

**Ingeniería en Sistemas de Información**

**Cátedra Ingeniería y Calidad de Software**

**TRABAJO PRÁCTICO N° 8**

**SCRUM:**

**PLANIFICACIÓN DE RELEASE Y DE SPRINT**

*Curso*: 4K1

*Número de grupo:* 3

*Integrantes:*

* Beato Dominguez, Valentina- 86278
* Brocanelli Novillo, Valentina- 87030
* De los Santos, Martina - 87699
* Firpi Padro, Gonzalo - 85954
* Pacheco, Franco Agustín - 82419
* Rojo, Agustín - 85681

*Fecha de entrega:* 10/10/2023

*Docentes del curso:* Boiero, Gerardo

Crespo, Mickaela

**INTRODUCCIÓN:**

La gestión ágil de proyectos se ha convertido en una metodología ampliamente adoptada en el desarrollo de software debido a su capacidad para adaptarse a los cambios y entregar valor de manera continua. Uno de los aspectos centrales de esta gestión ágil es la adopción que hace SCRUM dentro de un conjunto de ceremonias para guiar el proceso de desarrollo.

En este trabajo, explicaremos en detalle la importancia de la planificación en el contexto de la gestión ágil de proyectos, centrándonos en la simulación de la ceremonias de sprint planning tomando como base un ejemplo práctico de “Taxi-Mobile” para aplicarlas.

Además, a lo largo de este trabajo, analizaremos los procesos involucrados en la Sprint Planning, desde la selección de historias de usuarios hasta la estimación del esfuerzo de cada uno de los miembros del equipos y las asignación/ división de las tareas.

**1- CONDICIONES DE CONTEXTO**

A continuación, se presentan las condiciones de contexto relevantes para el proyecto de Taxi Mobile:

**Disponibilidad de recursos:** Es importante tener en cuenta la disponibilidad de los recursos necesarios para el desarrollo del producto, incluyendo el equipo de desarrollo, el tiempo y cualquier otra herramienta o tecnología necesaria. Tenemos integrantes del equipo con distintos conocimientos tecnológicos y recursos.

**Restricciones de tiempo:** Se debe considerar cualquier restricción de tiempo que pueda afectar el desarrollo del producto. Por ejemplo, tenemos dos integrantes que van de vacaciones por tres días para ver un concierto.

**Requisitos legales y regulatorios:** Es esencial cumplir con todas las regulaciones y requisitos legales relacionados con la operación de servicios de taxis y aplicaciones móviles en el área geográfica.

**Infraestructura tecnológica:** Considerar la infraestructura tecnológica disponible para el desarrollo y despliegue del sistema, incluyendo servidores, redes y dispositivos móviles compatibles. Hay integrantes que no tienen notebook, lo cual significa que sólo podrán trabajar desde sus casas con sus computadoras de escritorio.

**Capacitación del equipo:** Asegurarse de que el equipo de desarrollo esté capacitado y tenga el conocimiento necesario para trabajar en el proyecto, incluyendo la familiarización con las tecnologías y herramientas utilizadas.

**2-PLAN DE RELEASE**

**Release Planning:**

* ¿Cuánto tiempo?
  + 60 días
* ¿Cuántos sprints?
  + 4 sprints
* ¿Duración del sprint?
  + 15 días
* ¿Qué voy a trabajar en cada sprint?

| Sprint | User Story | Story Points | Aclaración |
| --- | --- | --- | --- |
| 1° | -Loguear taxista  -Buscar taxis cercanos | 5 | El equipo está preparado para poder trabajar en ese tiempo con esa complejidad. Como equipo nos parece conveniente que el primer sprint no sea el más complejo para poder enfocarnos en entender bien el negocio, la incertidumbre es mayor en el primer sprint. |
| 2° | -Ocupar taxi -Pedir taxi | 7 | Objetivo del sprint: Funciones básicas para pasajeros y taxistas.  Con un objetivo crítico de implementar funciones clave, todos los miembros del equipo estarán disponibles y trabajarán para garantizar un enfoque total en este sprint desafiante. |
| 3° | -Ver ubicación de pasajero | 5 | Objetivo del sprint: Mejoras en la experiencia de taxistas  Para este sprint incorporamos únicamente una user story pero de un peso considerable. |
| 4° | -Notificar a taxista  -Liberar taxi | 5 | Objetivo del sprint: Completar el MVP y preparar para el lanzamiento  Este último sprint nos encargamos de realizar las últimas user story requeridas para la primer release del producto, teniendo una carga de trabajo similar a la que encontramos en sprints anteriores |

* ¿Qué US son parte del release?

Las pertenecientes al mvp

* Buscar taxis cercanos
* Loguear taxista
* Pedir taxi
* Ocupar taxi
* Ver ubicación de pasajero
* Notificar a taxista
* Liberar taxi

**3-MINUTA PARA SPRINT 1 Y SPRINT BACKLOG**

| Sprint 1 |
| --- |
| Duración del sprint en días: 15 días |
| Objetivo del sprint: Configuración básica de la aplicación |
| Historias de usuario:   * Loguear taxista * Buscar taxis cercanos |
| Equipo Scrum:   * Valentina Brocanelli Novillo * Valentina Beato * Franco Pacheco * Gonzalo Firpi * Martina De Los Santos * Agustin Rojo |
| Capacidad del Equipo en Horas Ideales: 195 horas |

**Definition of done:**

* Todos los requisitos de la User Story se han implementado.
* Se han realizado pruebas exhaustivas, incluyendo pruebas unitarias, de integración, de aceptación y de regresión, y todas han pasado con éxito.
* El código ha sido revisado y aprobado por otros miembros del equipo.
* El código está completo y ha sido debidamente comentado para facilitar su comprensión y mantenimiento.
* La User Story se ha documentado adecuadamente, incluyendo especificaciones técnicas, instrucciones de uso y cualquier información relevante.
* Se ha creado y actualizado la documentación de usuario, asegurándose de que sea precisa y comprensible.
* El código ha sido almacenado en un repositorio de control de versiones, como Git, y está disponible para todo el equipo.
* Se ha seguido una buena práctica de control de versiones, con confirmaciones de código significativas y mensajes descriptivos.
* Se ha creado una rama específica en el repositorio para la User Story y se ha realizado la fusión (merge) correctamente.
* Se ha asegurado la compatibilidad y la integración adecuada del código en el repositorio principal o rama principal (por ejemplo, rama "master" o "main").
* Se han abordado y resuelto todas las dependencias y problemas relacionados con la infraestructura, garantizando que la implementación no cause conflictos ni interrupciones en el entorno de producción.

| US | Tareas | Horas ideales |
| --- | --- | --- |
| Loguear Taxista | 1. Diseño de la Interfaz de Inicio de Sesión  2. Desarrollo de la Pantalla de Inicio de Sesión  3. Integración con la Base de Datos  4. Pruebas unitarias  5. Pruebas de integración  6. Pruebas de autenticación  7. Pruebas de regresión  8. Documentación | 3 horas  3 horas  4 horas  6 horas  8 horas  8 horas  13 horas  5 horas |

Descripción de tareas:

1.Diseño de la Pantalla de Inicio de Sesión para Taxistas:

Crear un diseño detallado y amigable de la pantalla de inicio de sesión dirigida a los taxistas.

Obtener la aprobación del diseño por parte del equipo de diseño y desarrollo, asegurando que sea intuitivo y satisfaga las necesidades de los taxistas.

2.Implementación de la Pantalla de Inicio de Sesión:

Desarrollar la interfaz de inicio de sesión para taxistas, incorporando campos para el nombre de usuario y la contraseña.

Validar y procesar la entrada de datos para garantizar la autenticación segura de los taxistas.

3.Integración con la Base de Datos de Taxistas:

Conectar la pantalla de inicio de sesión con la base de datos de usuarios taxistas, permitiendo la verificación de credenciales.

Asegurar que las credenciales de inicio de sesión se autentiquen de manera efectiva y segura en la base de datos.

4.Pruebas Unitarias de la Pantalla de Inicio de Sesión:

Ejecutar pruebas exhaustivas a nivel de unidad para confirmar que la pantalla de inicio de sesión funcione de manera correcta.

Identificar y solucionar cualquier error o problema detectado durante las pruebas.

5.Pruebas de Integración con el Sistema:

Realizar pruebas de integración para garantizar la interacción adecuada de la pantalla de inicio de sesión con otros componentes del sistema.

Verificar que la autenticación y el flujo de datos sean coherentes en el contexto general de la aplicación.

6.Pruebas de Autenticación de Taxistas:

Llevar a cabo pruebas específicas de autenticación, incluyendo casos con credenciales válidas e inválidas.

Confirmar que los taxistas puedan iniciar sesión de manera segura y que el sistema maneje adecuadamente diferentes situaciones de autenticación.

7.Pruebas de regresión

Realizar pruebas de regresión en la aplicación para asegurarse de que las nuevas implementaciones o cambios no hayan introducido regresiones o problemas en funcionalidades previamente existentes.

8.Documentación

Documentar todos los aspectos relevantes del proceso, incluyendo el diseño, la implementación, la integración y las pruebas. Esto incluye la creación de documentación detallada que describa los pasos realizados y las decisiones tomadas en cada fase del proyecto.

| US | Tareas | Horas ideales |
| --- | --- | --- |
| Buscar Taxis Cercanos | 1. Diseño de la Interfaz de Búsqueda de Taxis Cercanos  2. Desarrollo de la Pantalla de Búsqueda de Taxis Cercanos  3. Integración con la Base de Datos de Taxis  4. Pruebas unitarias  5. Pruebas de Integración  6. Pruebas de regresión  7. Pruebas de Búsqueda de Taxis  8. Documentación | 5 horas  14 horas  6 horas  12 horas  13 horas  20 horas  5 horas  8 horas |

Descripción de tareas:

1.Diseño de la Interfaz de Búsqueda de Taxis Cercanos:

En esta fase, se iniciará creando esquemas de diseño iniciales para la pantalla de búsqueda de taxis cercanos. La idea es obtener la aprobación del diseño por parte del equipo de diseño y desarrollo, asegurando que sea intuitivo y cumpla con las expectativas de los usuarios.

2.Desarrollo de la Pantalla de Búsqueda de Taxis Cercanos:

Luego, nos centraremos en la implementación de la interfaz de búsqueda de taxis cercanos para los pasajeros. Esto implica mostrar una lista de taxis disponibles en función de la ubicación actual del pasajero. La pantalla deberá ser fácil de usar y proporcionar resultados precisos y actualizados.

3.Integración con la Base de Datos de Taxis:

Para que la búsqueda sea efectiva, es crucial conectar la pantalla de búsqueda con la base de datos de taxis disponibles. Esto permitirá la verificación de credenciales y garantizará que la lista de taxis se actualice en tiempo real en función de la disponibilidad.

4.Pruebas Unitarias de la Pantalla de Búsqueda de Taxis:

Con el desarrollo completo, se llevarán a cabo pruebas exhaustivas a nivel de unidad para confirmar que la pantalla de búsqueda de taxis funciona correctamente. Cualquier error o problema identificado se abordará de manera inmediata.

5.Pruebas de Integración con el Sistema:

Para garantizar que la pantalla de búsqueda se integre adecuadamente con otros componentes de la aplicación, se realizan pruebas exhaustivas de integración. Esto asegurará que la autenticación y el flujo de datos sean coherentes en el contexto general de la aplicación.

6.Pruebas de Regresión:

Además de las pruebas mencionadas, se realizarán pruebas de regresión para confirmar que las nuevas implementaciones no hayan afectado negativamente las funcionalidades existentes en la aplicación.

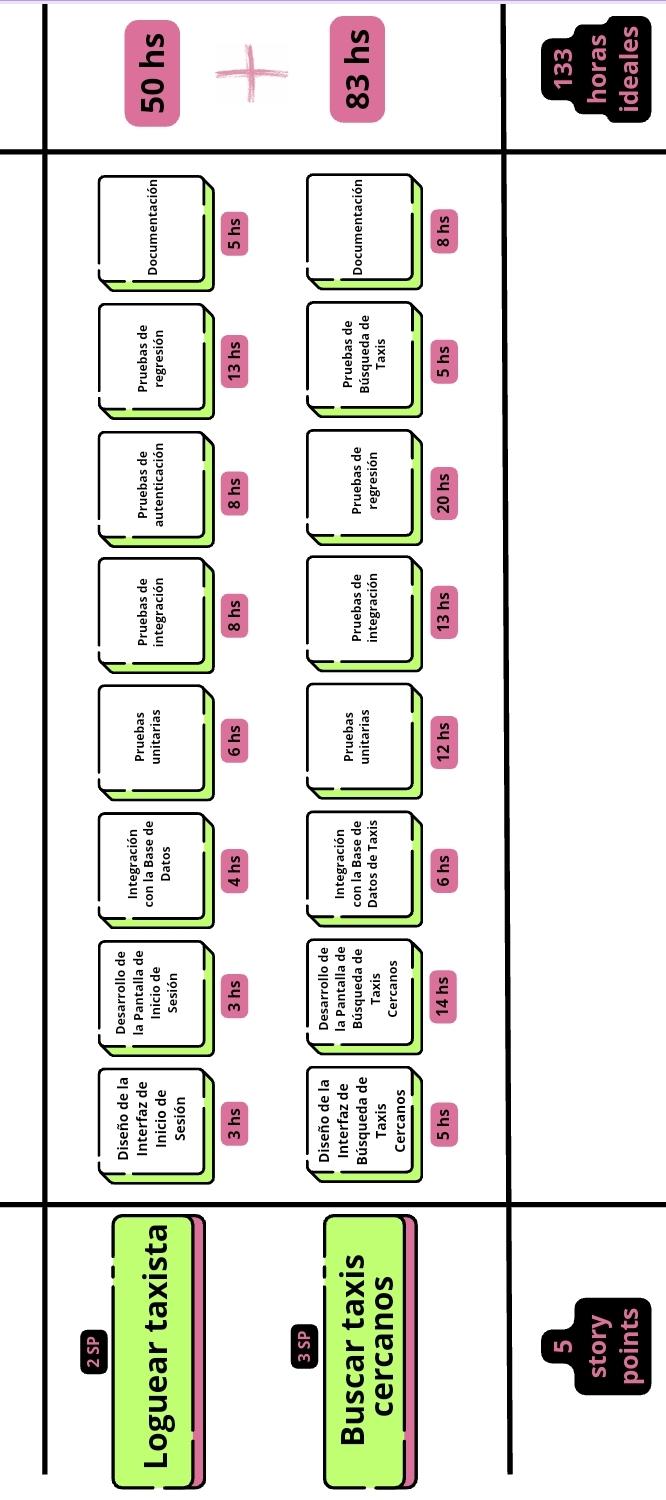
7.Pruebas de Búsqueda de Taxis:

Por último, se llevarán a cabo pruebas específicas de búsqueda de taxis cercanos en diferentes ubicaciones para asegurarse de que funcione de manera efectiva y satisfaga las necesidades de los usuarios.

8.Documentación:

Se creará documentación detallada que incluirá descripciones de la pantalla de búsqueda de taxis, instrucciones de uso para los usuarios y detalles técnicos para el equipo de desarrollo.

**Sprint backlog:**



**Sprint planning del sprint 1:**

**Sprint Planning**

Objetivo: Planificar y establecer los objetivos y entregables del sprint actual.

Participantes: Equipo de desarrollo, Product Owner, Scrum Master.

Tiempo dedicado: 4 horas.

Agenda:

1. Presentación de las Historias de Usuario: El Product Owner presenta las historias de usuario seleccionadas para el sprint, explicando su importancia y valor.

2. Discusión y Clarificación: El equipo de desarrollo tiene la oportunidad de hacer preguntas y aclarar cualquier duda sobre las historias de usuario presentadas.

3. Estimación: El equipo de desarrollo estima el esfuerzo necesario para implementar las historias de usuario utilizando una técnica de estimación, como el Planning Poker.

4. Selección de Historias: El equipo de desarrollo selecciona las historias de usuario que se compromete a completar durante el sprint.

5. Definición de Objetivos del Sprint: Se establecen los objetivos específicos del sprint, lo que se espera lograr al final del período.

6. Planificación de Tareas: El equipo de desarrollo descompone las historias de usuario en tareas más pequeñas y crea un plan de trabajo para el sprint.

**Daily Scrum**

Objetivo: Realizar reuniones diarias de seguimiento para mantener al equipo alineado y abordar cualquier impedimento.

Participantes: Equipo de desarrollo, Scrum Master.

Tiempo dedicado: 15 minutos.

Agenda:

1. Actualización del Progreso: Cada miembro del equipo comparte brevemente qué hizo ayer, qué planea hacer hoy y si ha encontrado algún obstáculo.

2. Identificación de Impedimentos: Se discuten y registran los obstáculos o problemas que impiden el progreso y se planifica cómo resolverlos.

3. Alineación del Equipo: Se asegura de que todos los miembros del equipo estén alineados con el objetivo del sprint y las prioridades actuales.

**Sprint Review**

Objetivo: Revisar y demostrar las historias de usuario completadas al final del sprint.

Participantes: Equipo de desarrollo, Product Owner, Stakeholders.

Tiempo dedicado: 2 horas

Agenda:

1. Presentación de Historias Completadas: El equipo de desarrollo presenta las historias de usuario que se completaron durante el sprint.

2. Demostración: Se realiza una demostración de las funcionalidades implementadas para que los Stakeholders puedan ver cómo funcionan.

3. Evaluación del Producto: El Product Owner y los Stakeholders evalúan el producto y proporcionan retroalimentación.

4. Revisión de objetivos: Se verifica si se han alcanzado los objetivos establecidos al comienzo del sprint.

5. Actualización del Backlog del Producto: Basado en la retroalimentación y los cambios, el Product Owner actualiza el backlog del producto con nuevas prioridades o requisitos.

**Sprint Retrospective**

Objetivo: Reflexionar sobre el sprint y buscar mejoras en el proceso y el equipo.

Participantes: Equipo de desarrollo, Scrum Master.

Tiempo dedicado: 1,5 horas.

Agenda:

1. Recopilación de Comentarios: Se recopilan comentarios del equipo sobre lo que funcionó bien durante el sprint y las áreas que necesitan mejora.

2. Identificación de Mejoras: El equipo discute y propone mejoras específicas para implementar en el próximo sprint.

3. Planificación de Acciones: Se acuerdan las acciones concretas que se tomarán para implementar las mejoras identificadas.

4. Compromiso de Mejora Continua: El equipo se compromete a implementar las mejoras y a mantener una cultura de mejora continua.

**Calculo de total de horas**

User Story 1: 50 horas

User Story 2: 83 horas

Sprint Planning: 4 horas

Daily Scrum: 3 horas 45 minutos

Sprint Review: 2 horas

Sprint Retrospective: 1 hora 30 minutos

Total Acumulado: 144 horas 15 minutos

Duración sprint: 15 días

| **Persona** | **Días Disponibles** | **Días para otras actividades scrum** | **Horas por dia** | **Horas de esfuerzo disponibles** |
| --- | --- | --- | --- | --- |
| **Agustin Rojo** | 12 | 2 | 3-4 | 30-40 |
| **Valentina Brocanelli Novillo** | 11 | 2 | 3-5 | 27-45 |
| **Valentina Beato Dominguez** | 12 | 2 | 4-5 | 40-50 |
| **Gonzalo Firpi** | 11 | 2 | 2-4 | 18-36 |
| **Franco Agustín Pacheco** | 10 | 2 | 2-4 | 16-24 |
| **Martina De Los Santos** | 10 | 2 | 3-5 | 24-40 |
| **Total** |  |  |  | 155-235 |

**Conclusión:**

Como conclusión podemos mencionar que el hecho de hacer las ceremonias SCRUM aplicado a un ejemplo, nos ha proporcionado una valiosa visión práctica de como SCRUM y la planificación adecuada trabajan en conjunto para lograr el éxito en el desarrollo de software. Hemos comprendido que la planificación adecuada es esencial para definir claramente el trabajo que se llevará a cabo durante un sprint, lo que facilita la entrega de valor continua y eficiente.

Además, a lo largo de esta experiencia, hemos aplicado los conceptos de gestión ágil de proyectos, destacando la importancia de la colaboración y la comunicación dentro del equipo de desarrollo. Hemos visto cómo la selección de historias de usuario, la estimación del esfuerzo y la división de tareas son pasos críticos en la Sprint Planning, y cómo estos elementos se integran en el Sprint Backlog para guiar la ejecución del sprint.

Finalmente, este trabajo nos ha permitido vivenciar el ambiente de SCRUM y apreciar cómo esta metodología ágil le permite a los equipos lograr una mayor flexibilidad ante necesidades cambiantes, ajustándose a estas gracias al enfoque iterativo que proporciona SCRUM.