



Universidad Tecnológica Nacional

Ingeniería en Sistemas de la Información

## PRÁCTICO N° 8 - SCRUM - Release and Sprint Planning

Cátedra de Ingeniería de Software

Grupo N°9

Curso: 4K4

### Integrantes:

- |                          |               |
|--------------------------|---------------|
| • Acevedo Hernán         | Legajo: 72380 |
| • Aresu Barella, Bruno   | Legajo: 73446 |
| • Bardin, Tomás          | Legajo: 72783 |
| • Boné, Nicolás          | Legajo: 72624 |
| • D'uva, Micaela         | Legajo: 74300 |
| • Frattin Juan           | Legajo: 71939 |
| • Molina, Matias         | Legajo: 75477 |
| • Pereira Duarte, Martin | Legajo: 76367 |

### Docentes:

- Covaro, Laura Ines
- Robles, Joaquín Leonel
- Crespo, María Mickaela

Fecha de entrega: 04/05/2021

## Contexto del Equipo

### *Condiciones de contexto de release planning*

- Los sprints tendrán una duración de 2 semanas, considerando solamente los días hábiles (10 días hábiles)
- Considerando que los integrantes del equipo trabajan de Lunes a Viernes, se estima que la disponibilidad de cada integrante es un aproximado de 4 horas ideales por día hábil.
- Durante la ejecución del Release Plan no se consideran días feriados, ni periodos de vacaciones.
- En caso de presentarse alguna fecha de algún parcial o la entrega de algún trabajo práctico el tiempo que puede llegar a ser demandado fue considerado al momento de realizar la estimación aproximada de las horas ideales.
- El equipo no cuenta con la experiencia suficiente para estimar la **Velocidad**.
- El *Scrum Team* está compuesto por 8 personas, y cada una cuenta con experiencia suficiente en su rol:

Desarrollo	<ul style="list-style-type: none"><li>• Acevedo Hernán</li><li>• Boné Nicolás</li><li>• Pereira Duarte Martin</li></ul>
Scrum Master	<ul style="list-style-type: none"><li>• Molina Matias</li></ul>
Testing y QA	<ul style="list-style-type: none"><li>• Bardin Tomás</li><li>• Aresu Barella Bruno</li></ul>
UI/UX y Diseño Gráfico	<ul style="list-style-type: none"><li>• D'uva Micaela</li><li>• Frattin Juan</li></ul>

- Se considera que todos los miembros del *Scrum Team* realizan tareas de *Dev's*
- Se considera que la capacidad total ideal del equipo sin tener en cuenta las ceremonias de *Scrum*:
  - $10 \times 3 \times 8$  (días hábiles de sprint \* horas disponibles por miembro \* cantidad de miembros del equipo) = **240 horas ideales**.
- Se considera que la duración (en horas) de las ceremonias de *Scrum* durante el *Sprint* definido es:
  - **Sprint planning**: 4 horas
  - **Daily**: 15 minutos  $\rightarrow 15 \times 10 = 2:30$  horas
  - **Sprint review**: 2 horas
  - **Sprint retrospective**: 1 hora
- Se considera la capacidad total del equipo teniendo en cuenta las ceremonias de *Scrum* es:
  - $240 - 4 - 2.5 - 2 - 1$  (horas ideales sin ceremonias - duración del Sprint planning - duración de la Daily - duración del Sprint review - duración del

Sprint retrospective)= **230,5 horas ideales** (esta es la capacidad que se va a considerar para armar el *Release Plan*)

- Se considera que se parte de un *Sprint 0* en el cual se ejecutaron las tareas necesarias para empezar a trabajar con el MVP con una incertidumbre muy baja.
- Considerando la capacidad del equipo, se decidió implementar un solo **sprint**

**Tareas Sprint 0:**

- Diseño de prototipos y modelos
- Configuración del entorno de trabajo
- Diseño de la BD
- Generación de los ambientes de Desarrollo, Testing, Pre-Producción y Producción
- Spike técnica para integrar la API de Google Maps
- Spike técnica para integrar el Geoposicionamiento
- Desarrollar pruebas de conceptos para consumir la API de Google Maps

## Release y Sprint Planning

Minuta Sprint Planning	
<p><b>Sprint N°: 1</b> <b>Duración del Sprint:</b> 10 días <b>Objetivo del Sprint:</b> Implementar la funcionalidad que le permita al cliente solicitar el servicio de taxi, buscando alguno que se encuentre libre y cerca de la zona. Además se ofrece la funcionalidad de generar notificaciones cuando el cliente solicita el servicio del taxi. Se contempla el inicio de sesión de los taxistas y la funcionalidad que le permite al taxista ocupar, liberar el taxi y ver la ubicación del pasajero.</p> <p><b>Capacidad del equipo en horas ideales:</b> 230,5 horas = 230:30 horas</p> <p><b>Scrum Team:</b></p> <ul style="list-style-type: none"><li>● Acevedo Hernán</li><li>● Boné Nicolás</li><li>● Pereira Duarte Martin</li><li>● Molina Matias</li><li>● Bardin Tomás</li><li>● Aresu Barella Bruno</li><li>● D'uva Micaela</li><li>● Frattin Juan</li></ul>	
Definition Of Done	Sprint Backlog
<ul style="list-style-type: none"><li>● Diseño del prototipo</li><li>● Diseño revisado y aprobado por el PO</li><li>● Código:<ul style="list-style-type: none"><li>○ Código refactorizado</li><li>○ Código con formato estándar</li><li>○ Código comentado</li><li>○ Código en el repositorio</li></ul></li><li>● Pruebas:<ul style="list-style-type: none"><li>○ Test Unitarios</li><li>○ Test de Integración</li><li>○ Test de Aceptación</li></ul></li><li>● Sin defectos conocidos</li><li>● Documentación generada</li></ul>	<ul style="list-style-type: none"><li>● Buscar taxis cercanos</li><li>● Pedir taxi</li><li>● Notificar a taxista solicitud de taxi</li><li>● Loguear taxista</li><li>● Ocupar taxi</li><li>● Liberar taxi</li><li>● Ver ubicación del pasajero</li></ul>

## Sprint Backlog

Identificador de User Story	Tareas	Horas de la tarea	Story Points	Total Horas de la User
<b>Loguear taxista</b>	Desarrollar prototipo para loguear taxista	2	2	23
	Desarrollar la frontend para loguear al taxista	4		
	Desarrollar el backend para loguear al taxista	4		
	Realizar tests unitarios	3		
	Refactorizar y comentar código	2		
	Revisar código	1		
	Pushear código al repositorio	1		
	Realizar los tests de integración	2		
	Realizar los tests de aceptación	2		
	Generar documentación	2		
<b>Ver ubicación del pasajero</b>	Diseñar prototipo	3	5	28
	Diseñar frontend	5		
	Desarrollar backend para la comunicación con la geoposición	5		
	Revisar, refactorizar, comentar, y pushear código	4		
	Realizar tests unitarios	3		
	Realizar tests de integración	3		
	Realizar tests de aceptación	3		
	Generar documentación	2		
<b>Pedir taxi</b>	Diseñar prototipo para la confirmación del cambio de estado	4	5	49
	Desarrollar servicio que se comunica con el sistema de geoposicionamiento	5		

	Generar mapa y consumir la API de Google Maps	10		
	Desarrollar frontend	6		
	Desarrollar backend para la lógica del pedido del taxi	6		
	Revisar, refactorizar, comentar, y pushear código	5		
	Realizar tests unitarios	2		
	Realizar tests de integración	4		
	Realizar tests de aceptación	4		
	Generar documentación	3		
<b>Ocupar taxi</b>	Diseñar prototipo para la confirmación del cambio de estado	2	2	18
	Desarrollar frontend	4		
	Desarrollar backend para cambiar el estado del taxi	4		
	Revisar, refactorizar, comentar, y pushear código	3		
	Realizar tests unitarios	1		
	Realizar tests de integración	1		
	Realizar tests de aceptación	1		
	Generar documentación	2		
<b>Liberar taxi</b>	Diseñar prototipo para la confirmación del cambio de estado	2	2	13
	Desarrollar backend con la lógica del cambio de estado	2		
	Desarrollar la lógica del frontend	2		
	Revisar, refactorizar, comentar, y pushear código	2		
	Realizar tests unitarios	1		
	Realizar tests de integración	1		

	Realizar tests de aceptación	1		
	Generar documentación	2		
<b>Notificar a taxista solicitud de taxi</b>	Desarrollar el prototipo de las notificaciones	3	3	30
	Desarrollar backend con la lógica para gestionar notificaciones PUSH	6		
	Desarrollar la lógica del frontend	6		
	Revisar, refactorizar, comentar, y pushear código	4		
	Realizar tests unitarios	3		
	Realizar tests de integración	3		
	Realizar tests de aceptación	3		
	Generar documentación	2		
<b>Buscar taxis cercanos</b>	Diseñar prototipo de interfaz	3	3	34
	Desarrollar frontend que consuma la API de Google Maps	6		
	Desarrollar backend para buscar taxis cercanos según filtros	7		
	Revisar, refactorizar, comentar, y pushear código	4		
	Realizar tests unitarios	4		
	Realizar tests de integración	4		
	Realizar tests de aceptación	4		
	Generar documentación	2		
<b>Total</b>			<b>22 SP</b>	<b>195 Horas</b>

Si bien definimos que existen 230,5 horas destinadas a cada iteración, en este primer sprint se consumen un total de 195 horas. Las 35,5 horas no planificadas (que serían 4,43 horas por cada integrante aproximadamente) nos permiten definir un margen de horas que puede ser utilizado para cubrir cualquier imprevisto que no haya sido considerado ya sea por inexperiencia del equipo o cualquier inconveniente técnico.