



Desarrollo Basado en Plataformas

CS2B01 - 2022-1

PROYECTO 1 - PARTE WEB

Submitted By : Marvin Abisrror

mabisrror@utec.edu.pe

PROYECTO 1 - APLICACIÓN WEB

CS2B01

Objetivo General

En el proyecto de este curso deberás de crear una aplicación del tema que más te guste, para lo cual todos en tu grupo deberán estar de acuerdo. Para este proyecto deberán buscar un tema que les guste y empezar a crearlo!

Este proyecto se tratará de una aplicación que deberá tener dos partes: una parte web (utilizando HTML, CSS y Javascript) y la otra será un servidor en Flask, conectado con una Base de Datos PostgreSQL con la ayuda de psycopg2 y SQLAlchemy, como se ha visto durante las clases.

La **Figura 1** muestra el diagrama de arquitectura de esta aplicación.

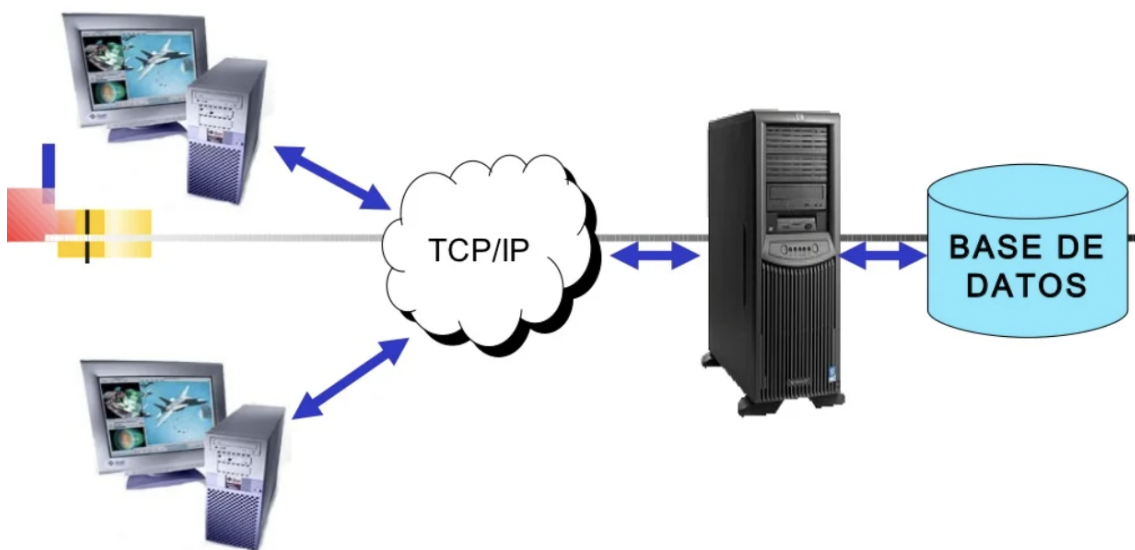


Figure 1: Diagrama de Arquitectura de Software del Proyecto

La aplicación web deberá cumplir con las restricciones que las plataformas web ponen a los desarrolladores tales como:

Cliente-Servidor

El proyecto utilizará el estilo arquitectónico **MVC** (*Modelo-vista-controlador*). El

usuario podrá comunicarse con el sistema utilizando una página web. Se podrá enviar y recibir información del servidor y manejará/visualizará dependiendo de las funcionalidades de cada proyecto. Al separar las responsabilidades de la interfaz de usuario de las de almacenamiento de datos, mejoramos la portabilidad del primero en múltiples plataformas y mejoramos la escalabilidad al simplificar los componentes del servidor.

Base de Datos

El proyecto deberá tener una conexión Servidor - Base de Datos estructurada y establecida. Todas las transacciones que se requieran hacer para tratar la data, tales como consulta, inserción, eliminación y modificación, deberán realizarse a través de instrucciones predefinidas del servidor en Flask. De esta manera, la capa de persistencia de datos se mantiene y aseguramos su integridad.

Indicaciones Generales

Entregables

- Recuerda que el trabajo es grupal (máximo 4 personas), y que el hecho de compartir código entre grupos no está permitido y será sancionado.
- La fecha de entrega y demostración será en la semana 8 en el horario de laboratorio de cada sección, del **09** al **13** de Mayo.
- Los features mínimos requeridos del proyecto incluyen:
 - Un módulo de registro y login para usuarios del sistema.
 - Poder hacer **CRUD** (*Create, Remove, Update, Delete*) a los recursos del sistema.
 - Tener relaciones de modelos. *One to many, many to many, etc.*
- Se desea del proyecto:
 - Incluir ***Unit testing*** para todos los endpoints del sistema.
 - Utilizar ***Flask DB migration***.
 - Mostrar los patrones de ***error handling*** utilizados.

- Seguir la guía de estilización **PEP 8** para las mejores prácticas al momento de desarrollar código (*Crear funciones, métodos, clases, variables, etc*). <https://www.python.org/dev/peps/pep-0008/>
 - Incluir el archivo **.gitignore** en el entregable, para evitar subir archivos basura a github.
 - Utilizar **github** y utilizar comentarios entendibles en cada **commit** realizado por cada integrante del grupo. Se realizará un análisis de los **commits** de los proyectos.
-
- Un **README.md** en el repositorio que incluya:
 - **Nombre del proyecto.**
 - **Integrantes.**
 - **Descripción del proyecto.**
 - **Objetivos principales / Misión / Visión.**
 - **Información acerca de las librerías/frameworks/plugins utilizadas en Front-end, Back-end y Base de datos.**
 - **El nombre del script a ejecutar para iniciar la base de datos con datos.**
 - **Información acerca de los API. Requests y Responses de cada endpoint utilizado en el sistema.**
 - **Hosts.**
 - **Forma de autenticación.**
 - **Manejo de [errores HTTP](#):**
 - 500 : Errores en el Servidor**
 - 400 : Errores en el Cliente**
 - 300 : Redirección**
 - 200 : Exitoso**
 - 100 : Informacional**
 - **Cómo ejecutar el sistema (Deployment scripts).**

- Revisa bien lo que entregas, después de la fecha de entrega no podrás hacer modificaciones.

Presentación

La presentación oral será realizada por todos los miembros del grupo, que contará con 20 minutos como máximo y se realizará dentro de las horas de laboratorio de la semana 8. Tendrá la siguiente estructura:

1. Presentación de integrantes.
2. Presentación del proyecto: explicación del problema y solución propuesta.
3. Demostración en vivo:
 - (a) Mostrar un **Happy path** o Flujo exitoso de la funcionalidad principal del sistema. Desde registro del usuario hasta la modificación/visualización de algún recurso.
 - (b) Mostrar un **Unhappy path**. Flujo con errores cometidos por el usuario durante su uso (**Nombre de usuario no encontrado, valor ingresado incorrecto, etc.**). Se debe manejar el error de forma amigable al usuario (**Mostrando un mensaje de error, validando la forma antes de enviar, etc.**).
 - (c) Mostrar los registros en la base de datos después de realizadas las operaciones de las demostraciones.
4. Ronda de preguntas sobre el **código, sistema, participación en el proyecto, etc.** Los demás compañeros tienen que participar.