



Entrega 1

PROYECTO SEMESTRAL
ENTREGA: 26 de abril, 23:59.

1. Objetivos

- Definir modelo de la plataforma mediante un diagrama E/R.
- Comenzar el desarrollo de funcionalidades.

2. Diagrama E/R

Antes de comenzar con el desarrollo de funcionalidades, una práctica indispensable es realizar un modelo de datos a partir de sus relatos de usuario. Esto les proveerá una visión general de los datos que debe manejar su aplicación antes de empezar y previene hacer grandes cambios durante el desarrollo, debido a detalles no considerados. Para esto, deben generar un diagrama E/R que muestre el modelo de datos considerado.

Este diagrama puede seguir el formato que proporcionamos como ejemplo en la siguiente imagen, o puede utilizar otro si se sienten más cómodos (como el enseñado en bases de datos), siempre y cuando esté debidamente documentado y cumpla con mostrar lo solicitado.

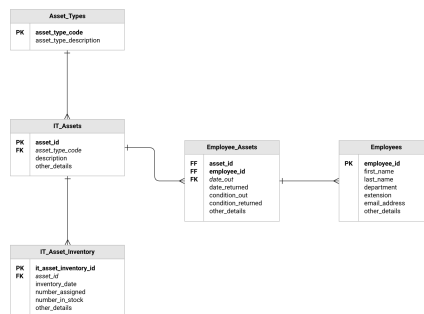


Figura 1: Ejemplo de Diagrama. [Fuente](#)

3. Funcionalidades y requisitos mínimos de desarrollo

Dada la planificación de Sprint acordada durante el *Sprint Review*, deben adherirse lo más posible a ella en cuanto al desarrollo de funcionalidades. También, deben mantener los requisitos mínimos de desarrollo. Es decir, su trabajo debe:

- Reflejarse en su tablero de *Projects*, donde se plasme un correcto flujo de tarjetas por cada funcionalidad trabajada.
- Seguir la filosofía de *Gitflow* para gestionar su repositorio.
- Respetar su guía de estilo revisada por *Rubocop*.
- Montar en producción su aplicación a *Render*.
- Tener el código actualizado de su aplicación en la rama *main* de su repositorio en Github.

4. *Sprint Review*

Tras esta entrega, nuevamente deben agendar su reunión de *Sprint Review* con su *product owner*. Esta debe realizarse durante los 3 días hábiles siguientes a la fecha de entrega.

5. Entrega

En su repositorio de *GitHub* asignado deben alojar el código de su aplicación. Incluyan un archivo `README.md` con la dirección de su aplicación montada en *Render*. La fecha de entrega es **26 de abril, 23:59**. Se revisará el último *commit* en su rama *main* antes de dicho plazo.

6. Distribución del puntaje

La entrega consta de un total de 24 puntos distribuidos de la siguiente manera:

- Diagrama E/R: 5 ptos
- Aplicación en Render: 3 ptos
- Projects: 2 ptos
- Rubocop: 2 ptos
- Gitflow: 2 ptos
- Features logradas: 6 ptos
- Calidad de features: 4 ptos

7. Política de integridad académica

Los/as estudiantes de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile deben mantener un comportamiento acorde a la Declaración de Principios de la Universidad. En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los/as estudiantes que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada estudiante conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería.

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un/a estudiante para los efectos de la evaluación de un curso debe ser hecho individualmente por el/la estudiante, sin apoyo en material de terceros. Por “trabajo” se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros.

En particular, si un/a estudiante copia un trabajo, o si a un/a estudiante se le prueba que compró o intentó comprar un trabajo, obtendrá nota final 1.1 en el curso y se solicitará a la Dirección de Pregrado de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral.

Por “copia” se entiende incluir en el trabajo presentado como propio, partes hechas por otra persona. En caso de que corresponda a “copia” a otros estudiantes, la sanción anterior se aplicará a todos los involucrados. En todos los casos, se informará a la Dirección de Pregrado de la Escuela de Ingeniería para que tome sanciones adicionales si lo estima conveniente.

También se entiende por copia extraer contenido sin modificarlo sustancialmente desde fuentes digitales como Wikipedia o mediante el uso de asistentes inteligentes como ChatGPT o Copilot. Se entiende que una modificación sustancial involucra el análisis crítico de la información extraída y en consecuencia todas las modificaciones y mejoras que de este análisis se desprendan. Cualquiera sea el caso, el uso de fuentes bibliográficas, digitales o asistentes debe declararse de forma explícita, y debe indicarse cómo el/la estudiante mejoró la información extraída para cumplir con los objetivos de la actividad evaluativa.

Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la referencia correspondiente.

Lo anterior se entiende como complemento al Reglamento del Estudiante de la Pontificia Universidad Católica de Chile (<https://registrosacademicos.uc.cl/reglamentos/estudiantiles/>). Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.

Compromiso del Código de Honor

Este curso suscribe el Código de Honor establecido por la Universidad, el que es vinculante. Todo trabajo evaluado en este curso debe ser propio. En caso de que exista colaboración permitida con otros/as estudiantes, el trabajo deberá referenciar y atribuir correctamente dicha contribución a quien corresponda. Como estudiante es un deber conocer el Código de Honor (<https://www.uc.cl/codigo-de-honor/>)