



## **Ingeniería y Calidad de Software**

### **SCRUM - Planificación de Release y de Sprint**

UTN FRC - 4k2 - Grupo 10

#### **Profesores:**

Ingeniera Cecilia Massano  
Constanza Garnero

#### **Alumnos:**

89819, Benjamin Pavon  
90606, Juan Pablo Lambertucci  
89978, Maria Paz Costamagna  
91185, Cerutti Joaquín  
86203, Moyano Santiago  
91122, Schmira Agustín

Fecha de presentación 11/10/2024

# Índice

Mínimo Viable Producto.....	
Plan de Release.....	
Sprint Planning.....	
Ceremonias Scrum.....	
Minuta Sprint.....	
Sprint Backlog.....	
Bibliografía.....	

# Mínimo Viable Producto

Frases verbales de las historias de usuario incluidas en el MVP (junto con Story Points):

## Taxista

- Loguear taxista - 2.
- Ocupar taxi - 2.
- Liberar taxi - 2.
- Ver ubicación del pasajero - 5.

## Pasajero

- Buscar taxis cercanos - 3.
- Pedir taxi - 5.
- Notificar al taxista solicitud de taxi - 3.

# Plan De Release

- ❖ Duración de los sprint: 2 semanas
- ❖ Cantidad de sprints: 3
- ❖ Duración total del proyecto: 6 semanas

Plan de release		
Duración de los sprint: 2 semanas Cantidad de sprints: 3 Duración total del proyecto: 6 semanas		
Numero de Sprint	User Storys	Total SP
SPRINT 1	Ver ubicación del pasajero - 5. Loguear Taxista -2 Ocupar Taxi -2	9
SPRINT 1	Notificar al taxista solicitud de taxi - 3. Liberar taxi - 2.	5
SPRINT 1	Pedir taxi -5 Buscar Taxi cercano -3	8

# Sprint Planning

## Miembros del equipo

1. Juan Pablo Lambertucci (Juampi)
2. Maria Paz Costamagna (Paz)
3. Benjamin Pavon (Benja)
4. Agustín Schmira (Agus)
5. Joaquin Cerutti. (Joaco)
6. Santiago Moyano (Santi)

## Capacidad del equipo

Integrante Grupo	Rol	Dias de Sprint disponible	Hrs por día disponible	Capacidad por integrante
Santi	FrontEnd Developer	11	3	33
Joaco	Tester	8	4	32
Agus	Adminsitrador DB	6	8	48
Paz	Diseñadora UX/UI	8	5	40
Benja	BackEnd Developer	9	4	36
Juampi	BackEnd Developer	10	5	50
Total				239

Capacidad total del equipo : 239 horas

## Contexto del equipo

El sprint durará 14 días, arrancando desde el lunes 14/10/2024 hasta el lunes 28/10/2024.

En estos 14 días no hay feriados por ende todos los días hábiles están disponibles para trabajar.

El integrante Santiago Moyano (Santi), tiene otro trabajo por ende solo puede dedicar 3 horas por día al proyecto, y es por eso que también trabaja 11 días, incluyendo un sábado.

Agustin Schmira, también tiene otro trabajo solo algunos días de la semana, por eso solo tiene 6 días de los 10 días hábiles, y es por eso que puede dedicarle 8 horas por día al proyecto.

El resto de los integrantes por ciertos motivos personales tiene los días disponibles aclarados más arriba.

## Ceremonias Scrum

**Sprint planning:** En un sprint de un mes, la sprint planning será de 8 horas, como nuestro sprint dura 2 semanas (14 días), el sprint planning tendrá una duración de 4 horas. Considerando que el equipo tiene 6 integrantes y que necesitaremos 4 horas de las horas disponibles por integrante, se consumen en total 24 horas de la capacidad total del equipo.

**Daily meetings:** Durarán 15 minutos, todos los días del sprint. Considerando los días disponibles por persona y que el sprint daily dura 15 minutos, el total de horas de la capacidad disponible que se consumirán serán 13 horas.

**Sprint Review:** En un sprint de un mes, el sprint review dura 4 horas, como nuestro sprint es de 14 días, el sprint review dura 2 horas. Considerando que somos 6 integrantes en el equipo y que necesitaremos 2 horas de las horas disponibles por integrante, consumiremos 12 horas de la capacidad disponible del equipo.

**Sprint retrospective:** En un sprint de 1 mes dura 3 horas, como nuestro sprint dura 14 días, la ceremonia durará 1.5 horas. Sabiendo que somos 6 integrantes y que necesitaremos 1.5 horas de las horas disponibles por integrante, consumiremos 9 horas de la capacidad disponible del equipo.

# Minuta Sprint

## Sprint Nro. 1

### Duración del Sprint en días: 14

**Objetivo del Sprint:** Implementar el logueo de taxista, visualización de ubicación del pasajero y actualizar el estado del taxi.

### Equipo Scrum:

- Juan Pablo Lambertucci - BackEnd Developer.
- Maria Paz Costamagna - Diseñadora UX/UI.
- Benjamin Pavon - BackEnd Developer.
- Agustín Schmira - Administrador DB.
- Joaquin Cerutti - Tester.
- Santiago Moyano - FrontEnd Developer.

### Definición de Hecho (DONE)

#### 1. Diseño revisado:

- El diseño de las interfaces que usarán los taxistas, ha sido revisado y aprobado.
- El código responsable debe estar comentado y pusheado a la rama correspondiente.
- El código debe cumplir con las reglas de nombrado.
- Debe implementar buenas prácticas.

#### 2. Funcionalidad completa:

- La funcionalidad del inicio de sesión de un taxista, cambiar el estado del taxi a ocupado, y visualizar la ubicación del pasajero está completamente implementada.
- La implementación del diseño de las interfaces está completamente realizada.

#### 3. Integración correcta:

- La funcionalidad para que los taxistas puedan ver la ubicación de los pasajeros está completamente integrada y operativa.

**4. Pruebas completadas:**

- Pruebas unitarias de cada una de las funciones (ocupar taxi, logueo del taxista, y visualizar ubicación del pasajero).
- Pruebas de integración con los sistemas que gestionan la asignación de pasajeros a taxistas.
- Pruebas de regresión para asegurarse de que no se estropee ninguna funcionalidad anterior.
- La funcionalidad para realizar las implementaciones siguen los estilos documentados en la hoja de estilo.

**5. Código en repositorio:**

- Todo el código ha sido revisado, aprobado e integrado en el repositorio.

**6. Cero defectos conocidos:**

- No se han encontrado defectos críticos que impidan el funcionamiento de las funcionalidades descritas.

**7. Prueba de aceptación realizada:**

- Se ha realizado una revisión con el Product Owner, validando que todas las user stories relacionadas a este Sprint, cumplen los criterios de DONE.

## Sprint backlog

Previsión	Conjunto de Tareas	Total de horas estimadas
Ocupar taxi (2)	<ul style="list-style-type: none"><li>• Utilizar librerías y funciones de React y Bootstrap para aplicación del taxista (0.5hs)</li><li>• Implementar código del botón que cambia de estado (6hs)</li><li>• Realizar lógica de cambio de estado usando patrón State (2hs)</li><li>• Arquitectura que soporte el manejo de taxis con estados en BD (9hs)</li><li>• Implementar control remoto de la BD con Data Grip (1hs)</li><li>• Realizar el procedimiento de almacenamiento en base que cambie el estado de libre a ocupado (2hs)</li><li>• Crear documentación (0.5hs)</li><li>• Implementar lenguajes de desarrollo mobile (4hs)</li><li>• Crear iconos (4hs)</li><li>• Definir esquema de colores (2hs)</li><li>• Automatizar pruebas (9hs)</li><li>• Implementar logs de errores (10hs)</li></ul>	<b>50</b>



Loguear taxista (2)	<ul style="list-style-type: none"> <li>• Utilizar librerías y funciones de React y Bootstrap para aplicación del taxista (2hs)</li> <li>• Realizar UI/UX del login (5hs)</li> <li>• Implementar integración con API de Facebook (5hs)</li> <li>• Implementar lógica de verificación de inicio de sesión con funcionalidad proveída por la API de Facebook (3hs)</li> <li>• Realizar lógica de extracción de datos del taxista (10hs)</li> <li>• Codificar lógica de controles de campos obligatorios (10hs)</li> <li>• Arquitectura que soporte el manejo de usuarios taxistas BD (1hs)</li> <li>• Realizar el procedimiento de almacenamiento en base de datos que tenga como salida los datos del taxista (2hs)</li> <li>• Crear iconos (2hs)</li> <li>• Definir esquema de colores (2hs)</li> <li>• Automatizar pruebas (6hs)</li> <li>• Implementar logs de errores (7hs)</li> <li>• Crear documentación (2hs)</li> </ul>	<b>57</b>
Ver Ubicación del pasajero (5)	<ul style="list-style-type: none"> <li>• Implementar integración con API de Google Maps para el sistema de posicionamiento online (GPS) en el celular del pasajero (20 hs)</li> <li>• Diseñar la interfaz gráfica del mapa donde se visualizará al pasajero (23 hs)</li> <li>• Integrar el sistema de mapas con la aplicación del taxista y la web del administrador de Central de Taxis (21 hs)</li> <li>• Desarrollar y ajustar la lógica de actualización en tiempo real de la ubicación del pasajero (implementar usando un Cronjob) (20 hs)</li> <li>• Desarrollar tablas en la BD para almacenar la información de ubicación (15 hs)</li> </ul>	<b>125</b>

	<ul style="list-style-type: none"><li>● Implementar la funcionalidad para obtener la ubicación del pasajero según el pedido de viaje (10 hs)</li><li>● Realizar pruebas de integración y unitarias (5 hs)</li><li>● Automatizar las pruebas de verificación de ubicación (5 hs)</li><li>● Revisión de código por otro integrante del equipo (4hs)</li><li>● Realizar documentación (2hs)</li></ul>	
Story points quemados en el sprint: 9		

## SPRINT NÚMERO 1 - SCRUM BOARD

STORY POINTS QUEMADOS: X

Período de Tiempo 14/10/2024 - 28/11/2024

User Stories

Tareas

Esfuerzo

Ver ubicación  
del pasajero  
- 5

Automatizar las  
pruebas de  
verificación de  
ubicación (5 hs)

Implementar la  
funcionalidad para  
obtener la  
ubicación del  
pasajero según el  
pedido de viaje (10  
hs)

Realizar pruebas de  
integración y unitarias  
(5 hs)

Implementar  
integración con API  
de Google Maps  
para el sistema de  
posicionamiento  
online (GPS) en el  
celular del pasajero  
(20 hs)

Revisión de  
código por otro  
integrante del  
equipo (4hs)

Diseñar la interfaz  
gráfica del mapa  
donde se  
visualizará al  
pasajero (23 hs)

Integrar el sistema de  
mapas con la aplicación  
del taxista y la web del  
administrador de  
Central de Taxis (21 hs)

Desarrollar tablas  
en la BD para  
almacenar la  
información de  
ubicación (15 hs)

Desarrollar y  
ajustar la lógica de  
actualización en  
tiempo real de la  
ubicación del  
pasajero usando  
un Cronjob (20 hs)

Realizar  
documentación  
(2hs)

125

<p><b>Logear Taxista - 2</b></p>	<div> <div>Utilizar librerías y funciones de React y Bootstrap para aplicación de taxista (2hs)</div> <div>Implementar lógica de verificación de inicio de sesión con funcionalidad proveída por la API de facebook (3hs)</div> <div>Realizar lógica de extracción de datos del taxista (10hs)</div> <div>Realizar el procedimiento de almacenamiento en base de datos que tenga como salida los datos del taxista (2hs)</div> <div>Crear documentación (2hs)</div> <div>Automatizar pruebas(6hs)</div> <div>Realizar UI/UX del login (5hs)</div> <div>Implementar integración con API de facebook(5hs)</div> <div>Codificar logica de control en campos obligatorios (10hs)</div> <div>Arquitectura que soporte el manejo de usuarios taxistas BD(1hs)</div> <div>Definir esquema de colores(2hs)</div> <div>Implementar logs de errores (7hs)</div> <div>Crear iconos (2hs)</div> </div>	<p>57</p>
<p><b>Ocupar Taxi -2</b></p>	<div> <div>Utilizar librerías y funciones de React y Bootstrap para aplicación del taxista (0.5hs)</div> <div>Realizar lógica de cambio de estado usando patrón state (2hs)</div> <div>Implementar control remoto de la BD con Data Grip (1hs)</div> <div>Implementar lenguajes de desarrollo mobile (4hs)</div> <div>Crear iconos (4hs)</div> <div>Automatizar pruebas (9hs)</div> <div>Realizar el código del botón que cambia de estado (6hs)</div> <div>Arquitectura que soporte el manejo de taxis con estados en BD ( 9hs)</div> <div>Realizar el procedimiento de almacenamiento en base que cambie el estado de libre a ocupado (2hs)</div> <div>Crear documentación (0.5hs)</div> <div>Implementar dependencias de google maps (0.5hs)</div> <div>Implementar logs de errores (10hs)</div> </div>	<p>50</p>
<p>Story points: 9</p>		<p>Horas ideales: 232</p>

## Bibliografía

- Material de cátedra Ingeniería y Calidad de Software - UTN FRC
- "SCRUM 2020, Planificación de release y sprint, Métricas Scrum.pdf" • La Guía Scrum, Ken Schwaber y Jeff Sutherland - Edición 2020.
- [ChatGPT.com](https://chatgpt.com)
- Ingeniería del Software, Roger S. Pressman - Séptima Edición.
- Estimación y planeamiento ágil, Mike Cohn - 2006