

Estructura de las Entregas Semanales con Seguimiento a Scrum:

1. Objetivos del Sprint

- Historia de usuario 3: Como desarrollador de la aplicación queiro simular el funcionamiento de ocupado / vacío de mi aplicativo para que me sirva de ayuda hasta que se pueda hacer la prueba con el dispositivo real.

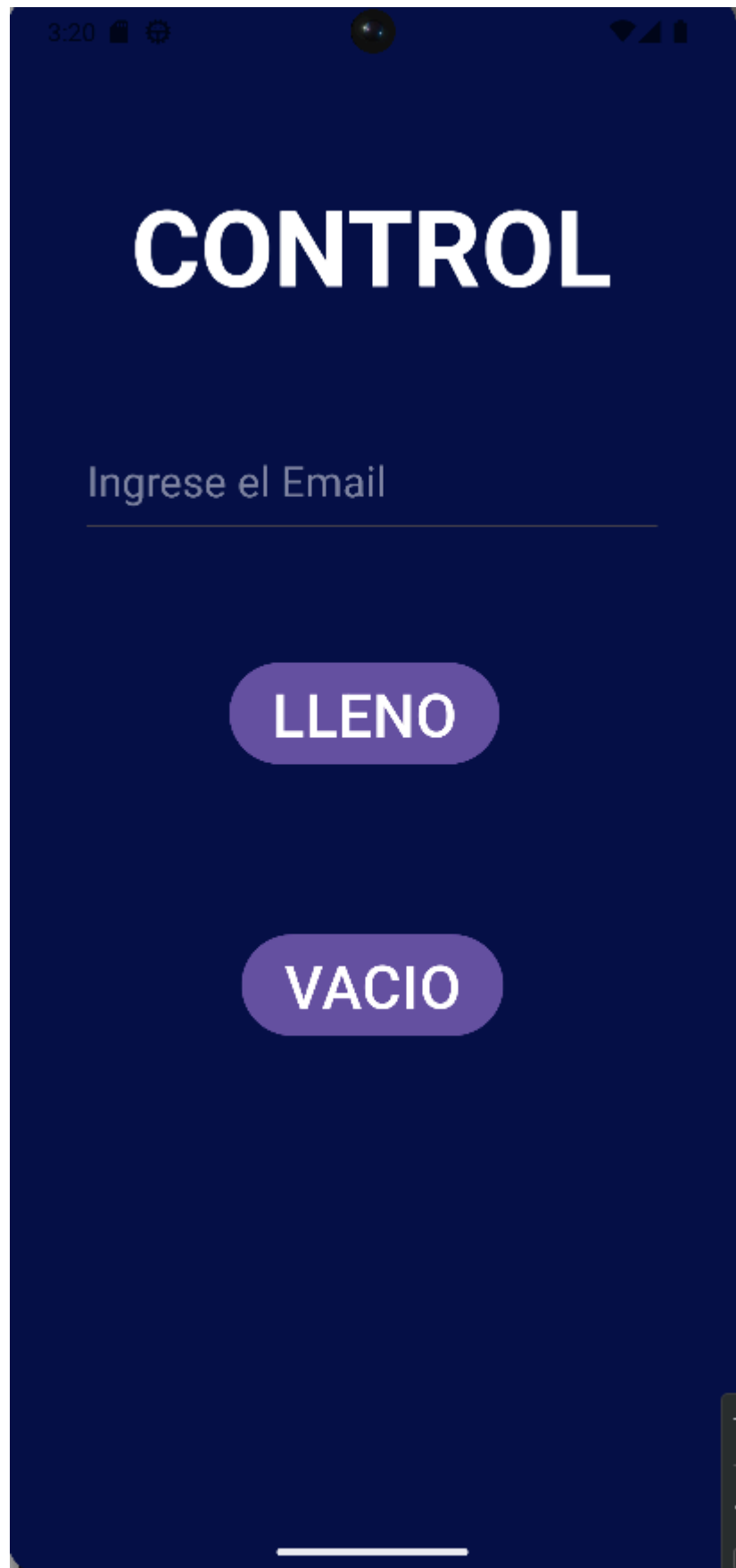
2. Seguimiento al Sprint (Scrum Events)

- **Daily Scrum:** Se realizaron las reuniones diarias, donde se vio dificultades para que dos aplicaciones diferentes puedan trabajar con la misma base de datos de firebase.
- **Sprint Planning:** Se decidió culminar la historia de usuario número 1 enfocada en el desarrollo de una interfaz intuitiva para que el usuario se encuentre más cómodo al utilizar la aplicación móvil y realizar la historia de usuario número 2 enfocada en que la aplicación sea agradable a la vista además de trabajar en la historia de usuario número 3 se probará el estado del estacionamiento.
- **Sprint Review:** Se pudo lograr la modificación de la base de datos desde otra aplicación.
- **Sprint Retrospective:** El equipo buscar la manera de experimentar con nuevas tecnologías desconocidas por el equipo para implementarlas al proyecto, como el chat bot.

3. Descripción de Funcionalidades Implementadas

- **Aplicación controladora:** Se creará una aplicación que permita modificar el estado de la aplicación para que sea entrada y salida de un espacio de estacionamiento.

4. Demostración de Funcionalidades



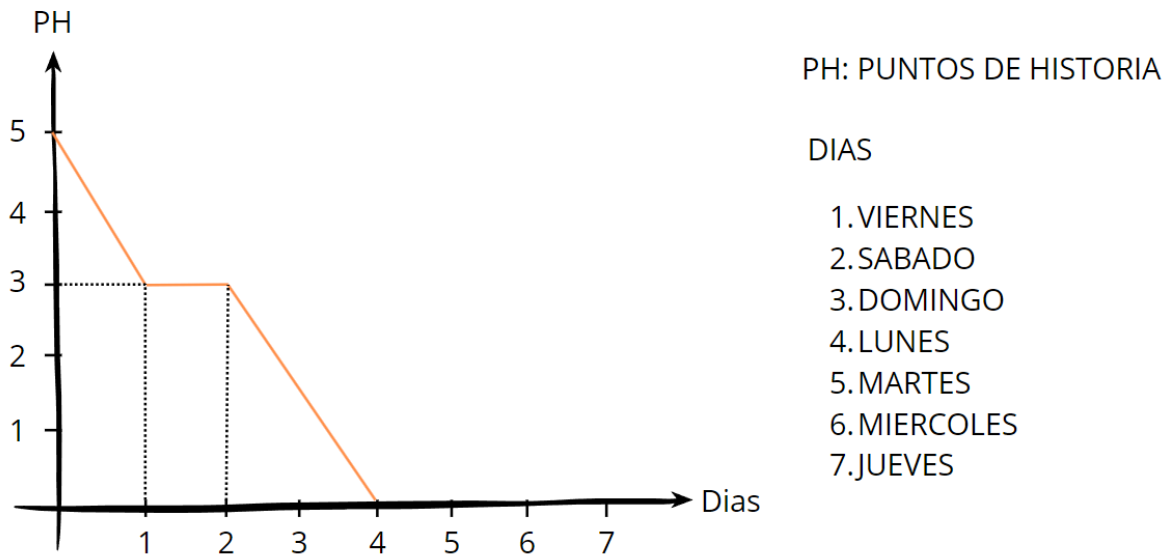
5. Pruebas Realizadas

Identificador de casos de prueba:		PR001	
Objetos a probar:		Botón Lleno – botón Vacío	
Dependencias:		Datos Garaje	
Ambiente		Computador personal	
Requisitos de procedimiento especiales:		La persona tiene que colocar el Email del usuario a modificar	
Descripción:		Almacenar datos en <u>firebase</u>	
Secuencia Normal - Pasos		Resultado Esperado	Resultado Obtenido
1	Se ingresa datos en email del estacionamiento a cambiar	Se acepta el email enviado	Se acepta el email enviado
2	Hacer <u>click</u> el botón Lleno	Se envía modifica la base de datos <u>Firestore</u>	Se envía modifica la base de datos <u>Firestore</u>
3	Hacer <u>click</u> el botón Vacío	Se envía modifica la base de datos <u>Firestore</u>	Se envía modifica la base de datos <u>Firestore</u>
Excepciones - Pasos			
2	Si ingresa un email no reconocido se enviará mensaje de para la corrección al presionar botón Lleno		
3	Si ingresa un email no reconocido se enviará mensaje de para la corrección al presionar botón Vacío		
Fecha de prueba		7/10/2024	Nombre del Probador
Resultado final		Aprobado	Tipo de Prueba
Prueba Alfa / Beta		N/A	Cantidad de Intentos de Confirmación
			Edward <u>Raul Llancay</u>
			Confirmación
			1



6. Seguimiento al Burndown Chart

Burndown Chart



7. Código Fuente

- <https://github.com/Patrick4235/aplicacion-movil-estacionamiento>

8. Avance respecto al Plan del Proyecto

- **Avance planificado:** Se planeó implementar una aplicativo que modifique el estado de ocupado / vacío de un estacionamiento.
- **Avance Real:** Se pudo realizar el aplicativo que permite cambiar el estado del garaje.
- **Discrepancia:** Hubo problemas en la verificación del estado por temas de coordinación de tiempo.

9. Desafíos y Soluciones

- **Desafío:** Se presentó una falla sin notificación que impide la ejecución del aplicativo
- **Solución:** Recreación del aplicativo

10. Plan de Acción para el Próximo Sprint

- **Historia de usuario 4:** Como Conductor quiero ver en tiempo real la disponibilidad de espacios para poder saber a que garage tengo que ir

11. Documentación Actualizada

- historia de usuario

COMO:	Conductor
QUIERO:	Ver en tiempo real la disponibilidad de espacios
PARA:	Poder saber a qué garaje tengo que ir
Criterios de Aceptación	
DADO:	Que un garaje este libre o lleno
CUANDO:	Un el conductor revise el mapa
ENTONCES:	El conductor verá un icono de verde si el garaje está disponible y rojo si está lleno.