) unsigned NOT NULL default '0' COMMENT 'Identificador de
  Sesión',
  `Respuesta` int(1) NOT NULL default '0' COMMENT 'Categoría seleccionada',
  `Id_Pregunta` int(3) unsigned NOT NULL default '0' COMMENT 'Identificador
  de pregunta',
  UNIQUE KEY `Identificador_Clave_resp_area` (`Id_Sesion`,`Id_Pregunta`),
  KEY `FK_Resp_Area_X_ciclo_".$EXT_ciclo."_".$EXT_anio."_2`
  (`Id_Pregunta`),
  KEY `Fk_Id_Sesion` (`Id_Sesion`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Mantenimiento de
Respuestas de las sesiones para el ciclo ".$EXT_ciclo." del ".$EXT_anio."';";
mysqli_query($EXT_link,$EXT_select_resp_area);

```

\$EXT\_select\_obs\_ciclo="

```

CREATE TABLE `sitebd`.`obs_ciclo_".$EXT_ciclo."_".$EXT_anio.` (
  `Id_Area` int(2) unsigned NOT NULL default '0' COMMENT 'Identificador de
  Aspecto Relevante',
  `Obs_Area_Sesion` longtext NOT NULL COMMENT 'Observación para
  Aspecto relevante',
  `Rec_Area_Sesion` longtext NOT NULL COMMENT 'Recomendación para
  Aspecto relevante',
  `Id_Sesion` int(5) unsigned NOT NULL default '0' COMMENT 'Identificador de
  sesión',
  UNIQUE KEY `Identificador_obs_ciclo` (`Id_Area`,`Id_Sesion`),

```

```
KEY `FK_Obs_Ciclo_".$EXT_ciclo."_.$EXT_anio."_2` (`Id_Sesion`),  
KEY `Fk_Id_Area` (`Id_Area`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Mantenimiento de  
Observaciones de sesiones para el ciclo ".$EXT_ciclo." del ".$EXT_anio."';  
mysqli_query($EXT_link,$EXT_select_obs_ciclo);  
}
```

## Funciones relacionadas con los ciclos y meses

- ⊖ convertCICLO2let
- ⊖ convertciclo2num
- ⊖ getciclo
- ⊖ getciclo\_num
- ⊖ meses\_de\_ciclo

```
function convertCICLO2let($ciclo){
    ....definición de la función...
}
function convertciclo2num($ciclo){
    ....definición de la función...
}
function getciclo($mes){
    ....definición de la función...
}
function getciclo_num($mes){
    ....definición de la función...
}
function meses_de_ciclo ($ciclo){
    ....definición de la función...
}
```

Definición de las funciones:

La función `convertCICLO2let` recibe como parámetro el ciclo en formato numérico y se devuelve el número en formato romano en minúscula.

```
function convertCICLO2let($EXT_ciclo){
    switch($EXT_ciclo){
        case 1:
            $EXT_ciclo_let="i";
            break;
        case 2:
            $EXT_ciclo_let="ii";
            break;
        case 3:
            $EXT_ciclo_let="iii";
    }
}
```

```

        break;
    default:
        $EXT_ciclo_let="o";
    }
    return $EXT_ciclo_let;
}

```

La función `convertciclo2num` recibe como parámetro el ciclo en formato Romano en mayúscula y devuelve el número de ciclo en formato numérico.

```

function convertciclo2num($EXT_ciclo){
    switch($EXT_ciclo){
        case "I":
            $EXT_ciclo_num=1;
            break;
        case "II":
            $EXT_ciclo_num=2;
            break;
        case "III":
            $EXT_ciclo_num=3;
            break;
    }
    return $EXT_ciclo_num;
}

```

La función `getciclo` recibe como parámetro el número de mes y devuelve a que ciclo pertenece en formato numérico romano en minúsculas.

```

function getciclo($EXT_EXT_mes_CICLO){//REGRESAR SITE ya(060806)
    if(($EXT_EXT_mes_CICLO>=1)&&($EXT_EXT_mes_CICLO<=5)){
        $EXT_EXT_ciclo_act="i";
    }
    elseif(($EXT_EXT_mes_CICLO>=6)&&($EXT_EXT_mes_CICLO<=10)){
        $EXT_EXT_ciclo_act="ii";
    }else{
        $EXT_EXT_ciclo_act="iii";
    }
    return $EXT_EXT_ciclo_act;
}

```

La función `getciclo_num` recibe como parámetro el número de mes y devuelve a que ciclo pertenece en formato numérico.

```

function getcilclo_num($EXT_EXT_mes_CICLO_NUM){
    if(($EXT_EXT_mes_CICLO_NUM>=1)&&($EXT_EXT_mes_CICLO_NUM<=5)){
        $EXT_EXT_ciclo_act=1;
    }
    elseif(($EXT_EXT_mes_CICLO_NUM>=6)&&($EXT_EXT_mes_CICLO_NUM<=10)){
        $EXT_EXT_ciclo_act=2;
    }else{
        $EXT_EXT_ciclo_act=3;
    }
    return $EXT_EXT_ciclo_act;
}

```

La función meses\_de\_ciclo recibe como parámetro el número de ciclo y devuelve los números de los meses que pertenecen a ese ciclo.

```

function meses_de_ciclo($EXT_EXT_ciclo1){//REGRESAR SITE ya(060806)
    switch($EXT_EXT_ciclo1){
        case 1:
            $EXT_EXT_meses=array(1,2,3,4,5);
            break;
        case 2:
            $EXT_EXT_meses=array(6,7,8,9,10);
            break;
        case 3:
            $EXT_EXT_meses=array(11,12);
            break;
    }
    return $EXT_EXT_meses;
}

```

## Función obtenersegundospasados

```
function obtenersegundospasados() {  
    ....definición de la función...  
}
```

```
function obtenersegundospasados() {  
    $EXT_iUsuarioActual=$_SESSION['sessionx_Usu'];  
    $EXT_inivel=$_SESSION['sessionx_nivel'];  
    $EXT_ihora=$_SESSION['sessionx_hora'];  
    $EXT_iminuto=$_SESSION['sessionx_minuto'];  
    $EXT_iseg=$_SESSION['sessionx_seg'];  
    $EXT_iyear=$_SESSION['sessionx_year'];  
    $EXT_imes=$_SESSION['sessionx_mes'];  
    $EXT_idia=$_SESSION['sessionx_dia'];  
    $EXT_delta2=time()-  
    mktime($EXT_ihora,$EXT_iminuto,$EXT_iseg,$EXT_imes,  
    $EXT_idia,$EXT_iyear);  
    $EXT_delta2=$EXT_delta2;//nuevo heisel 03 julio 2006  
    return $EXT_delta2;  
}
```

Esta función devuelve la diferencia de tiempo transcurrido desde que se ingreso al sistema un estudiante cuando va a llenar la ficha de registro por primera vez. El tiempo programado para completar la ficha de registro es de una hora.

## Función verificar\_para\_dar\_BAJA

```
function verificar_para_dar_BAJA (link){  
    ....definición de la función...  
}
```

Esta función se emplea para verificar si hay tutorados pendientes para dar de baja. Para esto, cuando un coordinador solicita dar de baja a un estudiante de otra facultad, se almacena en una tabla llamada historial\_tutorado.

Estructura



Dentro de la función, se consulta la tabla historial\_tutorado, los registros que sean de solicitar dar de baja (accion=1), donde los tutorados sean de la facultad del coordinador y que no hayan sido dados de baja aun (respuesta=0 ó null). Para saber que alguien solicito dar de baja a un tutorado el campo accion es 1 y al momento de dar de baja se genera un registro dentro de esta tabla cuyo identificador se almacena en el campo respuesta del registro donde se solicita, así para saber si ya se dio de baja al tutorado el campo respuesta no debe estar vacío.

historial\_tutorado

```
function verificar_para_dar_BAJA($EXT_B_link,$EXT_B_linkp){  
// para verificar si no hay TUTORADOS pra dar de baja  
    if(($_SESSION['sessionx_nivel']==5)&&  
        ($_SESSION['sessionx_recorridor_B']==1)){  
        $EXT_B_idcarrerasM=array();  
        $EXT_B_aux_correo_remitente=select Id_Escuelas from  
        personas where Usr=$_SESSION['sessionx_usr'];  
        $EXT_B_result_aux_correo_remitente=  
        mysqli_fetch_array(mysqli_query
```

**PK correlativo\_historial**

**fecha**

**accion**

**FK1 Carnet\_Tut**

**responsable**

**respuesta**

**descripcion**

```

        ($EXT_B_link,$EXT_B_aux_correo_remitente));
        $EXT_B_facultad_actual="select
nuke_facultad.descripcion,nuke_facultad.idfacultad from
`udbporta`.`nuke_facultad` inner join `udbporta`.`nuke_escuela` on
nuke_facultad.idfacultad=nuke_escuela.idfacultad where
nuke_escuela.idescuela='". $EXT_B_result_aux_correo_remitente['Id_Escuelas'
]+'"';

        $EXT_B_facultad_array=mysqli_fetch_array(mysqli_query($EXT_B_linkp,$
EXT_B_facultad_actual));
        $EXT_B_id_fac_actual=$EXT_B_facultad_array['idfacultad'];
        $EXT_B_sqlCA="Select
nuke_carrera.nc_idcarrera,nuke_carrera.nc_nombre from nuke_escuela inner
join nuke_carrera on nuke_escuela.idescuela= nuke_carrera.nc_idescuela
where nuke_escuela.idfacultad='". $EXT_B_id_fac_actual.'"';
        $EXT_B_resultCA=mysqli_query($EXT_B_linkp,$EXT_B_sqlCA);
        while($EXT_B_row=mysqli_fetch_array($EXT_B_resultCA))
            $EXT_B_idcarrerasM[]=$EXT_B_row['nc_idcarrera'];
        mysqli_free_result($EXT_B_resultCA);
$EXT_B_idcarrerasM=implode(',',$EXT_B_idcarrerasM);
        $EXT_B_mostrar="select
        historial_tutorado.correlativo_historial,
historial_tutorado.fecha,historial_tutorado.accion,
historial_tutorado.carnet,historial_tutorado.responsable,
historial_tutorado.respuesta, historial_tutorado.descripcion

        ,tutorado.Carnet_Tut,tutorado.Nombre_Tut,tutorado.Apellido_Tut,tutor
ado.IdCarrera
        from `sitebd`.`historial_tutorado` inner join
`sitebd`.`tutorado`
        on historial_tutorado.carnet=tutorado.Carnet_Tut
        where tutorado.IdCarrera IN ($EXT_B_idcarrerasM) and
historial_tutorado.accion=1 and historial_tutorado.respuesta=0 ";
        $EXT_B_mostrar.=" and tutorado.Estado=1";
        $EXT_B_mostrar.=" group by historial_tutorado.carnet";
        $EXT_B_mostrar_pasol=mysqli_query($EXT_B_link,$EXT_B_mostrar);
        if(mysqli_num_rows($EXT_B_mostrar_pasol)==0){
        }else{

```



```
//abrir la ventana CoordinadorDarBaja;
?>
    <script language="javascript">

        open_window('CoordinadorTareas.php?&ACCION=1',740,290,0,0,'Dar_Baja_
a_Estudiantes');

    </script>
    <?php
        }
        mysqli_free_result($EXT_B_mostrar_pasol);
    }
} //fin de la funcion que sirve para ver si hay tutorados para dar de baja
```

## Función verificar\_para\_dar\_ALTA

```
function verificar_para_dar_ALTA (link){
    ....definición de la función...
}
```

Esta función se emplea para verificar si hay tutorados pendientes para dar de alta. Para esto, cuando un coordinador solicita dar de baja a un estudiante de otra facultad, se almacena en una tabla llamada historial\_tutorado con el campo accion=1. Cuando se le da de baja al estudiante el campo respuesta se llena con el correlativo.

```
$EXT_A_mostrar="select historial_tutorado.correlativo_historial,
historial_tutorado.fecha,historial_tutorado.accion,
historial_tutorado.carnet,historial_tutorado.responsable,
historial_tutorado.respuesta, historial_tutorado.descripcion
,tutorado.Carnet_Tut,tutorado.Nombre_Tut,tutorado.Apellido_Tut,tutorado.Id
Carrera from `sitebd`.`historial_tutorado` inner join `sitebd`.`tutorado`
on historial_tutorado.carnet=tutorado.Carnet_Tut
where(tutorado.IdCarrera IS NULL or tutorado.IdCarrera='' or
tutorado.IdCarrera='0') and historial_tutorado.accion=1 and
historial_tutorado.respuesta<>0 and
historial_tutorado.responsable='$_SESSION[sessionx_Usu]";
$EXT_A_mostrar.=" and tutorado.Estado=1";
$EXT_A_mostrar.=" group by historial_tutorado.carnet";
```

Dentro de la función, se consulta la tabla historial\_tutorado [párrafo anterior], los registros que sean de solicitar dar de baja (accion=1) y que hayan sido dados de baja (respuesta diferente de cero), además el carnet correspondiente a cada registro encontrado, no debe tener carrera asignada y el responsable de la solicitud debe ser el coordinador que esta realizando esta consulta.

Definición de la función

```
function verificar_para_dar_ALTA($EXT_A_link){
// para verificar si no hay TUTORADOS pra dar de ALTA

    if((($_SESSION['sessionx_nivel']==5)&&($_SESSION['sessionx_recordar_A
']==1))){
```

```

$EXT_A_mostrar="select
historial_tutorado.correlativo_historial,
historial_tutorado.fecha,historial_tutorado.accion,
historial_tutorado.carnet,historial_tutorado.responsable,
historial_tutorado.respuesta,
historial_tutorado.descripcion,
tutorado.Carnet_Tut,tutorado.Nombre_Tut,
tutorado.Apellido_Tut,tutorado.IdCarrera
from `sitebd`.`historial_tutorado` inner join
`sitebd`.`tutorado`
on historial_tutorado.carnet=tutorado.Carnet_Tut
where(tutorado.IdCarrera IS NULL or tutorado.IdCarrera=''
or tutorado.IdCarrera='0') and historial_tutorado.accion=1
and historial_tutorado.respuesta<>0 and
historial_tutorado.responsable='$_SESSION[sessionx_Usu]';
$EXT_A_mostrar.=" and tutorado.Estado=1";
$EXT_A_mostrar.=" group by historial_tutorado.carnet";

$EXT_A_mostrar_pasol=mysqli_query($EXT_A_link,$EXT_A_mostrar);
if(mysqli_num_rows($EXT_A_mostrar_pasol)==0){
}else{
    //abrir la ventana CoordinadorDarBaja;
    ?>
    <script language="javascript">
    open_window('CoordinadorTareas.php?
    &ACCION=2',740,290,300,0,'Dar_Alta_a_Estudiantes');
    </script>
    <?php
}
mysqli_free_result($EXT_A_mostrar_pasol);
}
}

```

## Función verificar\_para\_ASIG\_TUTOR

```
function verificar_para_ASIG_TUTOR (link){  
    ....definición de la función...  
}
```

Esta función se emplea para verificar si hay tutorados pendientes para asignar tutor, esto puede ser cuando se ha cambiado de carrera y facultad un tutorado.

```
$EXT_AT_mostrar="select      historial_tutorado.correlativo_historial,  
historial_tutorado.fecha,historial_tutorado.accion,  
historial_tutorado.carnet,historial_tutorado.responsable,  
historial_tutorado.respuesta, historial_tutorado.descripcion,  
tutorado.Carnet_Tut,tutorado.Nombre_Tut,tutorado.Apellido_Tut,  
tutorado.IdCarrera from `sitebd`.`historial_tutorado` inner join  
`sitebd`.`tutorado` on historial_tutorado.carnet=tutorado.Carnet_Tut  
where tutorado.IdCarrera IN ($EXT_AT_idcarrerasM) and  
historial_tutorado.accion=3 and (historial_tutorado.respuesta=0 or  
historial_tutorado.respuesta is null)";  
$EXT_AT_mostrar.=" and tutorado.Estado=1 and tutorado.Usu='';  
$EXT_AT_mostrar.=" group by historial_tutorado.carnet";
```

Dentro de la función, se consulta la tabla `historial_tutorado` [párrafo anterior], los registros que sean de cambio de carrera (`accion=3`) y que el campo `respuesta` sea igual a cero o null (no se ha retomado por el coordinador para asignarle tutor), además la carrera del carnet correspondiente a cada registro encontrado debe pertenecer a una de las carreras de la facultad del coordinador. El procedimiento para conocer las escuelas de la facultad del tutor actual es el siguiente:

Primero se selecciona el identificador de la escuela a la que pertenece el coordinador y lo se almacena en la variable `$EXT_AT_result_aux_correo_remitente`.

```
$EXT_AT_idcarrerasM=array();  
$EXT_AT_aux_correo_remitente="Select Id_Escuelas from personas where  
Usu='$_SESSION[sessionx_Usu]';  
$EXT_AT_result_aux_correo_remitente=mysqli_fetch_array(mysqli_query($EXT_A  
T_link,$EXT_AT_aux_correo_remitente));
```

Luego se obtiene el identificador de la facultad en la variable \$EXT\_AT\_id\_fac\_actual

```
$EXT_AT_facultad_actual="select
nuke_facultad.descripcion,nuke_facultad.idfacultad from
`udbporta`.`nuke_facultad` inner join `udbporta`.`nuke_escuela` on
nuke_facultad.idfacultad=nuke_escuela.idfacultad where
nuke_escuela.idescuela='". $EXT_AT_result_aux_correo_remitente['Id_Escuelas
'] ."'";
$EXT_AT_facultad_array=mysqli_fetch_array(mysqli_query($EXT_AT_linkp,$EXT_
AT_facultad_actual));
$EXT_AT_id_fac_actual=$EXT_AT_facultad_array['idfacultad'];
```

Ahora en el vector \$EXT\_AT\_idcarrerasM se almacenan las carreras que pertenecen a la facultad correspondiente al coordinador actual

```
$EXT_AT_sqlCA="Select nuke_carrera.nc_idcarrera,nuke_carrera.nc_nombre
from nuke_escuela inner join nuke_carrera on nuke_escuela.idescuela=
nuke_carrera.nc_idescuela where
nuke_escuela.idfacultad='". $EXT_AT_id_fac_actual ."'";
$EXT_AT_resultCA=mysqli_query($EXT_AT_linkp,$EXT_AT_sqlCA);
while($EXT_AT_row=mysqli_fetch_array($EXT_AT_resultCA))
    $EXT_AT_idcarrerasM[]=$EXT_AT_row['nc_idcarrera'];
$EXT_AT_idcarrerasM=implode(',',$EXT_AT_idcarrerasM);
```

**Definición de la función**

```
function verificar_para_ASIG_TUTOR($EXT_AT_link,$EXT_AT_linkp){
// para verificar si no hay TUTORADOS QUE SE HAN CAMBIADO DE CARRERA
    if(($_SESSION['sessionx_nivel']==5)&&($_SESSION['sessionx_recordar_A
T']==1)){
        $EXT_AT_idcarrerasM=array();
        $EXT_AT_aux_correo_remitente="Select Id_Escuelas from
personas where Usu='$_SESSION[sessionx_Usu]";
        $EXT_AT_result_aux_correo_remitente=mysqli_fetch_array(
mysqli_query($EXT_AT_link,$EXT_AT_aux_correo_remitente));
        $EXT_AT_facultad_actual="select
nuke_facultad.descripcion,nuke_facultad.idfacultad from
```

```

`udbporta`.`nuke_facultad` inner join
`udbporta`.`nuke_escuela` on
nuke_facultad.idfacultad=nuke_escuela.idfacultad where
nuke_escuela.idescuela='". $EXT_AT_result_aux_correo_remitente[
'Id_Escuelas']. "'";
$EXT_AT_facultad_array=mysqli_fetch_array(
mysqli_query($EXT_AT_linkp,$EXT_AT_facultad_actual));
$EXT_AT_id_fac_actual=$EXT_AT_facultad_array['idfacultad'];
$EXT_AT_sqlCA="Select nuke_carrera.nc_idcarrera,
nuke_carrera.nc_nombre from nuke_escuela inner join
nuke_carrera on nuke_escuela.idescuela=
nuke_carrera.nc_idescuela where
nuke_escuela.idfacultad='". $EXT_AT_id_fac_actual. "'";
$EXT_AT_resultCA=mysqli_query($EXT_AT_linkp,$EXT_AT_sqlCA);
while($EXT_AT_row=mysqli_fetch_array($EXT_AT_resultCA))
    $EXT_AT_idcarrerasM[]=$EXT_AT_row['nc_idcarrera'];
$EXT_AT_idcarrerasM=implode(',',$EXT_AT_idcarrerasM);

$EXT_AT_mostrar="select
historial_tutorado.correlativo_historial,
historial_tutorado.fecha,historial_tutorado.accion,
historial_tutorado.carnet,historial_tutorado.responsable,
historial_tutorado.respuesta,
historial_tutorado.descripcion,
tutorado.Carnet_Tut,tutorado.Nombre_Tut,tutorado.Apellido_Tut,
tutorado.IdCarrera
from `sitebd`.`historial_tutorado` inner join
`sitebd`.`tutorado`
on historial_tutorado.carnet=tutorado.Carnet_Tut
where tutorado.IdCarrera IN ($EXT_AT_idcarrerasM)
and historial_tutorado.accion=3 and
(historial_tutorado.respuesta=0 or
historial_tutorado.respuesta is null)";
$EXT_AT_mostrar.=" and tutorado.Estado=1 and
tutorado.Usu=' '"; //modificado Heisel domingo agosto 20
$EXT_AT_mostrar.=" group by historial_tutorado.carnet";
$EXT_AT_mostrar_pasol=mysqli_query($EXT_AT_link,$EXT_AT_mostrar);

```

```

        if(mysqli_num_rows($EXT_AT_mostrar_pasol)==0){
        }else{
            //abrir la ventan                                ?>
            <script language="javascript">
            open_window('CoordinadorTareas.php?
            &ACCION=3',780,290,450,0,'Asignar_tutor_a_Estudiantes');

            </script>
            <?php
        }
        mysqli_free_result($EXT_AT_mostrar_pasol);
    }
} //fin de la funcion

```

## **Función verificar tareas coord**

```

function verificar_tareas_coord (link){
    ....definición de la función...
}

```

```

function verificar_tareas_coord($link_site,$link_p){
    verificar_para_dar_BAJA($link_site,$link_p);
    verificar_para_dar_ALTA($link_site);
    verificar_para_ASIG_TUTOR($link_site,$link_p);
}

```

Esta función se encarga de verifica si el coordinador tiene tareas pendientes, tal como la asignación de tutor a un tutorado que recién ha emigrado a su facultad.

En esta función se llaman a las 3 funciones anteriores.

Si tiene alguna tarea pendiente, se muestra que es lo que tiene pendiente, esto se realiza dentro de cada una de las funciones.

## Función ZAVEZEZIONG

La función ZAVEZEZIONG se emplea para almacenar los datos generales de las sesiones.

```
function ZAVEZEZIONG ($link, $observacionesgenerales, $ recomendacionesgenerales,
$fecha_Sesion, $fecha_registro){
    ....definición de la función...
```

Como primer paso, se identifica a cual tabla debe almacenarse la sesión, esto se hace de acuerdo al ciclo en el que se quiere registrar la sesión.

```
    $EXT_dia=dia_s($EXT_fecha_Sesion);
    $EXT_mes=mes_s($EXT_fecha_Sesion);
    $EXT_year=anio_s($EXT_fecha_Sesion);
    $EXT_ciclo_act=getciclo($EXT_mes);
    $EXT_anioact=$EXT_year;
    $EXT_complemento="_".$EXT_ciclo_act."_".$EXT_anioact;
```

Como se puede ver, se auxilia de la función `getciclo`, `dia_s`, `mes_s` y `anio_s`. La variable `$EXT_fecha_Sesion` es la fecha en la que se realizó la sesión, esta fecha es seleccionada por el tutor que está ingresando la sesión al sistema.

Luego se consulta la base de datos para conocer si esta sesión ya existía ó si hay que insertarla a la base de datos, de acuerdo a la fecha seleccionada por el tutor, es decir se consulta si hay una sesión para el tutorado en la fecha indicada.

```
$EXT_consulta_fecha_carnet="SELECT * FROM
`sitedb`.`reg_sesion".$EXT_complemento." WHERE
fecha_Sesion='".$EXT_fecha_Sesion.'" and Carnet_Tut='".$EXT_Carnet_Tut.'";
```

Se tiene esta variable `$_SESSION['sessionx_ID_S_ACT']`, que es donde se almacena el identificador de sesión al crearla.

Si la sesión no existía de acuerdo a la fecha seleccionada por el tutor, pueden ocurrir dos casos: Que se tenga que insertar en la tabla ó que el tutor haya modificado la fecha. Para conocer cual de los casos se trata se verifica la variable

`$_SESSION['sessionx_ID_S_ACT']`. Si esa variable está asignada y el valor es diferente de cero entonces es el caso en el que hay que actualizar en la tabla (tutor modificó la fecha seleccionada); en el caso contrario, donde la variable de sesión es nula ó igual a cero, hay que ingresar un nuevo registro a la tabla.



Siempre que se actualiza ó inserta se asigna a \$\_SESSION['sessionx\_ID\_S\_ACT'] el identificador de la sesión almacenada

```
function ZAVEZEZIONG($EXT_link, $EXT_observacionesgenerales,
$EXT_recomendacionesgenerales, $EXT_fecha_Sesion, $EXT_fecha_registro){
//EXT_fecha_Sesion es donde se almacena la fecha seleccionada por el
//tutor, que es cuando fue realizada la sesión
//EXT_fecha_registro es donde se almacena la fecha en la que ha sido
//ingresada la sesión al sistema
    $EXT_dia=dia_s($EXT_fecha_Sesion);
    $EXT_mes=mес_s($EXT_fecha_Sesion);
    $EXT_year=anio_s($EXT_fecha_Sesion);
    $EXT_ciclo_act=getcilclo($EXT_mes);
    //Para poner en formato i,ii,iii

    $EXT_anioact=$EXT_year;
    $EXT_complemento="_"$EXT_ciclo_act."_"$EXT_anioact;
    $_SESSION['sessionx_fecha_Sesion']=$EXT_fecha_Sesion;
    $EXT_Carnet_Tut=$_SESSION['sessionx_carnettutorado'];

    $EXT_Obser_Sesion=$EXT_observacionesgenerales;
    $EXT_Recom_Sesion=$EXT_recomendacionesgenerales;
    //Consulta para verificar si se va a insertar
    //o a modificar una sesión
    $EXT_consulta_fecha_carnet="SELECT * FROM
`sitebd`.`reg_sesion".$EXT_complemento."` WHERE
fecha_Sesion='".$EXT_fecha_Sesion.'" and
Carnet_Tut='".$EXT_Carnet_Tut.'"";
    $EXT_res_consulta_fecha_carnet=mysqli_query($EXT_link,
$EXT_consulta_fecha_carnet);

    if(mysqli_num_rows($EXT_res_consulta_fecha_carnet)==0){
//Si el resultado es cero filas es porque no se ha ingresado al
//sistema ó porque a una ya ingresada, se le ha modificado
//la fecha cuando fue realizada la sesión
//Si $_SESSION['sessionx_ID_S_ACT'] es igual a cero o no esta
//set, es porque no se ha almacenado en este instante
```

// es decir no es una modificación, si no que se debe insertar a  
//la tabla un nuevo registro  
//y almacenar la variable de sesión "sesionx\_ID\_S\_ACT"  
//Por otro lado si ya se tiene esa variable, entonces se debe  
//actualizar el registro ya existente, pues lo que ha sucedido e  
// es que se le ha modificado la fecha de realización de la  
//sesion.

```

if(($_SESSION['sessionx_ID_S_ACT']!=0) ||
    (!isset($_SESSION['sessionx_ID_S_ACT'])))
){
    $EXT_Sql1="INSERT INTO
    `sitebd`.`reg_sesion".$EXT_complemento."`
    ( `Id_Sesion`, `Carnet_Tut`, `fecha_Sesion`,
    `Obser_Sesion`,
    `Recom_Sesion`,`Responsable`,`fecha_registro`)
    values(0, '".$EXT_Carnet_Tut."', '".$EXT_fecha_Sesion."',
    '".$EXT_Obser_Sesion."', '".$EXT_Recom_Sesion."', '".$_SESSION['sessionx_Usu']."' ,
    '".$EXT_fecha_registro."");
    mysqli_query($EXT_link,$EXT_Sql1);
    $EXT_idsesionactualizar=mysqli_insert_id($EXT_link);
    $_SESSION['sessionx_ID_S_ACT']=$EXT_idsesionactualizar;
}
else{
    $EXT_Sql1="update
    `sitebd`.`reg_sesion".$EXT_complemento."` set
    `Carnet_Tut`='".$EXT_Carnet_Tut."',
    `fecha_Sesion`='".$EXT_fecha_Sesion."',
    `Obser_Sesion`='".$EXT_Obser_Sesion."',
    `Recom_Sesion`='".$EXT_Recom_Sesion.'" where
    `Id_Sesion`='".$_SESSION['sessionx_ID_S_ACT']."'";
    mysqli_query($EXT_link,$EXT_Sql1);
}

```

}else{

//Cuando el resultado es mayor a cero filas es porque este registro ya existía, //por  
lo tanto se modificara el registro

//y se almacena la variable de sesión sessionx\_ID\_S\_ACT

```

while($EXT_row=mysqli_fetch_array($EXT_res_consulta_fecha_carnet))
    $EXT_idsesionactualizar=$EXT_row['Id_Sesion'];
$EXT_Sql1="update `sitebd`.`reg_sesion".$EXT_complemento."` set
`Id_Sesion`='".$EXT_idsesionactualizar."',
`Carnet_Tut`='".$EXT_Carnet_Tut."',
`fecha_Sesion`='".$EXT_fecha_Sesion."',
`Obser_Sesion`='".$EXT_Obser_Sesion."',
`Recom_Sesion`='".$EXT_Recom_Sesion.'" where
`Id_Sesion`='".$EXT_idsesionactualizar.'";
mysqli_query($EXT_link,$EXT_Sql1);
$_SESSION['sessionx_ID_S_ACT']=$EXT_idsesionactualizar+0;

}
}

```

## **Función ZAVEZEZIONN**

Esta función se utiliza cuando se almacena cada una de las áreas al momento de completar la sesión.

```
function ZAVEZEZIONN ($link, $otrarazon, $OR, $mirazon, $Id_Area, $obsareasesion,
$recareasesion){
    ....definición de la función...
}
```

Parámetros:

link: vinculo de conexión a la base de datos.

otrarazon: Es la descripción de una dificultad encontrada y que quiere que aparezca como una dificultad.

oR: conjunto de dificultades activas identificadas para cada estudiante

mirazon: Conjunto de las dificultades con su respectiva respuesta, que se ha(n) seleccionado.

Id\_Area: El Área que se almacenara (Área Académica, Área Institucional, etc)

obsareasesion: la observación hecha para el área en cuestión

recareasesion; la recomendación hecha para el área en cuestión

```
function ZAVEZEZIONN($EXT_link, $EXT_otrarazon, $EXT_OR, $EXT_mirazon,
$EXT_Id_Area, $EXT_fecha_Sesion=$SESSION['sessionx_fecha_Sesion']{
    $EXT_dia=dia_s($EXT_fecha_Sesion);
    $EXT_mes=mes_s($EXT_fecha_Sesion);
    $EXT_year=anio_s($EXT_fecha_Sesion);
    $EXT_ciclo_act=getcilclo($EXT_mes);

    $EXT_anioact=$EXT_year;
    $EXT_complemento="_"$EXT_ciclo_act."_"$EXT_anioact;
    $EXT_Carnet_Tut=$SESSION['sessionx_carnettutorado'];

    $EXT_consulta_fecha_carnet="SELECT * FROM
`sitebd`.`reg_sesion".$EXT_complemento.` WHERE
fecha_Sesion='".$EXT_fecha_Sesion.'" and
Carnet_Tut='".$EXT_Carnet_Tut.'";
```

```
$EXT_res_consulta_fecha_carnet=mysqli_query($EXT_link,$EXT_consulta_fecha_carnet);
```

```
if(mysqli_num_rows($EXT_res_consulta_fecha_carnet)>0){  
while($EXT_row=mysqli_fetch_array($EXT_res_consulta_fecha_carnet))  
    $EXT_idsesionactualizar=$EXT_row['Id_Sesion'];
```

//Aquí se obtiene el identificador de la sesión que se va a actualizar, en

**\$EXT\_idsesionactualizar**

```
    $EXT_i=0;
```

//Comienza el almacenamiento de observación y recomendación si había alguno

Se verifica si hay observaciones ó recomendaciones para ingresar en esta área de la sesión.

Las observaciones y recomendaciones se almacenan en la tabla obs\_ciclo\_ii\_aaaa. Cada observación/recomendación que se ingresa a la tabla, se guarda con el identificador a que área pertenece esa observación/recomendación, así como también se almacena el identificador de la sesión en la cual se ingresó.

```
if((!empty($EXT_obsareasesion))||(!empty($EXT_recareasesion))){  
    $EXT_SqlArea="select * from  
    `sitebd`.`obs_ciclo".$EXT_complemento."` where  
    `Id_Sesion`='".$EXT_idsesionactualizar."' and  
    `Id_Area`='".$EXT_Id_Area."";  
    $EXT_resultadoArea=mysqli_query($EXT_link,$EXT_SqlArea);
```

SE verifica si se actualizara un registro existente ó si se insertara un nuevo registro.

```
if(mysqli_num_rows($EXT_resultadoArea)==0){  
    $EXT_Sql1="INSERT INTO  
    `sitebd`.`obs_ciclo".$EXT_complemento."` (Id_Area,  
    Obs_Area_Sesion, Rec_Area_Sesion, Id_Sesion) values  
    ('".$EXT_Id_Area."', '".$EXT_obsareasesion."',  
    '".$EXT_recareasesion."', '".$EXT_idsesionactualizar.'");
```

```
}else{  
    $EXT_Sql1="update `sitebd`.`obs_ciclo".$EXT_complemento."`  
    set `Id_Sesion`='".$EXT_idsesionactualizar.',  
    `Obs_Area_Sesion`='".$EXT_obsareasesion.',  
    `Rec_Area_Sesion`='".$EXT_recareasesion.',  
    `Id_Area`='".$EXT_Id_Area."` where  
    `Id_Sesion`='".$EXT_idsesionactualizar.'" and  
    `Id_Area`='".$EXT_Id_Area."";
```

```

    }
    mysqli_query($EXT_link,$EXT_Sql1);
}

```

//Finaliza el almacenamiento de observación y recomendación si había alguno  
// Comienza almacenamiento de respuestas dadas a las preguntas del área en cuestión  
Ahora se procede con las respuestas que se han dado a las dificultades que han sido posibles. En el vector EXT\_mi\_razon se tienen los identificadores de las preguntas en los índices y las respuestas dadas son los valores o respuestas otorgadas a las preguntas, en este bucle, \$EXT\_preg es el identificador de cada pregunta, y \$EXT\_resp es la respuesta brindada a cada pregunta.

```

foreach($EXT_mirazon as $EXT_preg=>$EXT_resp){
    $EXT_seleccarea="select
    resp_area_x_ciclo".$EXT_complemento.".Id_Sesion as
    id_d_sesion from
    `sitebd`.`resp_area_x_ciclo".$EXT_complemento."` where
    resp_area_x_ciclo".$EXT_complemento.".Id_Sesion=
    '".$EXT_idsesionactualizar.'" and
    Id_Pregunta='".$EXT_preg.'";
    $EXT_resulcarea=mysqli_query($EXT_link,$EXT_seleccarea);
    if(mysqli_num_rows($EXT_resulcarea)==0){
        $EXT_Sql1="INSERT INTO
        `sitebd`.`resp_area_x_ciclo".$EXT_complemento."`
        (Id_Sesion, Respuesta, Id_Pregunta) values
        ('".$EXT_idsesionactualizar."', '".$EXT_resp."',
        '".$EXT_preg.')";
        mysqli_query($EXT_link,$EXT_Sql1);
    }else{
        while($EXT_row=mysqli_fetch_array($EXT_resulcarea))
        $EXT_idsesionactualizar=$EXT_row['id_d_sesion'];
        $EXT_Sql1="update
        `sitebd`.`resp_area_x_ciclo".$EXT_complemento."` set
        `Id_Sesion`='".$EXT_idsesionactualizar."',
        `Respuesta`='".$EXT_resp."',
        `Id_Pregunta`='".$EXT_preg.'" where
        `Id_Sesion`='".$EXT_idsesionactualizar.'"and
        Id_Pregunta='".$EXT_preg.'";
        mysqli_query($EXT_link,$EXT_Sql1);
    }
}

```

```

    }
}
//Finaliza almacenamiento de respuestas dadas a las preguntas del área en cuestión
//Comienza actualización de dificultades personales encontradas.
Primero se ponen todas inactivas y luego en el próximo foreach se activan las
seleccionadas en esta sesión.

```

```

$setear="update `sitebd`.`dificultades` set Activa=0 where
Id_Area='". $EXT_Id_Area. "'
and Carnet_Tut='". $EXT_Carnet_Tut. "'";
mysqli_query($EXT_link,$setear);
foreach($EXT_OR as $id_Preg=>$activa){
    $setear_U="update `sitebd`.`dificultades` set Activa=1
where
Id_Area='". $EXT_Id_Area. "' and Id_Pregunta='". $id_Preg. "'
and Carnet_Tut='". $EXT_Carnet_Tut. "'";
mysqli_query($EXT_link,$setear_U);
}

```

```

//Finaliza actualización de dificultades personales encontradas
//Comienza Ingreso de otra dificultad que no se encuentra en el sistema para el estudiante
en estudio.

```

```

if(!empty($EXT_otrarazon)){
Primero se inserta en la tabla dificultades, para que salga exclusivamente para este
estudiante se le pone el carnet en el campo del carnet
$EXT_insertardificultad="insert into
`sitebd`.`dificultades` (Id_Area, Id_Pregunta,
Desc_Dificultad, Carnet_Tut,Activa) values
('". $EXT_Id_Area. "', 0, '". $EXT_otrarazon. "',
'". $EXT_Carnet_Tut. "', '1')";
$EXT_resinsertardificultad=mysqli_query
($EXT_link,$EXT_insertardificultad);
}

```

```

//finaliza el ingreso de otra dificultad
} //del if
} //de la función

```

## Función esta\_completa

```
function esta_completa ($link){
    ....definición de la función...
}

function esta_completa($EXT_link){
    $EXT_fecha_Sesion=$_SESSION['sessionx_fecha_Sesion'];
    $EXT_dia=dia_s($EXT_fecha_Sesion);
    $EXT_mes=mес_s($EXT_fecha_Sesion);
    $EXT_year=anio_s($EXT_fecha_Sesion);
    $EXT_ciclo_act=getcilclo($EXT_mes);

    $EXT_anioact=$EXT_year;
    $EXT_complemento="_".$EXT_ciclo_act."_".$EXT_anioact;
    $EXT_areasactuales=0;
    //Se busca en la tabla donde se almacenan las observaciones
    //se busca una fila que contenga una observación de la sesión
    //actual. Si se encuentra alguna fila se suma la variable
    // $EXT_areasactuales
    $EXT_seleccarea="select obs_ciclo".$EXT_complemento.".Id_Sesion
as id_d_sesion from `sitebd`.`obs_ciclo".$EXT_complemento."`
where
obs_ciclo".$EXT_complemento.".Id_Sesion=
'".$_SESSION['sessionx_ID_S_ACT'].'";

    $EXT_resulcareas=mysqli_query($EXT_link,$EXT_seleccarea);
    if(mysqli_num_rows($EXT_resulcareas)>0)
        $EXT_areasactuales++;

    //Se busca en la tabla donde se almacenan los datos generales de las sesiones
    //se verifica que el campo de la observación no este vacío
    //Si no esta vacío este campo se suma la variable
    // $EXT_areasactuales

    $EXT_Observ_gener=mysqli_fetch_row(mysqli_query($EXT_link,"select
Obser_Sesion from `sitebd`.`reg_sesion".$EXT_complemento."` where
Id_Sesion='".$_SESSION['sessionx_ID_S_ACT'].'"));
```



```
if(!empty($EXT_Observ_gener[0]))
    $EXT_areasactuales++;

//Al final si la variable $EXT_areasactuales es igual a cero, es porque
//no se ha encontrado ni una observación
if($EXT_areasactuales==0)
    return false;
else
    return true;

}
```