



Trabajo Práctico N°8

Planificación de Release y de Sprint

Ingeniería y Calidad de Software

Profesores: Ing. Cecilia Massano

Ing. Georgina González

Fecha: 09/10/2024

Grupo N°9

1. Alexis Andermatten, 70287
2. Rodrigo Bertero, 74901
3. Francisco Funes, 73037
4. Joaquin Molla, 90019

Índice

Objetivo del trabajo.....	2
Breve introducción al dominio.....	2
Mínimo Producto Viable.....	3
Definition of Done del equipo.....	4
Equipo de trabajo.....	6
Explicación del contexto de cada release.....	7
Release planning.....	7
Sprint planning.....	8
Minuta del Sprint.....	8
Sprint backlog.....	9
Bibliografía consultada.....	10

Objetivo del trabajo

El siguiente trabajo tiene como objetivo simular la ceremonia de Scrum conocida como Sprint Backlog, en la cual definiremos el Sprint Backlog. Para lograr esto, explicaremos el contexto de la aplicación sobre la cual vamos a trabajar, el equipo de trabajo que conformará el team de scrum y todas las definiciones necesarias para desarrollar el producto de software.

Breve introducción al dominio

Taxi Mobile es una aplicación móvil diseñada para mejorar la experiencia de los pasajeros y taxistas al momento de solicitar y coordinar servicios de taxi. Este sistema web mobile permite a los pasajeros solicitar el taxi más cercano y hacer un seguimiento en tiempo real de su ubicación y tiempo estimado de llegada. A su vez, a los taxistas les permitirá recibir notificaciones de nuevas solicitudes y manejar el estado de sus taxis.

El sistema poseerá 3 roles principales, los cuales son:

1. **Pasajero:** Utiliza la aplicación para solicitar taxis y desea que el proceso sea rápido y sencillo. Busca minimizar el tiempo de espera y contar con información precisa sobre la ubicación del taxi.
2. **Taxista:** Utiliza la aplicación para recibir notificaciones de nuevos pedidos y para ver la ubicación de los pasajeros. Requiere una interfaz simple y de bajo esfuerzo para manejar las solicitudes sin distracciones.
3. **Administrador de central de taxis:** Supervisa el estado y ubicación de todos los taxis y pasajeros en tiempo real. Necesita acceso a una interfaz web para gestionar la flota y apoyar a los taxistas.

Mínimo Producto Viable

El MVP se centra en validar la funcionalidad esencial que conecta a los pasajeros con los taxistas, permitiendo:

- Solicitar taxis cercanos y ver su ubicación.
- Manejar el estado de los taxis (ocupado, libre, fuera de servicio).
- Registrar taxistas y pasajeros mediante un sistema de login básico.

De esta forma, el sistema de gestión y administración de taxis en la central se reserva para futuras iteraciones, enfocándonos primero en la interacción directa entre pasajeros y taxistas, para probar la idea del mercado.

Las User Stories incluídas dentro del MVP, clasificadas por rol son:

- **Taxista**
 - a. Loguear taxista
 - b. Ocupar taxi
 - c. Liberar taxi
 - d. Ver ubicación del pasajero
- **Pasajero**
 - a. Buscar taxis cercanos
 - b. Pedir taxi
 - c. Notificar a taxista solicitud de taxi

Definition of Done del equipo

La Definition of Done es un conjunto de reglas que determinan cuándo un elemento está terminado. Terminado significa listo para poner en producción a disposición del usuario.

Definition Of Done	
Desarrollo completo de la User Story	<ul style="list-style-type: none"><input type="checkbox"/> Todas las tareas asociadas a la US están completas.<input type="checkbox"/> El código ha sido revisado y aprobado por al menos otro miembro del equipo.<input type="checkbox"/> Se realizó una prueba funcional que confirma que la US cumple con los criterios de aceptación definidos.
Pruebas unitarias y de Integración	<ul style="list-style-type: none"><input type="checkbox"/> Se escribieron pruebas unitarias que cubren al menos el 80% de las funciones implementadas.<input type="checkbox"/> Todas las pruebas unitarias fueron ejecutadas y pasaron.<input type="checkbox"/> Todas las pruebas de integración fueron ejecutadas y pasaron.<input type="checkbox"/> Las pruebas automáticas de CI/CD fueron ejecutadas y pasaron.
UI/UX Verificada	<ul style="list-style-type: none"><input type="checkbox"/> La interfaz cumple con los principios de UI/UX y es coherente con el resto de la aplicación.<input type="checkbox"/> Se realizó una revisión de UX para garantizar que la funcionalidad es simple e intuitiva.<input type="checkbox"/> La interfaz es responsive y fue probada en dispositivos con

	distintas dimensiones.
Documentación Técnica y de Código	<ul style="list-style-type: none"><input type="checkbox"/> El código se encuentra documentado para facilitar su comprensión y mantenimiento.<input type="checkbox"/> El código se encuentra subido en el repositorio.<input type="checkbox"/> La documentación del sistema y del usuario se encuentra actualizada con la US asociada.<input type="checkbox"/> El código cumple con al menos 1 principio SOLID.
Desempeño y Seguridad	<ul style="list-style-type: none"><input type="checkbox"/> Se realizó al menos una prueba de carga que garantice que la funcionalidad de la US soporta un volumen de usuarios definido.<input type="checkbox"/> De ser aplicable, se implementaron medidas de seguridad (validaciones y autenticación).<input type="checkbox"/> Se revisaron posibles riesgos de seguridad.
Revisión y Aprobación de la US	<ul style="list-style-type: none"><input type="checkbox"/> La US ha sido revisada y aprobada por el Product Owner.<input type="checkbox"/> Se encontraron defectos críticos o bloqueantes y fueron resueltos.
Preparación para el próximo Sprint	<ul style="list-style-type: none"><input type="checkbox"/> El Product Backlog fue actualizado para reflejar el progreso y lo siguiente a desarrollar.<input type="checkbox"/> Lo aprendido se documentó para una revisión en la Sprint Retrospective.

Equipo de trabajo

Capacidad ideal del equipo:

Persona	Días disponibles en el Sprint	Días para otras actividades de Scrum	Horas ideales por día		Horas de esfuerzo disponibles	
Alexis	12	3	3	4	30	40
Francisco	12	3	4	6	52	78
Joaquín	12	3	3	5	33	55
Rodrigo	12	3	4	6	52	78
Total					167	251

Este es el cálculo de capacidad de nuestro equipo para un sprint ideal, considerando las horas ideales que cada miembro puede disponer para el proyecto por día teniendo en cuenta los horarios de cursado de materias y las horas dedicadas a estudiar. No hemos identificado que existan otras obligaciones o compromisos de los miembros del equipo que debamos considerar.

Adicionalmente se descontó el tiempo estimado dedicado a todas las ceremonias de SCRUM el cual consideramos que es de tres días para un sprint de tres semanas. Nos basamos en el siguiente cálculo: 6 horas sprint planning + 2 horas sprint review + 2 horas sprint retrospective + (15 minutos * 12) 3 horas de daily scrum = 13 horas para ceremonias aproximadamente. Como el promedio de horas ideales por día para cada miembro del equipo es de 4,375 horas consideramos que se destinarán 3 días a ceremonias de SCRUM.

Explicación del contexto de cada release

Release planning

Tiempo del release:

3 semanas por sprint * 4 cantidad de sprints = 12 semanas.

Duración del release: 12 semanas.

Cantidad de Sprints = 4.

Duración por Sprint = 3 semanas.

Generalmente los Product Owners prefieren los Sprints cortos y a los desarrolladores les gustan los Sprints largos, por tal motivo con el equipo llegamos a un acuerdo de que la duración de los Sprints será de 3 semanas. Consideramos que de esta manera los Sprints son suficientemente cortos para proporcionar agilidad en la organización y suficientemente largos para lograr un flujo y recuperarse de eventuales problemas que puedan aparecer durante el Sprint.

User stories a realizar en el release y trabajo por sprint:

Sprint N°	User Story	Story Points
1	Loguear taxista	2
1	Buscar taxis cercanos	3
2	Pedir taxi	5
3	Notificar a taxista solicitud de taxi	3
3	Ocupar taxi	2
3	Liberar taxi	2
4	Ver ubicación del pasajero	5
	Total SP	22

Sprint planning

Minuta del Sprint

Sprint Nro. 1

Duración del Sprint en días: 15 días

Objetivo del Sprint: durante el primer sprint haremos hincapié en desarrollar dos de las funcionalidades más importantes para el negocio, las cuales formarán parte del primer incremento de nuestro mínimo producto viable (MVP). Dichas funcionalidades son: Loguear taxista y Buscar taxis cercanos.

Equipo Scrum:

- Alexis Andermatten - Desarrollador Backend y Scrum Master
- Rodrigo Bertero - Desarrollador Frontend y Product Owner
- Francisco Funes - Desarrollador Backend y Tester
- Joaquin Molla - Desarrollador Frontend y Tester

Capacidad del equipo en horas ideales:

El Sprint inicial comenzará el día jueves 10/10 y finalizará el día jueves 31/10. Hemos considerado que no se trabajarán los días sábados, domingos ni feriados. Dentro de las tres semanas que durará el primer Sprint nos veremos afectados por un feriado el día 11/10. Nuestro equipo está conformado por cuatro integrantes, del cual un integrante viajará a Amsterdam durante 1 semana. A continuación se detallan en una tabla los días disponibles de cada integrante para el Sprint Nro. 1 y las horas ideales de esfuerzo estimadas.

Persona	Días disponibles en el Sprint	Días para otras actividades de Scrum	Horas ideales por día		Horas de esfuerzo disponibles	
Alexis	11	3	3	4	33	44
Francisco	5	1	4	6	20	24
Joaquín	11	3	3	5	33	55
Rodrigo	11	3	4	6	44	66
Total					130	189

Sprint backlog

User story	Tareas	Horas ideales
Loguear taxista	Programar frontend	7
	Programar backend	10
	Crear base de datos y tablas	4
	Investigación para realizar integración con Facebook	5
	Programar pruebas unitarias	6
	Realizar pruebas unitarias y de integración	3
	Corrección de errores	7
	Revisión de código por un par	4
	Crear documentación de usuario final	6
	Realizar pruebas en plataforma Mobile	5
	Realizar pruebas de aceptación de usuario	2

	Combinar código con la rama principal	1
Buscar taxis cercanos	Programar frontend	10
	Programar backend	14
	Crear bases de datos y tablas	6
	Programar pruebas unitarias	6
	Realizar pruebas unitarias y de integración	4
	Investigación para resolverla obtención de datos de geoposicionamiento y tiempos	6
	Investigación para mostrar por pantalla un mapa con la ubicación de los taxis cercanos	6
	Corrección de errores	7
	Revisión de código por un par	4
	Crear documentación de usuario final	8
	Realizar pruebas en plataforma mobile	6
	Realizar pruebas de aceptación de usuario	3
	Combinar código con la rama principal	3
Total		143

Bibliografía consultada

<https://www.atlassian.com/es/agile/scrum/sprint-planning>

<https://mamaqueesscrum.com/2020/07/09/que-es-la-definition-of-done/>

<https://asana.com/es/resources/agile-scrum-ceremonies>

Kniberg, H. (2007). *Scrum y XP desde las trincheras*. C4Media.

<http://infoq.com/minibooks/scrum-xp-fromthetrenches>