**SISTEMA DE CAI (CENTRO DE APRENDIZAJE DE IDIOMAS)**

**MANUAL TECNICO**

UNIVERSIDAD POLITECNICA DE VICTORIA

INGENIERIA EN TECNOLOGIAS DE LA INFORMACION

TECNOLOGIAS Y APLICACIONES WEB

M.S.I. MARIO HUMBERTO RODRIGUEZ CHAVEZ

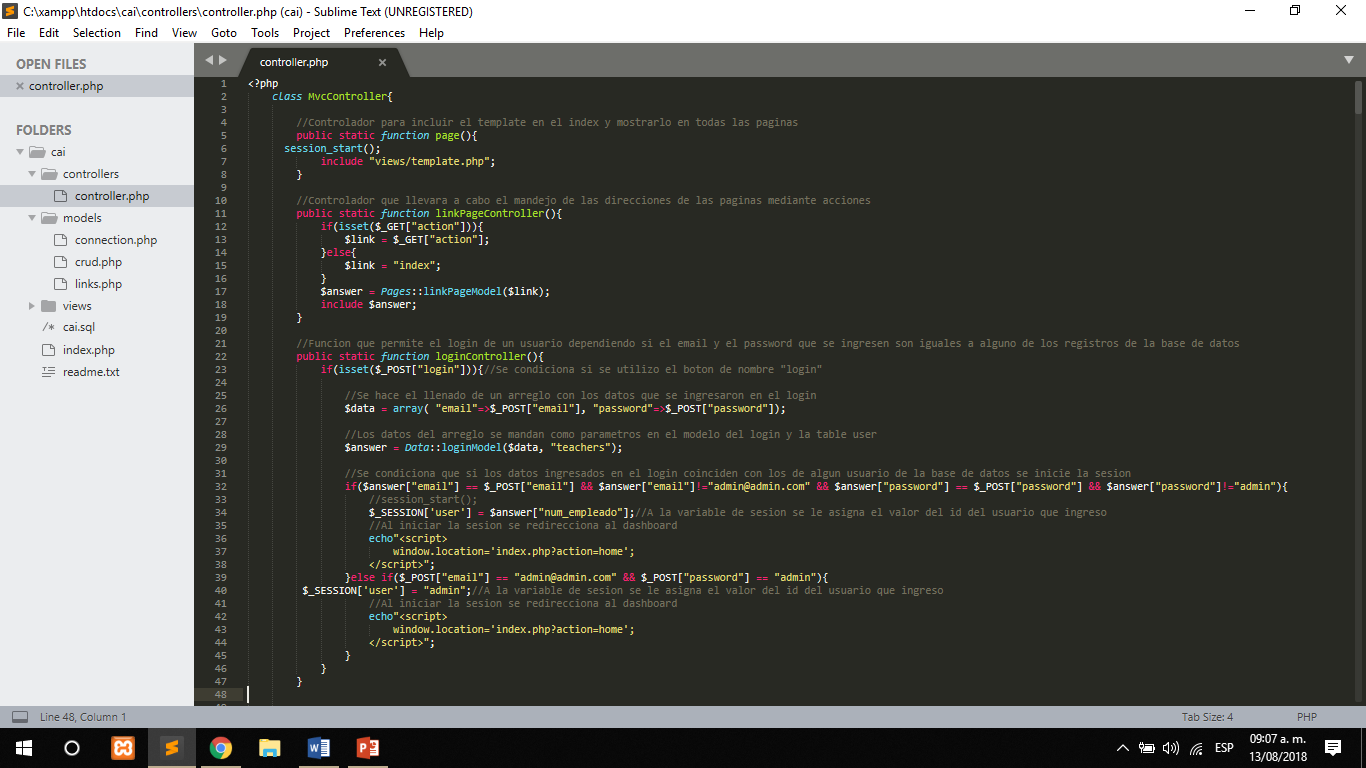
INTEGRANTES DE EQUIPO:

GOMEZ FLORES OSIEL

HERNANDEZ RODRIGUEZ MIGUEL ANGEL

CD. VICTORIA, TAMAULIPAS. MEXICO

Comenzando con la clase principal la cual es la clase “MvcController”, en dicha clase se encuentran todas las funciones (controladores) que se utilizan para que el sistema tenga un correcto funcionamiento

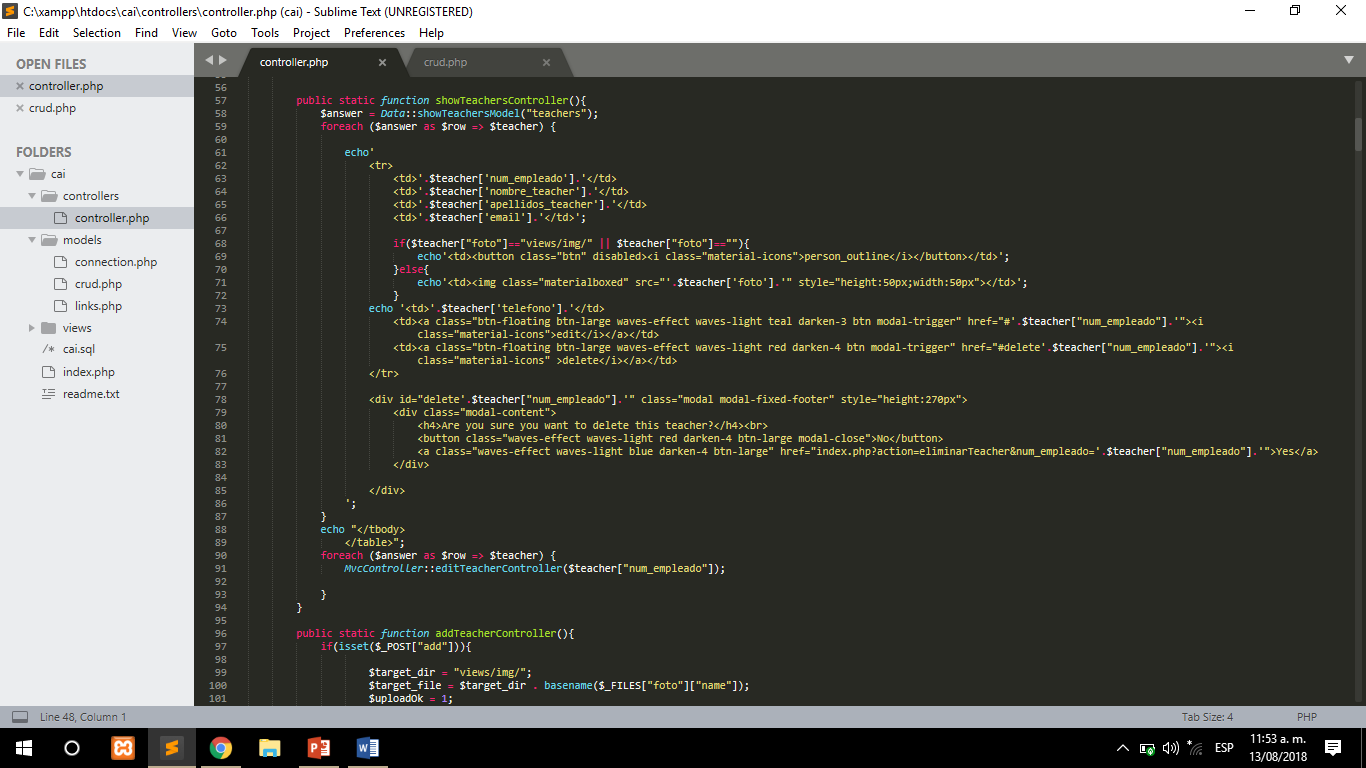


En esta captura hay 3 controladores: page(), linkPageController() y LoginController().

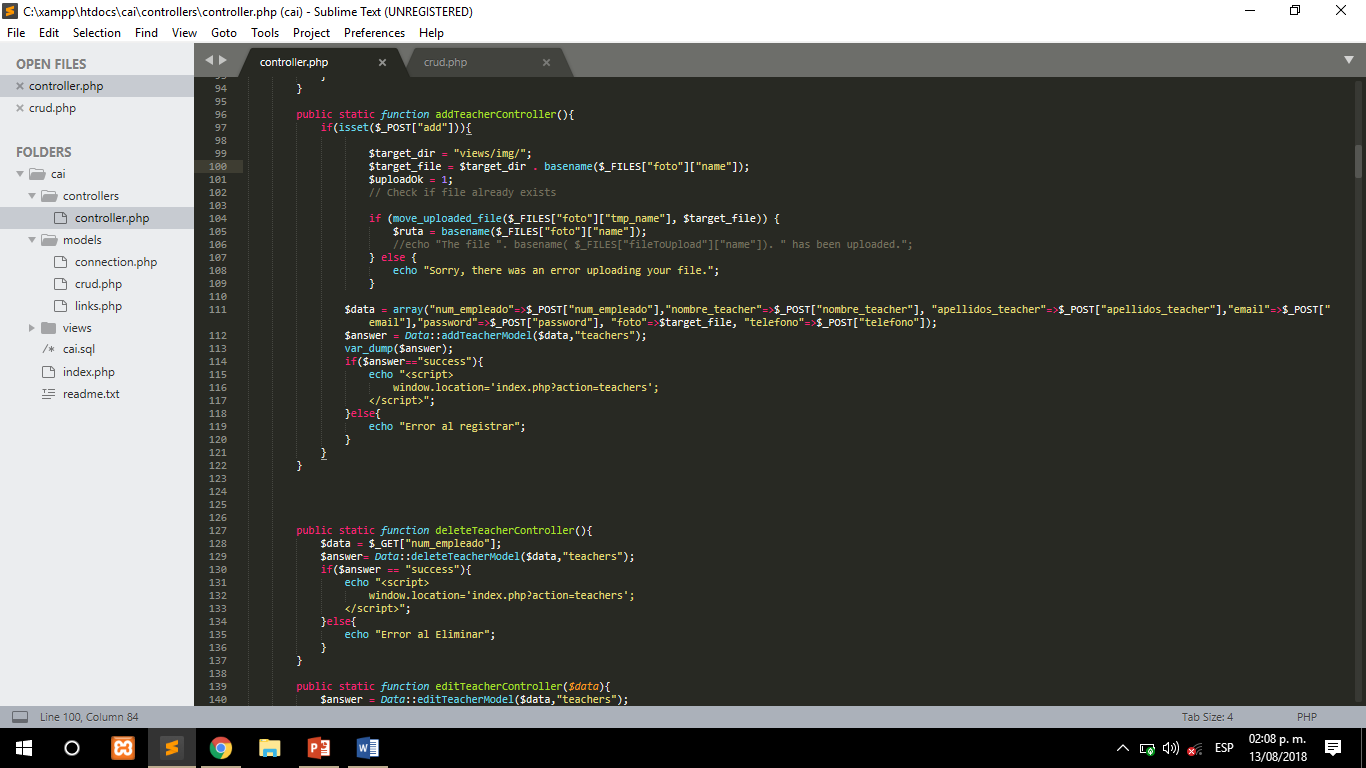
page: este controlador incluye donde sea llamado, la vista “template.php” mediante un “include()”.

linkPageController: este controlador obtiene mediante un “$\_GET[“action”]”, esto se obtiene de la url, obtiene el valor del “action” y lo asigna a una variable “$link”, si no existe “action” en la url, entonces se asigna “index” a la variable, y esa variable es mandada a un método del modelo el cual re-direcciona a una vista determinada.

loginController: este controlador realiza la función de iniciar sesión, si el botón “login” del formulario fue presionado o lanzado, esto lo hace mediante la obtención de los valores que el usuario llenó en el formulario de la vista de login, los cuales son el nombre de usuario y contraseña y los almacenamos en un array, y este array es mandado a un método del modelo donde se realizan todas las consultas a la base de datos, aquí se usa el método “Data::loginModel()” la cual recibe 2 parámetros, el array con el nombre de usuario y contraseña y el nombre de la tabla en la cual va a hacer la consulta, este método hace consulta en la tabla “teachers” y devuelve los datos del teacher donde coincidan los valores enviados, si los valores devueltos son iguales a los valores ingresados, es decir, si los datos coincidieron, entonces quiere decir que el usuario existe, por lo cual se inicia una variable “$\_SESSION[“user”]” y le asignamos el número de empleado del teacher y re-direcciona a la página principal del sitio poniendo un “action=home”. Si la consulta que devolvió el modelo no coincidió con los datos ingresados por el usuario, el sitio re-direcciona a la misma página de login.



ShowTeacherController: este controlador primeramente hace una consulta mediante el modelo “data::showTeacherModel” de la tabla “teachers” y almacena el resultado en una variable y después este resultado se imprime en una tabla con los resultados imprimiendo todos los campos de la tabla “teachers”, poniendo las facilidades para borrar y modificar, utilizando modales.



addTeacherController: este controlador hace la funcionalidad de agregar un nuevo teacher a la base de datos. Primeramente verifica si el botón “add” fue presionado para poder almacenar los datos del formulario llenado por el usuario, primeramente, almacena los datos de la imagen y la mueve a la carpeta de destino para las imágenes, después almacena en un array todos los datos del formulario como numero de emp,leado, nombre, apellidos, etc., y manda a un método del modelo “data::addTeacherModel”, el array con esos datos y el nombre de la tabla a insertar.

Según la respuesta del método del modelo, si es “success” entonces re-direcciona a la vista de “teachers” poniendo un “action” en la url de “teachers”, si no es “success” la respuesta del modelo, entonces muestra un mensaje de error en el registro del teacher.

deleteTeacherController: este controlador almacena en una variable un valor de la url obtenido por un “$\_GET[“num\_empleado”]” y esta variable se manda a un método del modelo llamado “data::deleteTeacherModel” junto con el nombre de la tabla de la base de datos par eliminar el teacher, la tabla “teachers” y según la respuesta del modelo, re-direcciona a la vista “teachers”