



Aspectos y consideraciones generales sobre el proyecto

IIC2513 Tecnologías y aplicaciones WEB

Primer semestre 2024

Introducción

El proyecto, a desarrollar a lo largo del curso, tiene como objetivo que los estudiantes enfrenten un desafío de diseño, programación, implementación y despliegue en ambiente web, que les permita poner en práctica los conocimientos adquiridos durante este semestre.

Las entregas del proyecto están vinculadas entre sí y la última de las entregas tendrá como resultado una aplicación totalmente funcional con un lado cliente y otro servidor (¡en la nube!).

Para lograr lo anterior, usted deberá cumplir con objetivos intermedios, mejor conocidos como “entregas parciales de proyecto”, de manera de llevar un diseño inicial a un “producto” totalmente funcional. *Lo anterior revela la importancia de no atrasarse ni saltarse una entrega y procurar mantenerse al día.*

Los enunciados que guiarán cada entrega parcial darán la línea general de la funcionalidad que deberá implementar, pero **sin entrar en mayores detalles ni puntos específicos** de tal manera que usted, junto a su grupo, demuestren su capacidad de trabajo en equipo y resolución de problemas.

Salvo excepciones, condiciones y restricciones, que se señalará oportunamente en cada enunciado de entrega, usted tendrá total libertad en cuanto a la interfaz y la implementación de su trabajo.

Para el desarrollo del proyecto, deberá formar un grupo (que se mantendrá para todas las entregas) de sólo 3 personas. 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 no tengan grupo (también deberán contestar un formulario que se habilitará con ese objetivo).

Marco general de las entregas

El resumen del proyecto en una línea:

Deberán construir una aplicación Web.

La aplicación web a construir será un juego por turnos con al menos tres participantes.



Reglas y características generales del proyecto de tipo juego

El juego a realizar será propuesto por cada grupo. Las propuestas serán evaluadas por el equipo docente y se acordará entre ambas partes (grupo y ayudante) sus características finales, esto será parte de la retroalimentación de la entrega 1.

El juego, en la **entrega final** al menos:

1. Deberá tener registro y autenticación de usuarios (login y password)
2. Deberá usar una base de datos para almacenar información relevante
3. Deberá ser un juego basado en turnos, es decir, una persona realiza todas sus jugadas y luego esperará a que el otro jugador la realice (por ejemplo como en el juego Ataque/Risk). Es posible también que todos los jugadores hagan sus jugadas y las entreguen al servidor el cual, una vez que reciba TODAS las jugadas (o se cumpla un tiempo de envío), para que el servidor resuelva combinando simultáneamente todas las jugadas.
4. El lado cliente debe tener una interfaz gráfica atractiva, completa y que permite realizar múltiples acciones (como seleccionar, drag and drop, u otras que el diseño permita. No tiene que tener todas las acciones, pero si se espera que sea una interfaz no-trivial)
5. Se deben manejar diversos perfiles de usuarios y acciones para que estos puedan realizar acciones acorde a su perfil. Por ejemplo, perfil jugador, perfil administrador, perfil superusuario, perfil "moderador" de juego, etc.
6. Debe tener un proceso (estados) que se manejen por base de datos. Por ejemplo, estado de los turnos, turnos pasados por jugador, indicar qué jugador posee el turno actual, etc.
7. Debe tener páginas dinámicas, adicionales a las estáticas, que varíe su contenido dependiendo de las acciones que el usuario realice
8. Debe tener un panel que permite ver resumen de acciones hechas por usuario (un análogo sería una aplicación de banco que muestre transacciones realizadas en un periodo fijo de tiempo)
9. Debe manejar seguridad (encriptación) de la información, cuando corresponda
10. Debe tener un panel de administración que permita auditar las acciones realizadas por los distintos usuarios y otras acciones acordes a este perfil, como por ejemplo dar de baja a usuarios, suprimir juegos u otras

Indicaciones generales

1. 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. Noten que las cápsulas, guías y apoyo de los ayudantes será para las plataformas que se señalan.

2. El código deberá entregarse en GitHub. Les indicaremos, una vez oficializados los grupos, cuáles son los repositorios asignados por grupo. Es importante que todos los integrantes utilicen el GitHub de manera colaborativa. **SE REVISARÁ EL USO DE GITHUB PARA EVALUAR SI ALGUIEN TRABAJÓ O NO.**
 3. El resultado final esperado es que ustedes realicen el despliegue real de su aplicación web (juego o cadena de producción) en modo producción en la nube y que ésta quede disponible para ser utilizada por cualquier usuario de Internet.
 4. Para evaluar cada entrega, **todos** los miembros del equipo tendrán una reunión de corrección con el ayudante que se les haya asignado. Hay que aclarar que es **RESPONSABILIDAD DEL GRUPO** el contactar y agendar la reunión con el ayudante.
 5. El avance logrado en cada entrega se evaluará en base a lo mostrado, cuando corresponda, en el repositorio Github en la fecha límite de la entrega. Para entregas más avanzadas se mirará lo entregado en las plataformas respectivas y se harán pruebas funcionales, es decir evaluación si la aplicación hace o no lo que debiese estar haciendo. Recordar que Github se utilizará como una medida de trabajo de cada integrante del grupo. Si el ayudante detecta que uno o más integrantes del grupo tienen baja interacción en su repositorio de Github, puede descontar a esos alumnos un porcentaje de la nota del grupo, todo esto se detalla más adelante.
 6. En el caso de que un grupo no entregue nada, la nota será de un 1.0.
-

Condiciones y restricciones

1. La aplicación debe desarrollarse usando **al menos**: Node.js, Koa, HTML, CSS, Javascript y React.
 2. Para la construcción de la aplicación pueden usar JQuery si lo estiman pertinente.
 3. Para base de datos **deberán** usar Postgres y para conexión a la base de datos pueden utilizar pg, knex o un ORM como sequelize
 4. Pueden usar también EJS, SACC/SCSS, less, stylus u otros preprocesadores de CSS
-



5. **No está** permitido el uso de lenguajes que compilen a JavaScript, como CoffeeScript o TypeScript. Tampoco podrán usar bibliotecas de componentes HTML/CSS/JS como Bootstrap o Zurb Foundation.
6. **Cualquier package, framework o biblioteca adicional que quieran usar debe ser aprobado por el equipo docente.**

Entregables

- **Entrega 1 (E1):** 8 de abril, pondera un 15% de la nota del proyecto
- **Entrega 2 (E2):** 13 de mayo, pondera un 20% de la nota del proyecto
- **Entrega 3 (E3):** 3 de junio, pondera un 30% de la nota del proyecto
- **Entrega 4 (E4):** 21 de junio, pondera un 35% de la nota del proyecto

Evaluación y Descuentos

Cada entrega de proyecto será evaluada con una nota de 1 a 7 siguiendo una escala discreta que se detalla a continuación:

Calificación	Nota asociada	Criterios por banda
A	7,0	La entrega excede con creces lo solicitado
B+	6,0	La entrega cumple algo más allá de lo solicitado
B	5,5	La entrega cumple con lo mínimo solicitado en el enunciado
C	4,0	La entrega cumple parcialmente con lo mínimo solicitado
D	2,5	La entrega está muy por debajo del mínimo solicitado
E	1,0	No se realizó la entrega en los plazos o en la forma

Cada banda de calificación tiene un criterio asociado, pero para alcanzarlo significa que se cumple con los criterios exigidos para el criterio anterior. Así, para tener la opción a la banda de la A, tendrá que cumplir con todos los criterios dados por la banda B+ y recursivamente hacia abajo.

Las calificaciones del trabajo práctico y los cumplimientos asociados a cada calificación serán **guiados activamente por el cuerpo docente, principalmente a través de los/las ayudantes**, con quienes los alumnos podrán tener una guía de lo que significa el criterio para cada banda y podrán acordar las actividades necesarias para cumplir con cierta banda de calificación.



Si alguna persona se sintiera perjudicada por el actuar de algún otro integrante del grupo, deberá comunicarse con el ayudante asignado para el caso o la ayudante de bienestar. El ayudante analizará el caso y podrá haber un descuento acorde a la falta detectada.

La calificación y el feedback para cada entrega se dará a conocer por el ayudante guía en una reunión que debe ser agendada por el grupo. **Para cada miembro, la asistencia es de carácter obligatorio.**

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

Adicionalmente, el equipo docente verificará la utilización de GitHub y la actividad de cada integrante del grupo en el repositorio. Si se detecta una diferencia significativa entre los integrantes del grupo tanto en el espacio de evaluación de la reunión como en el repositorio, el integrante que haya trabajado notoriamente menos, será notificado y **será penalizado con un 25% en la nota que haya obtenido el grupo**. Si en la siguiente entrega se repite la situación, el equipo docente se pondrá en contacto con el alumno para discutir la penalización asociada a esta entrega.

Los descuentos son acumulativos. Esto es, si una persona no asiste a la reunión con el ayudante y además presenta una participación significativamente menor, primero se le baja la nota por su contribución y luego se aplica el 50% asociado a la inasistencia.

Atraso

Para cada entrega se entregará un cupón de descuento. En caso de no usarlo, este se acumula para ser usado en las siguientes entregas. Los días de atraso incluyen los fines de semana. Es decir, si la entrega está fijada para el viernes, y el grupo quiere usar 2 cupones, deberá entregar el domingo.

Recomendaciones

1. Diseñen muy bien las reglas de negocio de su aplicación, el entorno, los estados, la información requerida y su modelo de almacenamiento.
2. Planifiquen el trabajo para que les permita la colaboración entre los integrantes del equipo.
3. Pregunten y consulten, ***usen los canales dispuestos por el cuerpo docente para hacer consultas***, colaboren entre ustedes (**NO COPIEN**) los ayudantes están para apoyarlos.
4. **Trabajen con tiempo**, no esperen a último momento para despejar dudas.
5. Comiencen con una interfaz de usuario (UI) simple y minimalista; no es necesario que su aplicación se vea "bonita" desde el comienzo
6. Se recomienda (no es obligatorio) realizar prototipos no funcionales (*mockups*) de su aplicación. La idea es que las utilicen en las entregas siguientes así que preocúpense de diseñarlos adecuadamente. Los detalles de cada entrega se darán en su enunciado específico, este es un marco general para las entregas.



Pontificia Universidad Católica de Chile
Escuela de ingeniería
Departamento de ciencias de la computación
Profesor: Hernán Cabrera

7. Es importante que conozcan los detalles de diseño y decisiones tomadas en la construcción de su aplicación. El conocimiento en detalle de lo que se está realizando es parte de los aspectos que serán evaluados.
8. Siempre podrán, justificadamente, cambiar alguna regla que ustedes hayan definido para su aplicación o bien mejorar algún aspecto de la misma, de la interfaz, modelo de datos, funcionalidad, etc.