



Proyecto Semestral - Sección 1

1. Objetivos

- **Demostrar la comprensión** de conceptos de tecnologías web, como HTML, CSS, JavaScript y React, para desarrollar una aplicación web funcional y atractiva.
- **Trabajar de manera conjunta** en la planificación, diseño y codificación de una aplicación web, dividiendo tareas y responsabilidades de manera equitativa.

2. Introducción

El proyecto a desarrollar a lo largo del curso tiene como objetivo que los estudiantes, en un contexto de trabajo en equipo, enfrenten un desafío de diseño, programación, implementación y despliegue de una aplicación web, que les permita poner en práctica los contenidos del curso vistos durante el semestre.

Las entregas del proyecto están vinculadas entre sí, y el resultado final del proyecto será una aplicación totalmente funcional con un lado cliente y otro servidor.

Para lograr lo anterior, deberán cumplir con objetivos intermedios, que corresponderán a las “entregas parciales de proyecto”, de manera que el desarrollo sea incremental y guiado tanto por los contenidos como por el equipo de ayudantes.

Salvo excepciones, condiciones y restricciones, que se señalarán oportunamente en cada enunciado de entrega, cada equipo tendrá total libertad en cuanto a la UI y la implementación del proyecto.

3. Metodología de trabajo

Para el desarrollo del proyecto, los estudiantes deberán formar un grupo de 3 personas que se mantendrá para todas las entregas. Para inscribir los grupos se creará un formulario en *Google Forms* donde podrán registrarlos.

Quienes no tengan grupo, al final del periodo de inscripción, serán asignados aleatoriamente a quienes estén solos.

4. Reglas generales sobre las entregas

El proyecto a desarrollar deberá cumplir al menos con las siguientes funcionalidades:

- Deberá tener registro y autenticación de usuarios (con *login* y *password*).
- Deberá usar una base de datos para almacenar toda información relevante de manera persistente.
- El lado del cliente (*front-end*) deberá tener una interfaz gráfica atractiva, completa e intuitiva que permita realizar múltiples acciones (tales como seleccionar, hacer *drag-and-drop*, u otras que el diseño permita. No tiene que soportar todas las acciones, pero sí se espera que sea una interfaz no-trivial).
- Se deben manejar diversos perfiles de usuarios y acciones para que estos puedan realizar acciones acorde a su perfil. Por ejemplo, perfiles de jugador, administrador, “superusuario”, “moderador”, etc.

- Debe tener un proceso (estados) que se maneje por base de datos. Por ejemplo, estado de un juego, estado de solicitud, estado de horario, etc.
- Debe tener páginas dinámicas, adicionales a las estáticas, que varíen su contenido dependiendo de las acciones que el usuario realice.
- Debe tener un panel que permita ver el resumen de acciones hechas por un usuario (un análogo sería una aplicación de banco que muestre las transacciones realizadas en un período fijo de tiempo).
- Debe manejar la información de manera segura (por medio de encriptación) cuando corresponda

La propuesta de aplicación a desarrollar por cada grupo podrá ser elegida entre 3 opciones:

- **Juego de tablero:** en esta primera opción, podrán diseñar y desarrollar su propio juego de tablero en línea, inventando algo completamente nuevo o reimaginando un juego clásico existente. Se espera:
 - Diseño de un tablero interactivo y visualmente atractivo.
 - Implementación de reglas de juego lógicas y emocionantes.
 - Integración de elementos visuales para mejorar la experiencia.
 - Posibilidad de jugar con múltiples jugadores en línea.
- **Planificador:** en esta segunda opción, deberán crear un planificador en línea que tenga como función ayudar a sus usuarios a gestionar tareas, eventos, clases, metas etc. La idea es que sea una plataforma intuitiva que permita a los usuarios planificar lo que sea que quieran planificar (dependiendo del tema en el que esté basada su aplicación) de manera efectiva. Se espera:
 - Creación de un diseño limpio y funcional para la gestión de tareas y eventos.
 - Incorporación de funciones de recordatorio y notificación.
 - Posibilidad de establecer metas y realizar un seguimiento del progreso.
 - Personalización de la interfaz de usuario tal que esta pueda adaptarse a las preferencias de estos.
- **Red social:** esta última opción se basa en diseñar y desarrollar una red social con un enfoque temático único. Ya sea que elijas un tema artístico, deportivo, educativo o de otro tipo. Pueden basarse en redes sociales existentes pero la idea es que usen su creatividad para agregarle algo nuevo. Se espera:
 - Diseño de perfiles de usuario personalizables y atractivos.
 - Implementación de un sistema de publicación y comentarios.
 - Integración de características únicas relacionadas con el tema elegido.
 - Implementación de un chat donde los usuarios puedan comunicarse.

Adicionalmente, se podrán proponer proyectos fuera de las temáticas listadas siempre que sea conversado y justificado con el ayudante a cargo del grupo. Cualquier proyecto propuesto deberá incluir un conjunto de funcionalidades cuya implementación sea de una dificultad similar a la de los proyectos listados.

5. Indicaciones generales

- Para alojar sus aplicaciones en Internet, se recomendarán plataformas a su debido tiempo. Si desean utilizar otro proveedor al recomendado pueden hacerlo, pero deberán justificar y documentar su elección.

- El código deberá desarrollarse, alojarse y entregarse en GitHub, por lo que a cada grupo se le asignarán repositorios según sea necesario. Es importante que todos los integrantes de cada grupo utilicen el repositorio y GitHub de manera colaborativa. **Se revisará el uso de GitHub para evaluar que todo miembro haya trabajado.**
- El resultado final esperado del proyecto es que se logre realizar el despliegue de la aplicación web (en modo producción) en un servidor externo, de manera que esta quede disponible para ser utilizada por cualquier usuario de Internet.
- Para evaluar cada entrega, todos los miembros del equipo deberán asistir a una reunión de corrección con el ayudante asignado al grupo. **Es responsabilidad del grupo el contactar y agendar la reunión con su ayudante.**

6. Condiciones y restricciones

- La aplicación deberá desarrollarse usando, al menos, las siguientes tecnologías: Node.js, Koa, HTML, CSS, JavaScript y React. Opcionalmente, se permite el uso de jQuery si se estima pertinente.
- Para la base de datos se deberá usar PostgreSQL y para conexión a base de datos se podrá utilizar pg, knex o un ORM como Sequelize.
- Se puede hacer uso de EJS, SASS/SCSS, Less, Stylus u otros preprocesadores de CSS.
- Se puede hacer uso de lenguajes que compilen a JavaScript, tales como CoffeeScript o TypeScript.
- Además, se puede hacer uso de bibliotecas de componentes HTML/CSS/JS como Bootstrap o Zurb Foundation.
- Cualquier paquete, *framework* o librería adicional que se quiera utilizar deberá ser aprobada por el equipo docente.

7. Fechas de entrega

El proyecto se divide en 5 entregas:

Evaluación	Publicación	Entrega	Ponderación
Entrega 0	viernes 18 de agosto	viernes 25 de agosto	10 %
Entrega 1	viernes 1 de septiembre	viernes 15 de septiembre	20 %
Entrega 2	viernes 13 de octubre	viernes 27 de octubre	20 %
Entrega 3	viernes 3 de noviembre	viernes 17 de noviembre	25 %
Entrega 4	viernes 17 de noviembre	viernes 1 de diciembre	25 %

8. Evaluación y descuentos

Cada entrega de proyecto será evaluada con una nota de 1 a 7, donde la distribución de puntaje se indicará en el enunciado de cada entrega. Sin embargo, cada entrega (a excepción de la E_0) tendrá **ítems esenciales** que todos los equipos deberán alcanzar. En caso de no estar realizados, el grupo podrá optar a una nota máxima 4,0.

Tal como se menciona en el programa del curso, la última entrega es de **carácter reprobatorio**. Es decir, el no obtener una calificación mayor o igual a 3.95 en dicha entrega significará que la nota de proyecto se calculará como el mínimo entre la nota obtenida y 3.90.

El espacio de evaluación con el ayudante guía debe ser gestionado por el grupo. En esta reunión se evaluará que todos los integrantes manejen lo implementado en la entrega correspondiente. Además, se les dará el *feedback* e indicaciones para la siguiente entrega.

En caso de no ponerse en contacto con el ayudante o no asistir (sin una justificación adecuada), el grupo o integrante tendrá un descuento del 50 % sobre el puntaje obtenido en la entrega.

9. Evaluación de pares

Al final de cada entrega, se abrirá un formulario de evaluación de pares, donde los integrantes del grupo tendrán que evaluar a sus compañeros del 1 al 5.

Cada estudiante tendrá **8 puntos a distribuir entre sus dos compañeros**. Por defecto, se entregan 4 puntos a cada integrante (no es necesario completar el formulario si ese es el caso). En caso de querer asignarle 5 puntos a un compañero, es porque hizo un trabajo sobresaliente y, por ende, implica que tuvo que compensar por alguien que no trabajó del todo bien y merece menos puntaje.

La nota luego se calcula como:

$$E = 1 + ((E_{\text{grup}} - 1) \cdot F_g)$$

Donde F_g es un factor calculado a partir de las evaluaciones de los compañeros. Puede moverse entre 0 y 1,25.

10. Política para entregas atrasadas

Durante el proyecto completo, cada grupo puede utilizar (3) cupones de atraso. En caso de no usarse en esta entrega, pueden ser utilizados durante el resto del proyecto. Los días de atraso incluyen fines de semana (utilizar un cupón de atraso un viernes, mueve la entrega al sábado, no al siguiente día hábil).

Si un grupo no utiliza sus cupones durante todo el semestre, se le bonificará con 2 décimas por cada cupón no usado, sobre la nota final del proyecto. Esto solo si está aprobando el curso.

11. Recomendaciones

- Diseñar bien la lógica de cada parte de la aplicación, el entorno, los estados, la información requerida y su modelo de almacenamiento.
- Planificar el trabajo para que permita y facilite la colaboración entre integrantes del equipo.
- Preguntar y consultar, usando los canales dispuestos por el equipo docente, colaborando entre pares (sin copiar) y apoyándose en los ayudantes.
- Trabajar con tiempo, planificando cómo abordar una entrega en cuanto se publica el enunciado.
- Comenzar con una interfaz de usuario (UI) simple y minimalista, e ir evolucionando el diseño a medida que se avanza en la implementación.
- Es importante conocer los detalles de diseño y decisiones tomadas en la construcción de la aplicación. El conocimiento en detalle de lo que se está realizando es parte de los aspectos que serán evaluados.