



# **Universidad ORT Uruguay**

## **Facultad de Ingeniería**

### **Obligatorio - Ingeniería de Software Ágil 2**

[Link al repositorio](#)

### **Entrega 2**

Integrantes:

Danilo Biladoniga - 231749

Tomás Núñez - 257564

Germán Oller - 242312

# Índice

<b>Definición del proceso de ingeniería en el contexto de KANBAN</b>	<b>2</b>
Roles dentro del equipo	2
Ceremonias	2
<b>Explicación del tablero y su vínculo con el proceso de ingeniería</b>	<b>3</b>
<b>Configuración del pipeline y su vínculo con el tablero</b>	<b>4</b>

# Definición del proceso de ingeniería en el contexto de KANBAN

## Roles dentro del equipo

Mantuvimos los mismos roles de la anterior entrega.

Miembro	Rol/es
Danilo Biladóniga	Developer, Tester
Tomás Núñez	Developer, Tester, Product Owner
Germán Oller	Developer, Tester

## Ceremonias

Al igual que en la anterior entrega, realizamos 3 tipos de ceremonias distintas. No vimos la necesidad de realizar una kickoff, pero mantuvimos las standup y la retrospective. La nueva ceremonia fue la review, en la cual el equipo se reunió con el PO (Product Owner) para analizar la resolución de los issues seleccionados.

# Explicación del tablero y su vínculo con el proceso de ingeniería

En la anterior entrega no modificamos código, por lo que utilizamos solamente un tablero con 3 columnas.

To do	Doing	Done

Sin embargo, vimos la necesidad de actualizar este tablero, para adaptarlo a los nuevos requerimientos.

To do	TDD	Doing	Review	Done

Agregamos 2 nuevas columnas a las que ya teníamos.

La columna To do no se vió modificada, en ella indicamos las tareas pendientes por realizar, incluídos los bugs que seleccionamos para solucionar.

TDD es una de las nuevas columnas. Se utiliza para indicar aquellos bugs que se encuentran en desarrollo mediante TDD.

Otra columna que no sufrió cambios fue Doing. Las tareas que se están realizando son colocadas en esta columna. Excepcionalmente, también colocamos un issue que no necesitaba TDD.

La última de las 2 nuevas columnas es Review, que utilizamos para indicar aquellos issues que se encuentran disponibles para revisar, ya sea por quién lo está resolviendo o por algún otro miembro del equipo.

Por último, Done fue utilizada de igual manera que en la anterior entrega, para indicar las tareas o issues finalizados.

El flujo es el siguiente:

- Tareas: To Do -> Doing -> Done
- Issues: To Do -> TDD -> Doing (excepcionalmente) -> Review -> Done

## Configuración del pipeline y su vínculo con el tablero

Utilizamos GitHub Actions para configurar nuestro pipeline.

Buscando llevar un control exhaustivo de los posibles errores decidimos que, al momento de realizar un Pull Request, se realicen validaciones automáticas sobre el frontend y el backend. Si el cambio que se quiere mergear causa que o bien el frontend no compile, o bien alguna de las pruebas unitarias del backend falle, el propio GitHub no nos permitirá realizar el merge. Esta validación se suma a la ya mencionada en la anterior entrega, de contar con la autorización de los demás integrantes del equipo.

Cómo estamos utilizando la metodología trunk-based, el Pull Request se debería realizar cuando el issue pasa a la columna Done del tablero, por lo que para poder considerar terminado el issue se utiliza previamente el pipeline.