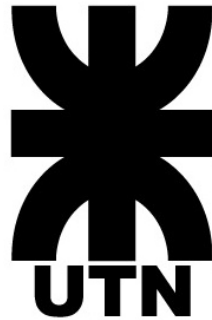


UNIVERSIDAD TECNOLÓGICA NACIONAL



FACULTAD REGIONAL CÓRDOBA

INGENIERÍA EN SISTEMAS DE INFORMACIÓN

Ingeniería y Calidad de Software

“Planificación de Release y de Sprint”

Curso: 4K3

Grupo 4:

- 57572, Amado, Marco
- 78751, Corvalán, Martín Pablo
- 68160, Guevara, Denis
- 71656, Lozada, Mauro
- 80228, Ramos, Mauricio
- 79811, Vargas, Bruno

Docentes:

- Ing. Laura Covaro
- Ing. Cecilia Massano
- Pilar Avila

Fecha de presentación: 11/10/23

Índice:

Índice:	2
Contexto:	3
Capacidad del equipo:	3
Product Backlog:	3
User Stories y división en tareas:	4
Plan de Release:	7
Sprint Planning:	8
Sprint Backlog:.....	10

Contexto:

El equipo de trabajo cuenta con conocimientos de desarrollo web y mobile tanto en back-end como front-end y en tecnologías como Angular, Django, C# y Kotlin.

El sprint definido por el equipo tendrá una duración de dos semanas, en donde se trabajarán 10 días por sprint ya que se cuentan únicamente los días laborales de la semana sin tener en cuenta sábado y domingo.

Capacidad del equipo:

Integrante	Días disponibles	Días para otras actividades Scrum	Horas por día	Horas para esfuerzo
Marco	7	1	3	18
Martin	8	2	5	30
Denis	8	2	4	24
Mauro	7	1	5	30
Mauricio	6	1	5	25
Bruno	7	1	5	30
Total				157

La capacidad total del equipo por sprint es de 157 horas. El cálculo de las horas se realizó teniendo en cuenta que los miembros del equipo tienen otras actividades durante el día.

Product Backlog:

Product Backlog			
Identificación	US	Story Point	MVP
1	Pedir taxi	5	SI
2	Buscar taxis cercanos	3	SI
3	Ver ubicación del pasajero	5	SI

4	Loguear taxista	2	SI
5	Ocupar taxi	2	SI
6	Liberar taxi	2	SI
7	Notificar a taxista y a central pedido de taxi	3	SI
8	Ver mapa de taxis	5	NO
9	Loguear pasajero	2	NO
10	Marcar taxi como fuera de servicio	2	NO
11	Registrar central de taxis	1	NO

User Stories y división en tareas:

Pedir Taxi	
Tareas	Horas ideales
Investigar sobre la API de Google Maps para geoposicionamiento	8
Diseñar la interfaz de usuario	7
Implementar front-end	7
Generar mapa usando librería gráfica	5
Implementar el back-end e integrar API de Google Maps para obtener datos de geoposicionamiento	10
Automatizar pruebas	6
Realizar revision de codigo	1
Realizar documentación	2
Total de horas ideales	46

Buscar taxis cercanos	
Tareas	Horas ideales
Investigar sobre geoposicionamiento	6
Diseñar interfaz de usuario	4
Implementar el front-end	5
Generar mapa	5
Implementar el back-end e integrar API de Google Maps para obtener datos de geoposicionamiento	6
Realizar pruebas con el ambiente simulado	6
Realizar revision de codigo	2
Realizar documentación	2
Total de horas ideales	36

Ver ubicación del pasajero	
Tareas	Horas ideales
Diseño de interfaz de usuario	5
Implementar el front-end	6
Implementar el back-end e integrar API de Google Maps para obtener datos de geoposicionamiento	8
Implementar visualización en mapa	6
Realizar pruebas con el ambiente simulado	6
Realizar revision de codigo	3
Realizar documentación	4
Total de horas ideales	38

Loguear Taxista	
Tareas	Horas ideales
Investigar sobre la API de Facebook	4
Diseñar interfaz de usuario	3
Implementar front-end	3
Implementar el back-end e integrar con API de Facebook	4
Realizar pruebas	2
Realizar revision de codigo	1
Realizar documentación	2
Total de horas ideales	19

Ocupar Taxi	
Tareas	Horas ideales
Implementar front-end	2
Implementar el back-end	2
Realizar pruebas	3
Realizar revision de codigo	1
Realizar documentación	1
Total de horas ideales	9

Liberar Taxi	
Tareas	Horas ideales
Implementar front-end	2
Implementar el back-end	2
Realizar pruebas	3
Realizar revision de codigo	1
Realizar documentación	1
Total de horas ideales	9

Notificar a taxista y a central pedido taxi	
Tareas	Horas ideales
Investigar sobre envío de notificaciones push	8
Diseñar el formato de la notificación push	7
Implementar las notificaciones para los dispositivos	8
Implementar el back-end	3
Realizar pruebas	5
Realizar revision de codigo	3
Realizar documentación	3
Total de horas ideales	37

Plan de Release:

El release tendrá lugar en dos Sprints, cada sprint tendrá una duración de 10 días de trabajo.

A continuación se indican las características que se incluyen en cada sprint:

Sprint 1			
Prioridad	US	Story Points	Horas ideales
1	Pedir taxi	5	46
2	Buscar taxis cercano	3	36
3	Ver ubicación de pasajero	5	38
4	Loguear taxista	2	19
Total		15	139

Sprint 2			
Prioridad	US	Story Points	Horas ideales
5	Ocupar taxi	2	9
6	Liberar Taxi	2	9
7	Notificar a taxista y a central pedido taxi	3	37
Total		7	55

Sprint Planning:

Sprint Planning	
<p>Sprint Nro. 1</p> <p>Duración del Sprint en días: 10 días</p> <p>Objetivo del Sprint: El objetivo de este sprint es construir funcionalidades que permitan simular el pedido de un viaje</p> <p>Equipo Scrum:</p> <ul style="list-style-type: none"> • Amado, Marco • Corvalán, Martin Pablo • Guevara, Denis • Lozada, Mauro • Ramos, Mauricio • Vargas, Bruno <p>Capacidad del equipo en horas ideales: 157</p>	
Definición de Hecho para el equipo	Sprint Backlog
<ul style="list-style-type: none"> • Código Completo: <ul style="list-style-type: none"> ○ Código Refactorizado. ○ Código con formato que respete la Guía de estilo definida. ○ Código Comentado. ○ Código en el repositorio. ○ Código Inspeccionado. • Documentación de Usuario actualizada. • Probado: <ul style="list-style-type: none"> ○ Prueba de unidad hecha. ○ Prueba de interacción 	<ul style="list-style-type: none"> • Pedir taxi. • Buscar taxis cercanos. • Ver ubicación de pasajero. • Loguear taxista. <p>Se adjunta el sprint backlog en la página 10</p>

<p>hecha.</p> <ul style="list-style-type: none"> ○ Pruebas de seguridad pasadas. ○ Prueba de regresión hecha. ○ Plataforma probada o Lenguaje probado. ○ Prueba en dispositivos/navegadores pasada. <ul style="list-style-type: none"> ● Cero defectos conocidos. ● Prueba de aceptación realizada. 	
---	--

Sprint Backlog:

Sprint 1					
Identificación	US	Story Points	Tareas	Horas ideales por Tarea	Horas ideales de la US
1	Pedir taxi	5	Investigar sobre la API de Google Maps para geoposicionamiento	8	46
			Diseñar la interfaz de usuario	7	
			Implementar front-end	7	
			Generar mapa usando librería gráfica	5	
			Implementar el back-end e integrar API de Google Maps para obtener datos de geoposicionamiento	10	
			Automatizar pruebas	6	
			Realizar revision de codigo	1	
2	Buscar taxis cercanos	3	Investigar sobre geoposicionamiento	6	36
			Diseñar interfaz de usuario	4	
			Implementar el front-end	5	
			Generar mapa	5	
			Implementar el back-end e integrar API de Google Maps para obtener datos de	6	

			geoposicionamiento		
			Realizar pruebas con el ambiente simulado	6	
			Realizar revision de codigo	2	
			Realizar documentación	2	
3	Ver ubicación del pasajero	5	Diseño de interfaz de usuario	5	38
			Implementar el front-end	6	
			Implementar el back-end e integrar API de Google Maps para obtener datos de geoposicionamiento	8	
			Implementar visualización en mapa	6	
			Realizar pruebas con el ambiente simulado	6	
			Realizar revision de codigo	3	
			Realizar documentación	4	
4	Loguear taxista	2	Investigar sobre la API de Facebook	4	19
			Diseñar interfaz de usuario	3	
			Implementar front-end	3	
			Implementar el back-end e integrar con API de Facebook	4	

			Realizar pruebas	2	
			Realizar revision de codigo	1	
			Realizar documentación	2	
Total de Story Points		15	Total de Horas Ideales		139