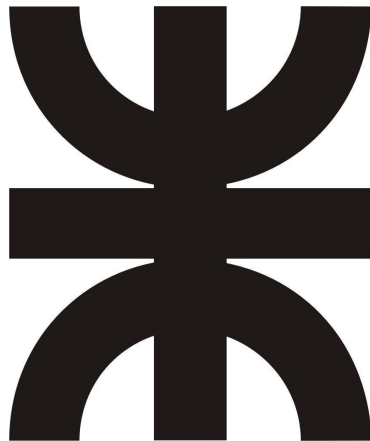


UNIVERSIDAD TECNOLÓGICA NACIONAL

FACULTAD REGIONAL CÓRDOBA



Departamento de Ingeniería en Sistemas de Información

Cátedra: Ingeniería y calidad de software

Trabajo práctico N° 4

“SCM - Herramientas de SCM”

Curso: 4K1

Docentes:

- Meles, Judith
- Crespo, Mickaela
- Boiero Rovera, Gerardo Javier

Grupo: 7

Integrantes:

- Battig Chavez Joaquin Antonio, 76056
- Falco Gonzalo Pedro, 80023
- Gonzalez Valentina, 78499
- Manzolido Evelyn, 76559
- Rojas Tobias, 407920

Fecha límite de entrega: 10/10/2023

ÍNDICE

Consideraciones Iniciales	2
Definición de Release	2
Plan de Release	2
Estimaciones	3
Capacidad de equipo	7
Minuta para Sprint Planning	8
Definición de hecho	9

Consideraciones Iniciales

Respecto a la duración de cada sprint, debido a que no existe en el equipo un historial de trabajo y la duración estimada en la que se hacen las tareas, además de la pocos conocimientos sobre algunas tecnologías utilizadas para desarrollar el MVP, se decidió como plan realizar iteraciones de 2 semanas de duración, pero solo se consideran los días hábiles.

Por Sprint se contará con las siguientes ceremonias propuestas por el Framework Scrum: 15 minutos de Daily x 10 días, 2 horas de Sprint Planning, 2 horas de Sprint Review y 1.5 horas de Retrospective.

4.1

Definición de Release

En el primer release se desarrollará el MVP. hemos decidido realizar 2 Sprints que tendrán una duración de 10 días hábiles cada uno respectivamente, con el objetivo de entregar valor al cliente en cada iteración al finalizar el sprint y para luego poder obtener una retroalimentación y poder aprender y validar que lo que venimos haciendo, es lo que el cliente espera; y si algo no es lo esperado, nos servirá para poder adaptarnos a esos cambios que pudieran surgir lo más rápido posible y evitar pérdidas de tiempo y dinero.

Al finalizar el sprint 1 ya tendríamos desarrollado el producto de software con lo mínimo necesario para poder validar la idea de negocio del mercado.

Luego, vendrá el segundo sprint el cual irá agregando nuevas funcionalidades que no fueron consideradas en el MVP (salvo Notificar a taxista solicitud de taxi y Buscar taxi cercano, las cuales todavía no fueron implementadas y forman parte del MVP).

4.2

Plan de Release

User Stories	Story Points	Sprint
Loguear Taxista	2	1
Ver ubicación del pasajero	5	
Liberar taxi	2	
Ocupar taxi	2	
TOTAL SP	11	
Pedir taxi	5	2
Buscar taxis cercanos	3	
Notificar a taxista la solicitud de taxi	3	
TOTAL SP	11	


El MVP cuenta en total con 22 Story Points. Como equipo, se estima que la capacidad es de 11 Story Points para el primer Sprint, y 11 Story Points para el segundo sprint. De esta forma el Release de este mínimo producto viable se podrá realizar en 2 Sprints (Velocidad).



ESTIMACIONES

- Estimación de horas por personas: 6hs
- Estimación de horas disponibles por iteración: 240 → 6hs * 5 integrantes * 8 días por iteración.

Sprint Backlog				
User Story	Story Points	Tarea	Estimación (hs)	Hs totales
Loguear Taxista	2	Codificar Interfaz de usuario loguearse taxi	4hs	25hs
		Desarrollar Frontend Javascript	4hs	
		Desarrollar Backend para logueo taxista	1hs	
		Realizar pruebas de logueo	2hs	
		Refactorizar código	4hs	
		Realizar pruebas de usuario	3hs	
		Realizar pruebas de regresión	2hs	
		Realizar “push” de código de repositorio	2hs	
		Generar documentación de la US	3hs	
Pedir taxi		Acordar el estilo de interfaz con el PO	5hs	
		Desarrollar módulo de comunicación de backend con sistema de geoposicionamiento	6hs	
		Solicitar	4hs	

	5	credenciales y desarrollar el módulo que hará la integración con Google maps		53hs
		Diseñar interfaz	4hs	
		Desarrollar código frontend	4hs	
		Desarrollar código backend	5hs	
		Realizar pruebas de regresión	4hs	
		Realizar pruebas unitarias	3hs	
		Refactorizar código	4hs	
		Realizar pruebas de usuario	3hs	
		Diseñar y ejecutar casos de prueba	4hs	
		Realizar push de código al repositorio	4hs	
		Generar la documentación de la US	3hs	
				
Liberar taxi	2	Diseñar interfaz gráfica	2.5hs	23hs
		Desarrollar código para capa cliente	4hs	
		Desarrollar código de lógica para cambio de estado de taxi	4hs	
		Revision de codigo	2.5hs	
		Realizar	3hs	

		pruebas unitarias		
		Realizar pruebas de usuario	3hs	
		Realizar push de código en repositorio	2hs	
		Generar documentación	2hs	
Ver ubicación del pasajero	5	Diseñar interfaz gráfica	7hs	
		Desarrollar código para cada lado cliente	6hs	
		Desarrollar backend de comunicación y sincronización de ubicación	6hs	
		Refactorizar código	3hs	
		Realizar pruebas unitarias	2hs	
		Realizar pruebas de regresión	3hs	
		Realizar pruebas de usuario	4hs	
		Generar documentación	4hs	
		Revision de codigo	5hs	
		Realizar push de código al repositorio	2hs	
				44hs
Ocupar taxi		Diseñar interfaz gráfica	3hs	
		Desarrollar código para capa lado	4hs	

	2	cliente		22hs
		Desarrollar código de lógica para cambio de estado de taxi	4hs	
		Revisión de código	3hs	
		Realizar pruebas unitarias	3hs	
		Realizar pruebas de usuario	2hs	
		Realizar push de código en repositorio	1hs	
		Generar documentación	2hs	
Hs Totales				167hs

8.1

CAPACIDAD DEL EQUIPO

MIEMBROS DEL EQUIPO	DÍAS DISPONIBLES	HS EN EVENTOS SCRUM	HORAS POR DÍA	HORAS TOTALES	HORAS DE ESFUERZO DISPONIBLE
Battig Joaquin	10	8	6	60	52
Falco Gonzalo	10	8	6	60	52
Gonzalez Valentina	10	8	6	60	52
Manzolido Evelyn	10	8	6	60	52
Rojas Tobias	10	8	6	60	52
TOTAL HS	-	40	-	300	260

MINUTA PARA SPRINT PLANNING

Minuta para Sprint Planning	
<p>Nro Sprint: 1</p> <p>Duración de Sprint (Días): 10 días</p> <p>Objetivo del Sprint: Construir un incremento que permita a un taxista loguearse, ver la ubicación del pasajero, liberar y ocupar el taxi.</p> <p>Equipo Scrum:</p> <ul style="list-style-type: none">• Battig Joaquin• Falco Gonzalo• Gonzalez Valentina• Manzolido Evelyn• Rojas Tobias <p>Capacidad del equipo (En HS ideales): 260 hs</p>	

10.1

DEFINICIÓN DE HECHO



Definición de Hecho (DONE)	
<input type="checkbox"/>	Todos los criterios de aceptación implementados y probados
<input type="checkbox"/>	Tareas de desarrollo completas
<input type="checkbox"/>	Código Refactorizado
<input type="checkbox"/>	Código con formato estándar
<input type="checkbox"/>	Código Comentado
<input type="checkbox"/>	Código en el repositorio
<input type="checkbox"/>	Código Inspeccionado
<input type="checkbox"/>	Código desplegado en ambiente
<input type="checkbox"/>	Maquetado que cumple con lineamientos
<input type="checkbox"/>	Funcionalidad validada
<input type="checkbox"/>	Integración con servicio correcto
<input type="checkbox"/>	Tareas de testing completas
<input type="checkbox"/>	Casos de prueba diseñados y revisados
<input type="checkbox"/>	Ciclos de prueba ejecutados y superados completos
<input type="checkbox"/>	Defectos validados
<input type="checkbox"/>	Casos de prueba automatizados
<input type="checkbox"/>	Etapas de testing superadas
<input type="checkbox"/>	Prueba de integración realizada y superada
<input type="checkbox"/>	Prueba de regresión realizada y superada
<input type="checkbox"/>	Bugs gestionados
<input type="checkbox"/>	Documentados y priorizados correctamente
<input type="checkbox"/>	Validados por el equipo
<input type="checkbox"/>	Documentación actualizada
<input type="checkbox"/>	Análisis funcional completo y validado por el Product Owner
<input type="checkbox"/>	User Stories completas y validadas por el Product Owner
<input type="checkbox"/>	Aprobado por el Product Owner

Index of comments

- 4.1 Los días de Planning y REview no se realiza la ceremonia de daily.
- 4.2 El objetivo eran realizar las users definidas en el MVP dado por el PO en la consigna, no la otras.
- 5.1 Validar este tiempo
- 8.1 Que hacen con las 100hs restantes? Son aplicadas a las ceremonias? JUSTIFICAR
- 10.1 No considera las fechas de inicio y fin, y el impacto en las horas ideales