

Universidad Tecnológica Nacional
Facultad Regional Córdoba
Ingeniería en Sistemas de Información



Ingeniería de Software

Trabajo Práctico N° 8
“SCRUM - Release & Sprint Planning”

Curso 4K1 - Grupo N°8 :

94.093	Nielsen Pellinacci, Juan Manuel
74.056	Lencina, Agustín
42.806	Guanuco, Alejandro
68.719	Albarracin, Gonzalo Nahuel
75.982	Correa, Ali



ÍNDICE

ÍNDICE	2
MVP (Producto Mínimo Viable)	3
DESARROLLO	4
Condiciones de contexto necesarias para la planificación del release	4
Plan de Release	4
Minuta del Sprint Planning	5
Definición de hecho (DoD)	5
Sprint Backlog	6
Sprint 1	6
BIBLIOGRAFÍA	7



MVP (Producto Mínimo Viable)

Objetivo:

- Permitir que los clientes (pasajeros) puedan trasladarse al lugar que desean solicitando un taxi cercano.
- Permitir a los taxistas ver la ubicación del pasajero que solicita un viaje para llevarlo.
- Permitir a los taxistas manejar el estado del taxi.

Criterio:

Las user stories incluidas permiten validar la idea de negocio en el mercado, focalizando en la funcionalidad que pone en contacto a Pasajeros con Taxistas. Si bien los taxistas deben poder iniciar sesión para vincularlos con los datos de su vehículo, el análogo para el pasajero no es requisito para poder realizar un pedido. La funcionalidad relacionada con la gestión de taxis por parte de la central no aporta el valor significativo para la validación del mercado por lo que se dejará para futuras iteraciones.

Frases Verbales de las historias de usuario incluidas en el MVP:

Taxista

- Loguear taxista
- Ocupar taxi
- Liberar taxi
- Ver ubicación del pasajero

Pasajero

- Buscar taxis cercanos
- Pedir taxi
- Notificar a taxista solicitud de taxi



DESARROLLO

Condiciones de contexto necesarias para la planificación del release

- **El equipo**
 - Cuenta con 5 miembros con las mismas aptitudes y capacidades técnicas.
 - Actualmente cursan materias de 4° año de la carrera de Ingeniería en Sistemas de Información, por lo que cuentan con conocimientos de programación en distintos lenguajes (como Python, C# y Java), manejo de base de datos, análisis y diseño de sistemas, entre otros.
 - Solo un miembro guía al equipo con sus experiencias en proyectos que usan Metodologías Ágiles, debido a que actualmente participa en un proyecto que utiliza Scrum.
 - El resto del equipo, no cuentan con suficiente experiencia en proyectos de desarrollo con metodologías ágiles.
- **Capacidad**
 - Cada integrante del equipo puede emplear en la jornada laboral de entre 3-7 hs
 - La disponibilidad horaria es diferente entre los miembros debido a que las condiciones laborales y de estudio son variadas en el equipo.
 - Capacidad del equipo en un Sprint de 3 semanas: 345 hs
 - Capacidad de cada miembro en un Sprint: 45-105 hs

Plan de Release

De acuerdo a nuestra capacidad como equipo, serán necesarios dos Sprints en los que seleccionamos las User Stories para que, en cada entrega, tengamos funcionalidades listas para ser probadas.

(En el sprint 0 se realizan las tareas correspondientes a creaciones de arquitectura, repositorios, reglas de versionado y definiciones necesarias para poder empezar el desarrollo de las User Stories)

1. SPRINT N°1 - Duración: 3 semanas

- a. Loguear taxista (2)
- b. Buscar taxis cercanos (3)
- c. Pedir taxi (5)
- d. Notificar taxista (3)

2. SPRINT N°2 - Duración: 3 semanas

- a. Ocupar taxi (2)
- b. Liberar taxi (2)
- c. Ver ubicación del pasajero (5)

Duración estimada del Release: 6 semanas (2 Sprints)

Supuestos: Estimamos que luego de estos 2 Sprints tendremos implementadas las User Stories pertenecientes al MVP, por lo tanto podremos contar con el primer release de producto para poder salir a validar su aceptación, funcionamiento y modelo de negocio en el mercado.



Sprint Planning

Minuta del Sprint Planning

Sprint Nro. 1

Duración del Sprint en días: 6

Objetivo del Sprint: Lograr crear una primera impresión del producto en la que se permite loguear a los usuarios en la aplicación y, además, puedan buscar y pedir un taxi cercano.

Scrum Master

- **ALBARRACÍN, Gonzalo Nahuel - 4 hs**

Scrum Team:

- **NIELSEN PELLINACCI, Juan Manuel (Full Stack) - 7 hs**
- **LENCINA, Agustín (Frontend) - 5 hs**
- **GUANUCO, Alejandro (Testing) - 3 hs**
- **CORREA, Ali (Base de datos) - 4 hs**

Capacidad del Equipo en Horas Ideales: 345 hs, con jornadas laborales que están el rango de 3 a 7 hs.

Definición de hecho (DoD)

- ☐ Código completo en repositorio
- ☐ Código revisado
- ☐ Establecer conexiones con las API's necesarias
- ☐ Pruebas de unidad aprobadas
- ☐ Pruebas de integración aprobadas
- ☐ Requisitos funcionales cumplidos
- ☐ Requisitos no-funcionales cumplidos
- ☐ Sin defectos encontrados
- ☐ Revisión y Aceptación del Product-Owner



Sprint Backlog

Sprint 1

US	Tareas	Esfuerzo estimado
1. Pedir taxi (5)	<ul style="list-style-type: none">→ Codificación de la interfaz para el pedido (20 hs)→ Unión de frontend y backend (15 hs)→ Obtención de georeferenciamiento online del pasajero y seguimiento en tiempo real del taxi seleccionado (30 hs)→ Revisión de Código (10 hs)→ Pruebas de conexión (14 hs)→ Automatización de pruebas (16 hs)	105 hs
2. Buscar taxis cercanos (3)	<ul style="list-style-type: none">→ Conectar con API de Google Maps (15 hs)→ Conectar con Modulo de GPS del celular (10 hs)→ Pruebas de conexión (10 hs)→ Unir frontend y backend (12 hs)→ Codificación de interfaz para visualización de Maps (8 hs)→ Codificación de interfaz para la búsqueda (10 hs)→ Revisión de Código (8 hs)→ Realización de pruebas de usuario (8hs)	81 hs
3. Notificar taxista (3)	<ul style="list-style-type: none">→ Codificación de la interfaz de notificación push (8 hs)→ Crear Tabla de notificaciones en Base de Datos (4 hs)→ Codificación del mensaje push (7 hs)→ Unión de frontend y backend (10 hs)→ Pruebas de conexión (8 hs)→ Implementar notificaciones Push (10 hs)→ Automatización de pruebas (8 hs)→ Revisión de Código (10 hs)	65 hs
4. Loguear taxista (2)	<ul style="list-style-type: none">→ Codificación de la interfaz de logueo (8 hs)→ Crear tablas en Base de Datos para registro del taxista (8 hs)→ Conectar con API de Facebook para loguearse (8 hs)→ Codificación validaciones (6 hs)→ Unión de frontend y backend (9 hs)→ Automatización de pruebas (8 hs)→ Revisión de Código (8 hs)	55 hs
13 Story Points		306 hs



BIBLIOGRAFÍA

- ❖ The Scrum Guide
<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
- ❖ Guía Scrum 2020 y métricas ágiles (brindada por la cátedra)