

Trabajo Práctico

Introducción al Desarrollo de Software

El presente trabajo práctico de la materia Introducción al Desarrollo de Software integra todos los temas aprendidos. Los equipos formados entre 6 y 8 alumnos de la materia, deberán aplicar buenas prácticas de programación, y utilizar metodologías ágiles para su resolución.

En esta oportunidad no tendremos un tema específico a resolver, sino que se presentarán varias ideas que se podrán desarrollar respetando los requerimientos abajo descritos. También aquellos equipos que quieran desarrollar una idea en particular podrán hacerlo previa aprobación de la cátedra, para esto tendrán que presentar la idea **dentro de los primeros 7 días a partir de la entrega de este enunciado**.

Ideas de posibles desarrollos:

a- Ludoteca Central.

La empresa ficticia "**Ludoteca Central**" desea lanzar su tienda online para vender juegos de mesa. Necesitan una plataforma web que permita a los clientes navegar el catálogo de productos, ver detalles de cada juego, registrarse, iniciar sesión y realizar compras. Los administradores podrán cargar y modificar productos, ver pedidos y gestionar el stock.

Los usuarios podrán agregar juegos al carrito de compras y efectuar la compra.

b- VideoZone.

La empresa ficticia "**VideoZone**" quiere desarrollar una plataforma donde los usuarios puedan compartir y descubrir videos interesantes. A diferencia de otras plataformas, los videos no se suben directamente, sino que los usuarios comparten el **link** (por ejemplo, de YouTube o Vimeo), junto con una descripción. Otros usuarios pueden **ver el contenido, dar me gusta, dejar comentarios, puntuar, buscar un video por nombre, categoría, verlos por puntaje**, etc.

c- TPBuddy.

En muchas materias de la facultad, los estudiantes deben formar grupos para realizar trabajos prácticos. Sin embargo, es común que no conozcan a sus compañeros o que les cueste organizarse. La facultad desea desarrollar una plataforma llamada "**TPBuddy**", que facilite la **búsqueda de compañeros**, la **formación de grupos** y la **gestión de trabajos prácticos** por materia.

La misma tiene que permitir crear un perfil por alumno con su información, sus horarios disponibles, sus materias actuales, si tiene o no grupo, cuántas personas le faltan para completar el grupo y cualquier otra información relevante para formar un buen grupo de TP.

Los alumnos tienen que poder "pedir unirse" a un grupo y alguien del mismo tiene que "aceptar" o "rechazar" la petición.

Cuando un grupo llega al límite de miembros, no debe permitir peticiones extras.

d- Proyecto a elección.

El mismo deberá ser web y mobile, ser validado por la cátedra dentro de los 5 días de entregado este enunciado.

Para cualquiera de las ideas se deberá tener los siguientes requerimientos.

Requisitos obligatorios:

- Uso de **Github**. Deben compartirlo con el corrector de la cátedra y todos deben participar aportando commits y desarrollo útil.
- Utilización de un tablero Kanban con tareas a desarrollar **Backlog de producto**, utilizando [Trello](#).
El tablero deberá tener las columnas necesarias que marquen los estados de cada una de las tareas. En cada iteración los entregables deberán estar pasados a “entregado” para que sea aceptado como tal. Por otro lado las tareas deberán tener un ID identificador (ver que hay plugins para esto) y fecha de inicio de la misma.
- Los hitos a presentar deberán entregar valor al cliente.
- Los integrantes del equipo deberán realizar commits al repositorio, según vaya avanzando el proyecto. Estos commits deberán estar asociados a una tarea de Trello y **en el mensaje del commit se deberá informar el ID correspondiente**.
- Utilizar una base de datos **MySQL con al menos 2 tablas y un script de creación/carga**.
- Tener una aplicación **backend** utilizando **Flask** corriendo que se conecte a la base de datos y cualquier otra dependencia. Los endpoints deberán ser realizados utilizando el “verbo” correcto y se utilizará el formato **JSON** para el envío o recibir información.
- Tener una aplicación para el **frontend** utilizando **Flask** que se conecte al backend respetando **RESTful**.
- **Presentación de informe** detallando las principales decisiones de diseño, dificultades encontradas y conclusiones.
- En caso de utilizar dependencias se deberá tener un archivo de init.sh que tenga la instalación de las mismas.

Requisitos deseables

- Uso de **Docker**
- **Deploy de la app y de la api** conectando la base de datos en pythonanywhere (se deberá crear una cuenta por proyecto)
- Documentación sobre cada endpoint de la API y ejemplos de uso (pueden utilizar swagger para esto)
- Presentación/Demo sobre el uso de la página
- Integrar una app mobile

Hitos del proyecto (entregables)

Las entregas deberán acordarse con el tutor asignado.

- Jueves 22/05. Se deberá entregar Backlog de producto, mockup del sitio web (pueden usar alguna app online o directamente en un dibujo)
- Jueves 03/06. Creación del repositorio, vistas del proyecto (templates), listado de servicios a utilizar.
- Jueves 10/06. Avances de la API y comunicación con el front
- Jueves 19/06. **1er entrega del TP Integrador (a confirmar)**
- Martes 26/06. **2da entrega del TP integrador (a confirmar)**

Defensa Oral

La entrega del Trabajo Práctico estará acompañada de una defensa oral tanto grupal como individual.

Esta se considera la instancia más importante de evaluación dentro de la materia.

Se espera que tanto el alumno como el grupo en su totalidad pueda responder de forma completa las preguntas que llegara a plantear el docente en lo que respecta a la funcionalidad del trabajo, las herramientas provistas por el lenguaje de programación y el diseño del sistema en general.

Documentación Requerida

Para la presentación del Trabajo Práctico, se pide al equipo la presentación de un informe que contenga.

- Carátula
- Integrantes
- Resumen
- Introducción
- Solución propuesta
- Tecnología utilizada
- Conclusión final

La información volcada en el documento será utilizada por el equipo docente previo a corregir el trabajo por lo cual el alumno puede expresar allí lo que crea conveniente para ayudar en el proceso de corrección.

Condiciones de Aprobación

NO se debe...

- usar variables globales
- usar ciclos infinitos
- escribir código repetido
- entregar un programa que no funcione
- entregar un programa con errores graves
- incumplir con la fecha de entrega

SÍ se debe...

- Evitar que el programa rompa frente a la interacción con el usuario
- modularizar el código
- usar buenas prácticas de programación
- utilizar un repositorio en github
- entregar un programa que cumpla con la funcionalidad completa especificada

Fecha de Entrega

El Trabajo Práctico cuenta con 2 fechas de entrega: 19/06/2025 y el 26/06/2025. Ambas tienen hora límite 18:00 hs. **Mientras que la primera fecha tiene posibilidad de recuperación, la segunda no.**

En esta línea, se espera que el grupo entregue el trabajo completo para la primera fecha, y -en caso de recibir correcciones por parte del equipo docente- tendrán posibilidad de realizar una segunda entrega en la segunda fecha. Esta primera instancia de entrega no restará puntos en caso de re-entregar en la segunda fecha. La segunda fecha definirá la nota del trabajo según corresponda.

Es importante saber que la primera entrega debe estar completa y sin errores graves.

Quienes decidan entregar únicamente en la segunda fecha (sin pasar por la primera) no tendrán posibilidad de correcciones intermedias y tendrán una nota final del trabajo según corresponda. Tanto la primera entrega como la segunda deben contener la totalidad de la funcionalidad solicitada, no se aceptarán entregas parciales o trabajos incompletos en ninguna de las fechas.

Importante

La aprobación del Trabajo Práctico es condición necesaria (junto con la aprobación del parcial) para regularizar la materia.