

Pontificia Universidad Javeriana – Sede Bogotá

**Clase:** Desarrollo web

**Profesor:** Juan Sebastián Angarita Torres

Ingeniería de sistemas

---

## **Proyecto: Aplicativo web para restaurante**

### **Contexto**

Su equipo ha sido contratado por un restaurante que desea poder comenzar a digitalizar su negocio. El cliente desea una página web que promocione sus comidas, permita a los clientes realizar pedidos y a los trabajadores del restaurante monitorear el estado de estos.

### **Negocio**

El restaurante en los últimos años ha tomado gran popularidad, y aunque cuenta con una única sede la cantidad de pedidos ha crecido año tras año. Actualmente el restaurante maneja domicilios, sin embargo, estos se realizan únicamente con una metodología clásica, donde el cliente llama y lo atiende un trabajador el cual anota el pedido y todos los datos. Posteriormente de manera manual y algo desorganizada se completa el proceso de envío.

La página web espera facilitar varios de estos procesos. Para realizar un pedido el cliente debe crear una cuenta donde suministra sus datos personales (nombre, apellido, correo, contraseña, teléfono y dirección). Tras haber creado su cuenta, el cliente puede seleccionar uno o más productos que ofrece el restaurante, agregarlos a su carrito de compras y confirmar el pedido. Todo producto tiene un nombre, precio, descripción e imagen (representada en una URL). También los productos cuentan con adicionales, un mismo producto puede tener varios adicionales y un adicional pertenecer a varios productos. Los adicionales únicamente tienen un nombre y precio (pueden ser gratuitos). Es importante tener en cuenta que si el cliente quiere varias unidades de un mismo producto no se deben crear múltiples registros del mismo producto en la orden, sino que se debe crear un único registro que señale con un atributo adicional la cantidad que desea el cliente.

Puede suponer que las compras se realizan con pago contra entrega o usando datafono en la dirección de entrega, es decir, no es necesario realizar ningún pago a través de la página. Cuando el cliente confirma su pedido este se agrega a la base de pedidos activos del restaurante.

De manera paralela en el proceso de los domicilios existen los operadores, estos son trabajadores encargados de actualizar el estado de los pedidos y asignarles un domiciliario. Los operadores cuentan con nombre, usuario y contraseña para ingresar al sistema. En general todo pedido cuenta con un estado, este puede ser: recibido,

cocinando, enviado y entregado. Cuando un pedido está en estado de recibido se actualiza su fecha de creación y cuando es entregado su fecha de entrega. Adicionalmente cuando el envío pasa a estado de enviado se le debe asignar un domiciliario disponible. Los domiciliarios tienen nombre, celular, cedula e información acerca de si están disponibles para realizar un envío.

Finalmente está el administrador, este entra al sistema a través de su propio portal. En él puede realizar las operaciones de CRUD sobre los operadores, domiciliarios, productos y adicionales.

### Adicionales

- Es importante para el restaurante que se pueda consultar el historial general de pedidos y que también un cliente puede revisar sus pedidos pasados.
- Si el usuario ha seleccionado algunos productos y los ha agregado al carrito, al volver a entrar en la página debe poder ver los productos anteriormente seleccionados.
- No usar herencia.

### Requerimientos

- **Feature 00:** Defina que clase de comida va a vender su restaurante.
- **Feature 01:** Diagrama de Clases, entidad-relación y mockups + navegación.

AC1 Diagrama de clases	<b>Given</b> la información del negocio <b>Then</b> generar un <u>diagrama de clases (UML)</u> para cada una de las entidades. Este debe incluir cardinalidad, nombre de las relaciones y métodos.
AC2 Diagrama entidad-relación	<b>Given</b> la información del negocio <b>Then</b> generar un diagrama entidad-relación. Este debe mostrar PK, FK, campos que pueden ser nulos, cardinalidad, campos únicos.
AC3 Mockups	<b>Given</b> la información del negocio <b>Then</b> generar los mockups del diseño de la página. <b>And</b> Cada mockup debe estar asociado a una URL <b>And</b> los mockups se deben poder navegar de manera interactiva o mostrar cómo sería la navegación. <b>And</b> diferenciar a cuáles pantallas puede acceder cada uno de los tipos de usuario
AC4 Logo	<b>Given</b> la información del negocio <b>Then</b> crear un logo <b>And</b> crear un nombre
AC5 Colores	<b>Given</b> la información del negocio <b>Then</b> definir una paleta de colores

	<a href="https://www.youtube.com/watch?v=yYwEnLYT55c">https://www.youtube.com/watch?v=yYwEnLYT55c</a> (No es necesario usar HSB, ni tener una paleta compleja, lo importante es definir los colores de su aplicación, variaciones, y colores neutrales)
--	---

- **Feature 02:** Repositorio – Github

AC6 Repositorio	<b>Given</b> código generado durante el desarrollo <b>Then</b> crear un repositorio en Github público.  <b>And</b> generar una rama de “desarrollo” y otra “producción” o “master”. <sup>1</sup>
AC7 Repositorio	<b>Given</b> los entregables del feature 1 <b>Then</b> cargar los entregables en el repositorio.
AC8 Repositorio	<b>Given</b> los entregables del feature 1 <b>Then</b> personalizar el repositorio con información del negocio y del desarrollo

- **Feature 03:** Base de datos

AC9 Entidades	<b>Given</b> la información de cada componente <b>Then</b> generar las entidades en el sistema y en la BD.
AC10 Repositorios	<b>Given</b> cada una de las entidades <b>Then</b> generar repositorios
AC11 Datos prueba	<b>Given</b> la información de cada componente <b>Then</b> generar datos iniciales de prueba que permitan simular el funcionamiento real de la página. <sup>2</sup>

- **Feature 04:** Main page.

AC12 Landing	<b>Given</b> la información del negocio y los diseños generados <b>Then</b> generar una landing page para promocionar el restaurante.  <b>And</b> permitir navegar al login de usuario.
AC13 Login	<b>Given</b> el correo y contraseña <b>Then</b> permitir al usuario realizar login y acceder a su portal
AC14 Signup	<b>Given</b> la información de una persona <b>When</b> cliente no está registrado <b>Then</b> Registrar cliente en el sistema
AC14	<b>Given</b> la información de los productos

<sup>1</sup> La rama de producción será la que verá el cliente y calificará. En la rama de desarrollo suba y junte los cambios de las demás personas del grupo. Cuando todo esté funcional en la rama de desarrollo suba los cambios a producción. Puede crear más ramas si lo cree necesario.

<sup>2</sup> Para la entrega 1, solo es necesario cargar datos relacionados a cliente, producto y adicionales.

Menú	<b>Then</b> mostrar la información de estos en tarjetas.
AC15 Detalle producto	<b>Given</b> el identificador de un producto <b>Then</b> mostrar sus detalles <b>And</b> permitir seleccionar sus posibles adicionales <b>And</b> permitir agregar al carrito de compras

- **Feature 05:** Portal de cliente.

AC16 Pedidos activos	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá ver sus pedidos activos.
AC17 Historial	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá ver su historial de pedidos.
AC18 Información usuario	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá realizar leer y modificar su información personal.

- **Feature 06:** Pedido cliente

AC19 Carrito	<b>Given</b> la selección de un producto y cantidad <b>And</b> la selección de adicionales <b>When</b> el cliente haya realizado login <b>Then</b> Agregar el producto al carrito <b>And</b> guardar la información de manera local.
AC20 Crear pedido	<b>Given</b> un carrito con varios pedidos <b>When</b> el cliente haya realizado login <b>Then</b> Crear pedido al sistema con estado recibido <b>And</b> actualizar fecha de creación.

- **Feature 07(portal Operador):** Manejo de pedidos

AC21 Login	<b>Given</b> usuario y contraseña <b>Then</b> Acceder al portal del operador.
AC22 Tabla pedidos	<b>When</b> el operador haya realizado login. <b>Then</b> Mostrar todos los pedidos activos.
AC23 Cocinar pedido	<b>Given</b> la información de un pedido <b>When</b> estado de pedido es recibido <b>Then</b> Modificar el estado del pedido a “cocinando”
AC24	<b>Given</b> la información de un pedido

Enviar pedido	<b>When</b> estado de pedido es “cocinando” <b>Then</b> modificar el estado a “enviado” <b>And</b> seleccionar un domiciliario disponible <b>And</b> modificar estado de domiciliario a no disponible
AC25 Completar pedido	<b>Given</b> la información de un pedido <b>When</b> estado de pedido es “enviado” <b>Then</b> modificar el estado a “entregado” <b>And</b> modificar estado de domiciliario a disponible

- **Feature 08(Portal administrador):** CRUD operadores

AC26 CREATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un operador <b>Then</b> registrarlo en la base de datos
AC27 READ	<b>When</b> se haya realizado login como administrador <b>Then</b> mostrar la información de los diferentes operadores
AC28 DELETE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un operador <b>Then</b> eliminarlo de la base de datos

- **Feature 09(Portal administrador):** CRUD domiciliarios

AC29 READ	<b>When</b> se haya realizado login como administrador <b>Then</b> mostrar la información de los diferentes domiciliarios
AC30 CREATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un domiciliario <b>Then</b> registrarlo en la base de datos
AC31 UPDATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un domiciliario <b>Then</b> actualizar sus datos en el sistema
AC32 DELETE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un domiciliario <b>Then</b> desactivarlo del sistema

- **Feature 10(Portal administrador):** CRUD Adicionales

AC33 READ	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un nuevo adicional <b>Then</b> generar un nuevo adicional
AC34 READ	<b>When</b> se haya realizado login como administrador <b>Then</b> mostrar la información de todos los adicionales
AC35 READ	<b>When</b> se haya realizado login como administrador <b>Given</b> la selección de un adicional <b>Then</b> modificar sus datos
AC36 READ	<b>When</b> se haya realizado login como administrador <b>Given</b> la selección de un adicional

	<b>Then</b> desactivarlo del sistema.
--	---------------------------------------

- **Feature 09(Portal administrador):** CRUD productos

AC37 READ	<b>When</b> se haya realizado login como administrador <b>Then</b> mostrar la información de los diferentes productos disponibles
AC38 CREATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un producto <b>And</b> la selección de sus posibles adicionales <b>Then</b> registrarlo en la base de datos
AC39 UPDATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un producto <b>Then</b> actualizar sus datos en el sistema
AC40 DELETE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un producto <b>Then</b> desactivarlo del sistema. Es decir, no se elimina del sistema, pero no está disponible para la compra.

- **Feature 10(Portal administrador):** Ver historial de pedidos

AC41 READ	<b>Given</b> la información del sistema <b>Then</b> mostrar el historial de pedidos entregados <b>And</b> filtrar por rango de fechas <b>And</b> filtrar por producto
--------------	--

- **Feature 11:** Roles

AC42	Given la información de roles y funcionalidades Then implementar el control de acceso para las funcionalidades
------	---

## Entregas

- Primera Entrega
  - Features: 1 al 7
- Segunda Entrega
  - Features: 8 a 10
  - SPA en Angular
- Tercera Entrega
  - Features: 11
  - Pruebas automatizadas a nivel del cliente web de al menos 2 casos de uso.
  - 25 pruebas automatizadas a nivel del backend.
  - Las pruebas automáticas deben cargar datos de prueba en la base de datos, los cuales deben removerse después de cada prueba

## **Calificación**

La nota de los grupos se dividirá en 2 partes (2.5 unidades C/U). La primera será la calificación del cliente, que se calculará a partir de la entrega de cada Sprint. La segunda será la calificación del product manager el cuál revisa que todos los requerimientos a nivel funcional de la página se cumplan.

### **Calificación del cliente:**

Cada semana se realizarán entre una y dos entregas en formato de sprint donde los grupos de trabajo deben mostrar adelantos FUNCIONALES de su trabajo. El cliente se fija de manera general en temas funcionales, UX, UI y CX, para dar su calificación por sprint.

Al final de semestre la calificación del cliente aporta para distinguir cuál ha sido el mejor proyecto durante todo el curso.

### **Calificación del Product owner:**

Este revisará al final de cada corte a nivel funcional y de código cómo va el producto general a entregar. Se usará un enfoque de calificación “substractiva” donde cada uno de los grupos empieza con las 2.5 unidades y por cada error cometido se van substrayendo decimas de la nota final.

Algunas de las penalizaciones son:

- Código no comentado
- Malos nombres de variables
- No hay validación de los formularios
- Problemas de rendimiento de la página
- No se cumplen con los requisitos funcionales propuestos
- Código duplicado (puede ser por mal uso de funciones o de componentes)
- No usar los patrones de diseño mostrados en clase
- Mal diseño (de base de datos, diagrama de clases, navegación)
- Usar paquetes para organizar el proyecto de manera coherente
- Usar recursos (imágenes y videos) de alta calidad donde no es necesario