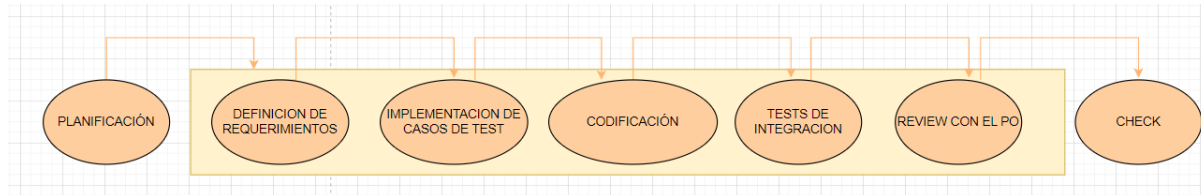


Proceso de ingeniería (4 versión)

El único cambio que sufrió el proceso sobre la entrega pasada fue que se actualizó como se realiza el “Test de Integración”, ya que pasó de ser realizado de manera manual a automática con la introducción de los tests de Selenium.



Planificación:

- **¿Quién lo hace?** El equipo de gestión de proyectos y/o ingenieros de software (Development Team) encargados de planificar y supervisar el proyecto. (Diego, Federico, Mateo)
- **¿Cómo lo hace?** Se realizan reuniones y discusiones con el cliente (El profesor) y el equipo de desarrollo para establecer los objetivos, requisitos y cronogramas del proyecto, y se elabora un plan detallado con las actividades a realizar y los recursos necesarios para su ejecución.
- **¿Cuándo se hace?** Antes de realizar el resto del trabajo.
- **¿Qué se obtiene?** Un plan de trabajo, que en nuestro caso es la cuarta versión del proceso de ingeniería.

Definición de requerimientos:

- **¿Quién lo hace?** El equipo de desarrollo. Junto a los stakeholders (Diego, Federico, Mateo)
- **¿Cómo lo hace?** Se crean las user story para cada funcionalidad que se desea desarrollar y se agregan a su respectiva tarjeta en el tablero Kanban. La tarjeta debe tener por cada **User Story** definido **Como Quiero Para** y sus **Criterios de Aceptación/Escenarios** que siguen el formato **Dado Cuando Entonces**.
- **¿Cuándo se hace?** Después de la planificación y antes de la Implementación de casos de test.
- **¿Qué se obtiene?** Las User Story con sus Criterios de Aceptación/Escenarios correspondientes a las funcionalidades que se definieron para esta entrega y un desarrollador asignado para implementar la funcionalidad.

Implementación de casos de test:

- **¿Quién lo hace?** El equipo de desarrollo y testers (Diego, Federico, Mateo)
- **¿Cómo lo hace?** Para el backend, se usa SpecFlow para escribir casos de prueba en lenguaje natural basados en los escenarios definidos en las historias de usuario.

Para el frontend, se usa Selenium para crear casos de prueba basados en los escenarios definidos en las historias de usuario.

- **¿Cuándo se hace?** Después de haber definido los requerimientos y antes de la codificación del software.
- **¿Qué se obtiene?** La implementación de casos de prueba automatizados que verifican el cumplimiento de los requerimientos y aseguran el correcto funcionamiento del software que se quiere implementar.

Codificación:

- **¿Quién lo hace?** El equipo de desarrollo (Diego, Federico, Mateo)
- **¿Cómo lo hace?** El desarrollador que escribió los tests de la fase anterior agrega/cambia el código en producción para que pasen los mismos
- **¿Cuándo se hace?** Después de la creación de la implementación de casos de test y antes de ejecutar los test de integración.
- **¿Qué se obtiene?** Una nueva funcionalidad del sistema que hace pasar las pruebas de la etapa anterior y que aún no ha sido verificada por el PO.

Test de Integración:

- **¿Quién lo hace?** El equipo de desarrollo de software en colaboración con los testers o QA (Diego, Federico, Mateo).
- **¿Cómo lo hace?** Se ejecutan los tests de integración definidos en Selenium. Si ocurren errores, se vuelve al paso de "Implementación de test" en pos de crear tests que tengan en cuenta los errores que ocurrieron.
- **¿Cuándo se hace?** Después de la codificación.
- **¿Qué se obtiene?** Un software completamente integrado y funcional que satisface los requerimientos y casos de prueba definidos en el tablero Kanban.

Review con el PO (Product Owner):

- **¿Quién lo hace?** El equipo de desarrollo y el product owner (Diego, Federico, Mateo)
- **¿Cómo lo hace?** El equipo de desarrollo se reúne con el PO y le muestra el resultado de la implementación de las tarjetas que se habían seleccionado para ser realizadas para esta entrega.
- **¿Cuándo se hace?** Después del Test de Integración.
- **¿Qué se obtiene?** Si el product owner encuentra que se cumplió la "definition of done" de la tarjeta, se da como hecha y se obtiene un incremento de valor. En cualquier caso se obtiene también un video de la revisión de los bugs y un acuerdo entre el equipo de desarrollo y el Product Owner sobre la calidad y funcionalidad del software.

Check:

- **¿Quién lo hace?** El equipo de ingenieros de software(Development Team) encargado de garantizar la calidad del software.(Diego, Federico, Mateo)
- **¿Cómo lo hace?** Se realiza una retrospectiva (utilizando DAKI) con un moderador.
- **¿Cuándo se hace?** Después de la review con el PO
- **¿Qué se obtiene?**
 - Informe de avance de la etapa.
 - Detalle de registro de esfuerzo por tipo de tarea
 - Totales de registro de esfuerzo por la entrega
 - Video de la retrospectiva con el “SM”

Definition of done:

Planificación:

- Se definió y documentó la cuarta versión del proceso de ingeniería, donde se definieron las actividades a realizar y los recursos necesarios para su ejecución así como la “definition of done” y los roles de cada integrante.
- Se creó y documentó el tablero kanban y su vínculo con el proceso de ingeniería.

Definición de requerimientos:

- Para las features, el PO definió junto con el equipo de desarrollo los requerimientos para cumplir con las funcionalidades que se pre seleccionaron para esta entrega, Definieron las User Stories y sus **Criterios de Aceptación/Escenarios** que siguen el formato **Dado Cuando Entonces y se** escribieron las respectivas tarjetas en el kanban de github: <https://github.com/users/mateoCosta1/projects/3/views/1> en la columna Requirement Definition.

Implementación de casos de test:

- Los casos de prueba del backend se han escrito en SpecFlow utilizando el formato Gherkin y están basados en los escenarios definidos en las User Stories.
- Los casos de prueba del frontend se han escrito en Selenium y están basados en los escenarios definidos en las User Stories.
- Los casos de prueba han sido revisados y aprobados por el equipo de desarrollo y testers (Diego, Federico, Mateo).
- La implementación de casos de prueba ha sido documentada correctamente.
- Los casos de prueba se han integrado correctamente en el pipeline.
- Las pruebas fallan porque aún no se ha implementado el código que las haga pasar.

Con esto, se garantiza que la implementación de casos de prueba se ha realizado de manera completa y satisfactoria.

Codificación:

- Se modificó el código fuente de la aplicación en el backend de tal manera que los casos de prueba escritos en SpecFlow pasen.
- Se modificó el código fuente de la aplicación en el frontend de tal manera que los casos de prueba escritos en Selenium pasen.

Test de Integración:

- Los desarrolladores han integrado y probado el código implementado para asegurarse de que funciona correctamente con otros componentes del software.
- A todos los casos de prueba definidos en el tablero Kanban les corresponde un test definido en Selenium.
- Todos los casos de prueba definidos en Selenium han sido ejecutados con éxito en el ambiente de integración.
- Se han resuelto todos los defectos encontrados durante las pruebas de integración.
- Se han identificado y documentado los riesgos asociados al cambio integrado.

Review con el PO:

- Se validó la implementación de todas las tarjetas codificadas con el product owner en una reunión
- Dicha reunión está grabada y el video está disponible.
- Se tiene el Código de software nuevo así como los nuevos test en SpecFlow.
- Se tiene un documento de la retrospectiva con un Informe de avance de la etapa, el detalle de registro de esfuerzo por tipo de tarea, totales de registro de esfuerzo por la entrega, un video de la retrospectiva con el "SM" y un video de la revisión de los bugs con el PO.

Check:

- Se tiene un documento de la retrospectiva con un Informe de avance de la etapa, el detalle de registro de esfuerzo por tipo de tarea, totales de registro de esfuerzo por la entrega, un video de la retrospectiva con el "SM" y un video de la revisión de los bugs con el PO.