



## PROYECTO

# DISEÑO Y DESARROLLO WEB CON DJANGO 2.0

PROGRAMA DE TECNOLOGÍA EN CÓMPUTO  
GENERACIÓN 37

## Sistema para el tema asignado

### Problema a resolver:

Una o varias de las sucursales del tema asignado han abierto recientemente y desean contar con un sistema de administración tanto del lado del cliente como del servidor para tener el control, brindar el servicio dicho a sus clientes así como la administración de sus productos, servicios, consultas y/o compras con los que cuentan.

### En cuanto a backend(Django 2.0):

#### Para el cliente:

- El cliente podrá consultar los productos/servicios disponibles, así como su precio y cantidad sin necesidad de iniciar sesión.
- El cliente deberá estar registrado en el sistema para realizar compras o agendar citas en línea según sea el caso.
- Se podrán registrar n cantidad de clientes.
- El cliente podrá realizar compras en línea, utilizando un carrito de compra, o agendar citas.
- El sistema tendrá un manejador de sesiones.
- El cliente podrá recuperar su contraseña por un correo enviado desde el servidor.

- El cliente podrá agregar/agendar y eliminar/cancelar productos/citas del carrito de compra/agendador.

### Para el administrador:

- El administrador podrá agregar, eliminar y modificar datos de los productos (nombre, precio, proveedor).
- El administrador podrá determinar si habrá descuento para cierto tipo de productos o en su caso añadir promociones.
- El administrador podrá modificar las cantidades de los productos existentes.
- Todos los datos se almacenarán en una base de datos(SQLite 3).
- El proyecto podrá verse desde cualquier lugar del mundo que tenga acceso a Internet, es decir que esté alojado en un servidor.

### Extras:

- El gerente podrá consultar el historial de ventas/citas realizadas por la tienda.
- El gerente podrá consultar el historial de ventas/citas por cliente.
- Se tendrá un API REST que se pueda consumir.

### En cuanto al frontend:

- Se tendrá que utilizar uno de los dos frameworks vistos en clase(Bootstrap 4 o Materialize).
- Se debe notar el uso del GRID.
- Al menos 15 elementos distintos del framework elegido deberán estar presentes en los templates.
- A imaginación de cada equipo se aplicará un script(Obvio de Javascript) con el que el usuario pueda interactuar.
- Otro script pero utilizando jQuery.
- Se utilizarán al menos una de las APIS vistas en clase.
- Se debe notar el cambio de tipo de fuente en al menos dos partes del sitio.

## Evaluación backend:

- Manual de uso:
  - Descripción
  - Pruebas de estrés
  - Ejecución
  - Flujo del sistema (Incluir capturas de pantalla).
  - Limitaciones (si hay alguna)
- Documentación con pydoc
- Diagrama de clases UML si es que se utilizan clases.
- Versionado (Repositorio de GitHub). Al menos 10 commits por integrante.
- Presentación del proyecto.

## Evaluación de frontend:

Documentación orientada a ingeniería de software:

- Documentación para convencer al cliente de que son la mejor opción (logo, misión, visión, valores etc.)
- Objetivo.
- Integrantes del proyecto (presentarse como especialistas en... bases de datos, desarrollo web etc.).
- Organigrama (Se debe marcar quién hizo cada parte del proyecto).
- Toma y análisis de requerimientos.
- Propuesta para solucionar el problema.
- Diagrama de actividades (Gantt). *"Si no metran los avances no cuentan"*.
- Integrar el manual de uso al final de esta documentación.
- La documentación será formal y se deberá entregar al inicio de sus presentaciones.
- Ninguna presentación deberá ser igual a la de otro equipo, de lo contrario no se calificará.

Valor de cada punto:

● Uso del framework Django	60%
● Script Javascript	5%
● Script jQuery	5%
● Uso de API	5%
● Cambio de fuente	2%
● 10 commits por integrante en GitHub	5%
● 100% Mobile-First	3%
● Documentación	15%
● Presentación	5%
Total	100%

**Fecha de presentación:**

➤ Por confirmar.

**Lugar y hora:**

➤ Por confirmar.

**Es requisito indispensable tener su sistema en Internet, de lo contrario no se revisará.**



**PROGRAMA DE TECNOLOGÍA EN CÓMPUTO**

