

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

Para esta entrega, hemos realizado una actualización en el tablero de GitHub con el objetivo de mantener un vínculo con el proceso de ingeniería definido previamente. También hemos realizado cambios para hacer más claros los nombres de las columnas y las actividades que deben realizarse en cada una de ellas.

En primer lugar, hemos agregado la columna **Tests de integración**, para reflejar la nueva actividad del proceso de ingeniería. En esta columna, se deberá realizar los tests exploratorios utilizando la herramienta Selenium. El objetivo de esta actividad es asegurarse de que todas las partes integradas del sistema funcionen correctamente y se comporten como se espera.

En segundo lugar, hemos decidido dividir la columna de **Codificación y Pruebas Unitarias** en dos columnas separadas: **Pruebas Unitarias** y **Codificación**, para reflejar la división realizada en el proceso de ingeniería. Con esto logramos una mayor claridad en las responsabilidades y actividades, lo que a su vez facilita el seguimiento y control del proceso de ingeniería en el tablero de GitHub.

Como resultado de la automatización de las pruebas de integración, actividad que ahora se refleja en la columna Tests de Integración, nos vimos en la necesidad de eliminar la columna de **Integración** debido a que perdió su sentido. Este cambio refleja a su vez las modificaciones del proceso de ingeniería.

Además, para adaptarnos a los ajustes realizados, hemos agregado la columna **Refactor**. Esta nueva columna tiene como objetivo centralizar la actividad de Refactorización del código del proceso de ingeniería, permitiéndonos mejorar la estructura, rendimiento y legibilidad del mismo sin modificar su comportamiento funcional.

Finalmente, hemos realizado un cambio en el nombre de la columna **BDD feature**, denominándose **Definición de Requerimientos**. Esta modificación tiene como objetivo facilitar la identificación y comprensión de la actividad que se está llevando a cabo, incluso para aquellos que no están familiarizados con el término "BDD". Al renombrar la columna, brindamos una descripción más clara y accesible de la etapa en la cual se crean las user stories y se establecen los criterios de aceptación. De esta manera, se promueve una mejor comprensión y seguimiento del proceso de definición de requerimientos en el tablero de GitHub.

En síntesis, el tablero resultante presenta las siguientes columnas.

- **To Do:** En esta columna se registran las actividades a realizar, es decir, el backlog. Aquí se agregan nuevas funcionalidades e issues.
- **Definición de Requerimientos:** En esta etapa, anteriormente llamada BDD feature, se crean las user stories para las nuevas funcionalidades, utilizando el formato "Como, Quiero, Para", y se presentan los criterios de aceptación como escenarios BDD con el formato "Dado, Cuando, Entonces".

- **Pruebas Unitarias:** Siguiendo la metodología BDD, se escriben los escenarios de testing para las actividades que incluyen desarrollo de backend.
- **Codificación:** En esta columna se desarrolla el código correspondiente. Para las funcionalidades que tengan pruebas unitarias, se deberá comprobar que todos los tests estén correctos antes de avanzar.
- **Refactor:** En esta columna debemos realizar el proceso de refactorización del código.
- **Tests de integración:** Agregamos esta columna para reflejar la nueva actividad del proceso de ingeniería. En esta etapa se llevan a cabo los tests exploratorios utilizando la herramienta Selenium.
- **Revisión del producto:** En esta columna se realiza una revisión del producto junto con el product owner para verificar que el producto cumpla con los criterios de aceptación.
- **Retrospective:** En esta etapa se evalúa el trabajo que se ha realizado, revisando las métricas y la eficiencia del equipo mediante el método DAKI.
- **Done:** Finalmente, en esta columna se registran las actividades finalizadas.

Para brindar mayor claridad, describiremos el flujo de trabajo de cada tarjeta de acuerdo a su tipo en el tablero:

- Si es un **issue**, el camino es:
  - To Do
  - Pruebas Unitarias (en caso de que incluya backend)
  - Codificación
  - Refactor
  - Tests de integración
  - Revisión del producto
  - Retrospective
  - Done
- Si es una **user story de backend**, el camino es:
  - To Do
  - Definición de requerimientos
  - Pruebas Unitarias
  - Codificación
  - Refactor
  - Tests de integración
  - Revisión del producto
  - Retrospective
  - Done
- Si es una **user story de frontend**, el camino es:
  - To Do
  - Definición de requerimientos
  - Codificación
  - Refactor
  - Tests de integración
  - Revisión del producto

- Retrospective
- Done