**Documentación de proyecto.**

**Grupo #9**

1. Agregar Platillo.
   1. Pane Agregar Platillo.

Se encarga de pedir al asistente de x restaurante los datos necesarios del platillo para almacenarlos.

1. Búsqueda.
   1. Búsqueda.

Busca los platos por nombre o descripción llamando al método que se encarga de mostrar la lista de platos.

1. Categoría.
   1. Categoría.

Clase con la descripción de las categorías, además de sobrescribir los métodos equals y hashcode.

* 1. Lista Categorías.

Ayuda a tener n categorías en el sistema.

1. Información Platos.
   1. Lista Platos.

Almacena todos los platos de x restaurante en un contenedor el cual ayuda a mostrarlos, recorriendo la lista de platos.

* 1. Pane Información Platos.

Al momento de dar click en un plato mostrado, el asistente es llevado a otra ventana donde se muestra la información referente al plato seleccionado. (Tipo, Servido, Imagen)

1. Inicio de Sesión.
   1. Pane Inicio Sesión.

Se encarga de validar que el usuario exista en el sistema y que la contraseña sea la correcta haciendo su respectiva validación, asimismo se encarga de manejar las sesiones, es decir valida que el usuario ingresado escoja su respectivo rol enviándolo a la sesión correspondiente.

1. Menú Bar Administrador.
   1. Pane Organiza Administrador.

Sesión del administrador. Página en construcción.

1. Menú Bar Asistente.
   1. Pane Organiza Asistente.

Sesión del asistente, página de inicio del asistente, muestra las opciones que el usuario asistente tiene.

1. Menú Bar Cliente.
   1. Pane Organize Cliente.

Sesión del cliente, página de inicio del cliente, muestra las opciones que el usuario cliente tiene.

1. Modelo.
   1. Persistencia.

Maneja los archivos csv( usuario, restaurante, platos) del sistema, sirviendo como enlace entre los archivos y el sistema para su respectiva manipulación.

* 1. Platos.

Clase que guarda los datos de los platos con sus respectivos setters y getters.

* 1. Proyecto.

Main del proyecto.

* 1. Restaurante.

Clase que guarda los datos de los restaurantes con sus respectivos setters y getters.

* 1. Usuario.

Clase que guarda los datos de los usuarios con sus respectivos setters y getters.