



# **Universidad ORT Uruguay**

## **Facultad de Ingeniería**

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

[Link al repositorio](#)

### **Entrega 3**

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

Mantuvimos los 3 tipos de ceremonias realizadas en la anterior entrega, las standup, la retrospective y la review.

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

En la anterior entrega simplemente realizamos corrección de errores, por lo que utilizamos un tablero con 5 columnas.

To do	TDD	Doing	Review	Done

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

To do	Requirement definition	Test cases implementation	Frontend implementation	Backend implementation	Testing	Refactoring	Done

Modificamos nuestro tablero con el objetivo de reflejar el uso de BDD en el proyecto. Las columnas coinciden con el ciclo de esta forma de desarrollo, separando la etapa de Application Implementation en Frontend implementation y Backend implementation.

## Configuración del pipeline y su vínculo con el tablero

No modificamos la configuración de nuestro pipeline. Al realizar un Pull Request, es necesario que tanto el backend como el frontend compilen, y los test unitarios del backend deben arrojar resultados positivos. En caso de que alguna de las condiciones no se cumpla, no es posible realizar un merge.

El objetivo de fijar estas condiciones en el pipeline es minimizar la posibilidad de cometer errores al agregar código nuevo. Idealmente, se realiza el Pull Request en el momento que la issue pasa a la columna Done, habiendo recorrido previamente todo el ciclo de BDD.