



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS DE COMPUTACIÓN
IIC2143 – INGENIERÍA DE SOFTWARE (I/2018)

Entrega 0

PROYECTO SEMESTRAL
ENTREGA: 29 DE MARZO, 19:59.

1. Objetivos

- Definir los requisitos funcionales de su proyecto, en forma de relatos de usuario.
- Contactarse y reunirse con su *product owner*.
- Configurar el ambiente de desarrollo y producción de su aplicación.

2. Relatos de usuario

Antes de comenzar con el desarrollo mismo de su aplicación hace falta definir los requisitos que debe satisfacer. Como equipo, deben generar un conjunto de relatos de usuario para cada una de las funcionalidades. Como han visto en clases, esto incluye definir un **rol de usuario**, una **meta** y el **beneficio** obtenido. Además, cada relato debe incluir **condiciones de satisfacción**. Pueden construirlas en formas de tarjetas con reverso, e incluso agruparlas en forma de *storyboard*.

3. Reunión con *product owner*

Como los requisitos son definidos y discutidos con su cliente, es necesario que se reúnan con su *product owner* antes de la entrega. Deben contactarse y agendar una reunión con **todos los integrantes presentes**. De esta forma pueden discutir alcances no claros o definir más funcionalidades de las ya establecidas. Recuerden compartir además su tablero *Trello* con su *product owner*.

4. Ambiente de desarrollo

A su vez, es necesario comenzar con la configuración de ambiente de desarrollo. Para eso, cada grupo debe lanzar a producción mediante *Heroku* una aplicación mínima de *Ruby on Rails* que muestre algún mensaje del estilo “*Hello World*”.

Deben incluir una configuración (no vacía) de *Rubocop* a ser utilizada durante el resto del desarrollo, está puede

cambiar en el futuro, pero es necesario su proyecto pase la revisión¹ de *Rubocop* dada su configuración. Podrán encontrar una [configuración](#) de ejemplo en el repositorio del curso, además de un archivo [.gitignore](#) que considera archivos de *Ruby on Rails*.

Finalmente, considerar que para esta entrega no se evaluará la utilización de *Docker* pero si se evaluará individualmente para la siguiente.

5. Entrega

En su repositorio de *GitHub* asignado deben crear su aplicación mínima. Incluyan un archivo `README.md` con la dirección de su aplicación montada en *Heroku*. Además, en una carpeta llamada **relatos** deben incluir el/los documentos de relatos de usuario. La fecha de entrega es 29 de Marzo, 19:59. Se revisará el último *commit* antes de dicho plazo.

6. Política de integridad académica

Los alumnos 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 alumnos que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada alumno conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería en el SIDING.

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un alumno para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno, 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. Si un alumno copia 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. 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 Alumno de la Pontificia Universidad Católica de Chile. Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.

¹Existen comandos que arreglan errores simples, como `rubocop -a`