



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

## **Tarea 1**

### **IIC2513 Tecnologías y aplicaciones WEB**

**Segundo semestre 2020**

**Fecha límite de entrega de tarea: 02 de septiembre, 20:00 hrs.**

Las tareas tienen como objetivo que enfrenten un desafío de diseño y programación que les permita practicar los conocimientos adquiridos durante este semestre.

Las tareas están vinculadas entre sí y la última de las tareas tendrá como resultado una aplicación totalmente funcional con un lado cliente y otro servidor. Es decir, ustedes deberán cumplir con objetivos intermedios de la tarea de manera de llevar un diseño e implementación 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 dan la línea general de la funcionalidad que deben implementar, pero sin entrar en mayores detalles ni puntos específicos de tal manera que ustedes demuestren su capacidad de trabajo en equipo y resolución de problemas.

Salvo excepciones, tendrán total libertad en cuanto a la interfaz y la implementación de su trabajo.

Recuerde inscribir a su grupo (2 a 3 personas) en el formulario Google Forms que se habilitará para ese efecto

**Fecha límite de entrega de tarea 1: 02 de septiembre, 20:00 hrs.**

### **Indicaciones**

---

El objetivo de esta primera entrega es el diseño de su aplicación en los aspectos de página de ingreso a la aplicación (landing page), reglas de negocio que regirán a su proceso (el juego), casos de uso que tendrá su aplicación, funcionalidades, botones, “pestañas”, estructura de las páginas y otros aspectos que permitan entender el juego. En síntesis, cómo esperan que se comporte su aplicación

En particular, se deberá entregar un documento con:

1. Tipo de juego que implementarán
2. Las características del juego (que debe contener los aspectos mínimos según el tipo de juego)
3. Las reglas detalladas del juego a nivel de usuario
4. Las reglas que tendrá el servidor para tratar cada turno (que se hará primero, que después, cómo se dirige un empate, cómo se determina al ganador, etc.)
5. El diseño de la página “landing page” que tendrá un “acerca de..”, las reglas y otra información de carácter general. La navegación por esa página
6. El registro de usuario
7. El registro para comenzar o integrarse a un juego
8. La navegación para hacer una jugada y para enviar la jugada
9. La interfaz de usuario para visualizar los mapas correspondientes
10. Tableros, despliegue, mecánica de juego (botones para acciones, radio buttons, listas desplegables, etc.) puntuación, mensajes, etc.
11. Modelo de datos preliminar para su juego con las entidades principales y la relación entre ellas
12. Diseño orientado a objetos para “las piezas” del juego (territorio, personajes, unidades, especialistas, etc.) con las clases principales y atributos relevantes

Páginas de ejemplo los pueden inspirar en este tipo de juegos basados en turnos:

<https://ds5.sim-metry.net/index.php/main/index>

<https://www.22ndfleet.com/division/star-trek/>

**La entrega se realizará entregando un documento en PDF que incluya todo lo anterior.**

*NOTA: Si tienen un programa de mockups, por ejemplo, que genera otro formato distinto, pregúntenle al ayudante asignado a su grupo si se puede y las restricciones a ese tipo de entrega.*

**NO se aceptarán:**

- Entregas por mail (ya sea al profesor o ayudantes)
- Entrega de documentos que no sea en PDF (salvo el acuerdo previo con el ayudante)

## **Sobre los mockups**

Los mockups no son un diseño final, sino más bien un esquema de los elementos que tendrá su aplicación de juego, cómo están organizados dichos elementos y algo de diseño preliminar (cuando vean las herramientas para hacer mockups se darán cuenta de los elementos con los que pueden trabajar). Pueden ver varios ejemplos de mockups en la web.

Pueden hacer los diagramas a lápiz y después escanearlos o pueden usar la herramienta de diseño que más les guste. Algunos ejemplos de herramientas online

y colaborativas son Balsamiq, Cacao y Mockingbird, aunque en general tienen sólo trials gratuitos. Una herramienta de escritorio y completamente gratuita es Pencil.

Finalmente deben entregar un PDF que se pueda entender con facilidad.

## **Sobre el Modelo de Datos**

---

Representen su modelo de datos con un diagrama ER (Entidad Relación). Ésto les permitirá tener algo intermedio que luego podrán mapear tanto a tablas de bases de datos como a un modelo orientado a objetos. Traten de capturar lo más que puedan, pero no es necesario que sea absolutamente exhaustivo (muy probablemente cambiará durante el desarrollo). Sólo los atributos que obviamente se van a necesitar. Privilegien la simplicidad cuando sea posible.

Hay muchas herramientas para generar este tipo de diagramas y pueden utilizar la que más les guste. Si quieren un ejemplo: <https://erdplus.com>.

## **Condiciones y restricciones**

---

1. La aplicación **debe** desarrollarse usando: Node.js, koa, HTML, CSS, Javascript (pueden usar jquery y react también)
2. Para base de datos **deberán** usar Postgres y para conexión a la base de datos pueden utilizar pg, knex o un ORM como sequelize
3. Pueden usar también EJS, SACC/SCSS, less, stylus u otros preprocesadores de CSS
4. **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 Twitter Bootstrap o Zurb Foundation
5. Cualquier package o biblioteca adicional que quieran usar debe ser aprobado previamente por el equipo docente

## **Recomendaciones**

---

1. Piensen en distintas opciones de presentación y la mejor forma de incorporar la funcionalidad en su tarea. No se queden con la primera idea que le venga a la cabeza
2. Aprenderán mucho más si trabajan colaborativamente en su grupo, como equipo en lugar de repartirse el trabajo y realizarlo como unidades independientes
3. Si hay problemas con su compañero(a) y no lo pueden resolver, comuníquense con el profesor
4. Diseñen muy bien las reglas, el entorno, los turnos, la resolución de empates y conflictos

5. Planifiquen el trabajo para que les permita la colaboración entre los integrantes del equipo
6. Pregunten y consulten, usen foro, colaboren entre ustedes (NO COPIEN) los ayudantes están para apoyarlos
7. **Trabajen con tiempo, no esperen a último momento para comenzar con la tarea o despejar dudas**
8. Comiencen con una interfaz de usuario (UI) simple; **no es necesario que su aplicación se vea "bonita" o "completa" desde el comienzo**
9. No traten de resolver aún detalles específicos de integración, comunicación, consultas a BDD u otros temas de implementación ajenos al alcance de esta primera entrega
10. Siempre podrán, justificadamente, cambiar alguna regla, mejorar algún aspecto del juego, etc.

## **Dudas**

---

Para que todo el curso se vea beneficiado, hagan sus preguntas sobre las tareas, a través del foro del curso que se habilitará para tal efecto. **No se responderá ninguna duda de las tareas por e-mail.**