

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE HABITAT, INFRAESTRUCTURA Y
CREATIVIDAD

CARRERA DE: INGENIERÍA EN SISTEMAS DE INFORMACIÓN



Trabajo de Titulación

Tema: PROTOTIPO DE APLICACIÓN WEB PARA LA RESERVA Y
GESTIÓN DE SALAS DE ENSAYO MUSICALES EN QUITO.

AUTOR:

JONATHAN STEVE ROSERO GARCIA

QUITO DM, JULIO 2025

RESUMEN

Frente a la falta de una plataforma que pueda ofrecer información actualizada sobre las salas de ensayo disponibles dentro del Distrito Metropolitano de Quito, nace la necesidad de desarrollar una solución tecnológica que permita conectar a los músicos de la ciudad con espacios adecuados para la práctica musical. El presente proyecto de titulación tiene como principal objetivo el desarrollo de un prototipo funcional de una aplicación web que permita tanto a músicos, como a propietarios de salas de ensayo en Quito, el poder gestionar reservas de forma segura, eficiente y personalizada.

El prototipo de la aplicación incluye con funcionalidades como autenticación por roles, búsqueda mediante filtros, una visualización interactiva en mapas, sistema de reservas por fecha y hora, además de un apartado de gestión de salas desde un panel administrativo. Para el desarrollo del presente proyecto, se aplicó la metodología de SCRUM, lo que permitió desarrollar y validar cada módulo clave del sistema mediante 4 sprints.

Finalmente, los resultados obtenidos mediante una prueba de usabilidad con usuarios reales evidenciaron una alta aceptación del prototipo como solución a la problemática de la búsqueda y reserva de las salas de ensayo. En consecuencia, se señalaron mejoras tanto en la experiencia de interfaz, como en nuevas funcionalidades aplicables al producto final. Por lo que, se concluyó a la solución propuesta como una respuesta a la necesidad real dentro del contexto cultural musical de Quito, además de que la evolución hacia una aplicación móvil con un enfoque tanto social, como analítico, puede representar un aporte significativo al ecosistema artístico local.

ABSTRACT

Faced with the lack of a platform that can provide updated information on the rehearsal rooms available within the Metropolitan District of Quito, the need to develop a technological solution that connects the musicians of the city with suitable spaces for musical practice arises. The main objective of this degree project is the development of a functional prototype of a web application that allows both musicians and owners of rehearsal rooms in Quito to manage reservations in a secure, efficient and personalized way.

The prototype of the application includes functionalities such as authentication by roles, search by filters, interactive visualization on maps, reservation system by date and time, as well as a room management section from an administrative panel. For the development of this project, the SCRUM methodology was applied, which allowed the development and validation of each key module of the system through 4 sprints

Finally, the results obtained through a usability test with real users showed a high acceptance of the prototype as a solution to the problem of finding and booking rehearsal rooms. Consequently, improvements were noted both in the interface experience and in the new functions applicable to the final product. Therefore, the proposed solution was concluded as a response to a real need within the musical cultural context of Quito, and the evolution towards a mobile application with a social and analytical approach could represent a significant contribution to the local artistic ecosystem.

DEDICATORIA

La presente dedicatoria es hacia mis padres, los cuales me han apoyado incondicionalmente durante toda mi vida académica, como personal. Ellos son la razón por la cual puedo realizar este proyecto, además de trasmitirme ese amor hacia la música.

AGRADECIMIENTO

Agradezco a Dios, por haberme dado la oportunidad de tener unos grandes padres, los cuales me brindaron el apoyo suficiente para poder realizar mi carrera. Además, de agradecer a los más cercanos a mí por siempre apoyarme en cualquier meta que me proponga. A mi tutor y profesores los cuales han sido apoyo para el logro de este trabajo.

ÍNDICE

CAPÍTULO I: INTRODUCCIÓN	1
1.1 JUSTIFICACIÓN	1
1.2 PLANTEAMIENTO DEL PROBLEMA	4
1.3 OBJETIVOS	5
1.3.1 Objetivo General.....	5
1.3.2 Objetivos Específicos.....	5
1.4 ANTECEDENTES	5
1.5 ALCANCE.....	7
CAPÍTULO II: MARCO TEÓRICO Y CONCEPTUAL.....	8
2.1 MARCO CONCEPTUAL	8
2.1.1 Sala de Ensayo/Grabación Musical	8
2.1.2 Prototipo de Aplicación Web.....	9
2.1.3 Experiencia del Usuario (UX) & Diseño de Interfaz de Usuario (UI) mediante conceptos de Interacción Humano Computador (IHC)	10
2.1.4 Usabilidad	12
2.2 METODOLOGIA	12
2.2.1 Ingeniería de requerimientos.....	12
2.2.2 Técnicas para el levantamiento de requerimientos	12
2.2.3 SCRUM.....	13
2.2.4 Implementación de SCRUM, para el desarrollo del prototipo de aplicación web para la reserva y gestión sobre salas de ensayo.	15
2.2.5 Beneficios en cuanto la implementación de SCRUM para el desarrollo del prototipo de aplicación web la reserva y gestión sobre salas de ensayo.	16

2.3 HERRAMIENTAS	17
2.3.1 Flutter.....	17
2.3.2 Supabase	18
2.3.4 Git & GitHub	20
2.3.5 Servicios de APIs.....	21
2.3.4 Firebase Hosting	21
2.1 COMPARATIVA DE LAS HERRAMIENTAS UTILIZADAS	22
2.4.1 Supabase vs Firebase	22
2.4.2 Flutter Web vs React Native.....	24
2.4.3 Visual Studio Code vs Android Studio.....	25
2.4.4 Comparativa de Servicios de Hosting.....	27
2.4.5 Comparativa de Metodologías de Desarrollo	28
CAPÍTULO III: ALZA DE REQUERIMIENTOS.....	30
3.1 ALZA DE REQUERIMIENTOS	30
3.1.1) Primera Entrevista:	30
3.1.2) Segunda Entrevista:	32
3.2 REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES	33
3.2.1 Tabla de los Requerimientos Funcionales y No Funcionales	33
CAPÍTULO IV: DESARROLLO DEL PRIMER PROTOTIPO, MEDIANTE METODOLOGÍA SCRUM.....	36
4.1 DEFINICIÓN DE COMPONENTES CLAVE DE SCRUM	36
4.1.1 Roles de ‘SCRUM definidos:	36
4.1.2 HISTORIAS DE USUARIO.....	36
4.1.3 Product BackLog	42
4.1.4 Definición de Sprints en base al Product Backlog (Sprint Planning).....	47
4.2 SPRINT 1 – Autenticación y Accesos	48

4.2.1	Introducción	48
4.2.2	Objetivos del Sprint 1:	48
4.2.3	Historias de Usuario Asociadas al Sprint 1.....	48
4.2.4	Tareas del Sprint 1:	50
4.2.5	Modelo de Datos para el Sprint 1	51
4.2.6	Actividades Técnicas	51
4.2.7	Diagramas de Flujo Correspondientes al Sprint 1	53
4.2.8	RESULTADOS OBTENIDOS	53
4.2.9	Observaciones y Recomendaciones de los Usuarios (Sprint Review).....	54
4.2.10	Conclusiones del Sprint 1 (Sprint Retrospective).....	54
4.3	SPRINT 2 – Búsqueda, Reserva y Gestión de las Salas de Ensayo	55
4.3.1	Introducción	55
4.3.2	Objetivos del Sprint 2	55
4.3.3	Historias de Usuario Asociadas al Sprint 2.....	55
4.3.4	Tareas del Sprint 2	57
4.3.5	Modelo de Datos para el Sprint 2	59
4.3.6	Actividades Técnicas	61
4.3.7	Diagramas de Flujo del Sprint 2	63
4.3.8	Resultados Obtenidos (Sprint Review).....	65
4.3.9	Observaciones y Recomendaciones (Sprint Review)	66
4.3.10	Conclusión del Sprint (Sprint Restrospective).....	66
4.4	SPRINT 3 – Flujo de Pago, Gestión de Reservas y Apartado Administrativo.....	67
4.4.1	Introducción	67
4.4.2	Objetivos del Sprint 3	67
4.4.3	Historias de Usuario Asociadas al Sprint 3.....	67

4.4.4 Tareas del Sprint 3	68
4.4.5 Modelo de Datos para el Sprint 3	69
4.4.6 Actividades Técnicas	70
4.4.7 Diagrama de Flujo.....	72
4.4.8 Resultados Obtenidos.....	73
4.4.9 Observaciones y Comentarios (Sprint Review).....	73
4.4.10 Conclusiones del Sprint 3 (Sprint Restrospective)	73
4.5 SPRINT 4 – Personalización, Reseñas y Fidelidad.	74
4.5.1 Introducción.....	74
4.5.2 Objetivos del Sprint	74
4.5.3 Historias de Usuario Asociadas al Sprint 4.....	75
4.5.4 Tareas del Sprint 4	76
4.5.5 Modelo de Datos Sprint 4	76
4.5.6 Actividades Técnicas del Sprint 4.....	78
4.5.7 Diagrama de Flujo.....	80
4.5.8 Resultados Obtenidos.....	81
4.5.9 Observaciones y Comentarios (Sprint Review).....	81
4.5.10 Conclusión del Sprint 4 (Sprint Restrospective).....	82
4.6 DIAGRAMA DE CLASES DEL SISTEMA	82
CAPÍTULO V: PROCESO DE PRUEBAS DE USUARIO – RESULTADOS CON RETROALIMENTACIÓN.....	84
5.1 CRITERIOS DE EVALUACIÓN	84
5.2 PRUEBAS REALIZADAS	85
5.2.1 Tareas Realizadas.....	85
5.2.2 Prueba Pre-Test.....	88
5.2.3 Prueba Post-Test	90

5.2.4 Retroalimentación	92
5.2.5 Nuevas Historias de Usuario identificadas	93
5.3 MEJORA DEL PROTOTIPO	95
5.3.1 Nuevo Product Backlog	95
5.3.2 Mejoras realizadas	96
CAPÍTULO VI: CONCLUSIONES & RECOMENDACIONES	98
6.1 CONCLUSIONES	98
6.2 RECOMENDACIONES	98
REFERENCIAS.....	100
ANEXOS	102
ANEXO 1 – ENTREVISTA A JOSÉ TORRES.....	102
ANEXO 2 – ENTREVISTA A JOSUE GUERRA.....	108
ANEXO 3: ACTA DE REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES	112
ANEXO 4: ACTA DE REVISIÓN SPRINT 1 USUARIO CLIENTE:	113
ANEXO 5: ACTA DE REVISIÓN SPRINT 1 USUARIO ADMINISTRADOR	114
ANEXO 6: PRODUCTOS ENTREGADOS SPRINT 1	115
Pantalla Welcome:	115
Pantalla Login:	115
Pantalla Términos y Condiciones (Cliente – Administrador):.....	116
Pantalla Showrooms – Menu (Tipo de usuario: Cliente).....	116
Pantalla Mis Salas – Menu (Tipo de usuario: Administrador)	117
ANEXO 7: ACTA DE REVISIÓN USUARIO CLIENTE SPRINT 2:	118
ANEXO 8: ACTA DE REVISIÓN USUARIO ADMINISTRADOR SPRINT 2	119
ANEXO 9: PRODUCTOS ENTREGADOS SPRINT 2	120
CRUD Completo,.....	120

Pantalla para la Creación de una Sala	120
Pantalla con los detalles de la sala	121
Apartado de subir imágenes con vista previa:	121
Pantalla para la Gestión de Horarios Disponibles por Sala	121
Pantalla para Seleccionar las Fechas de la Reserva	122
Pantalla de Búsqueda	123
ANEXO 10: ACTA DE REVISIÓN USUARIO CLIENTE SPRINT 3	124
ANEXO 11: ACTA DE REVISIÓN USUARIO ADMINISTRADOR SPRINT 3	125
ANEXO 12: PRODUCTOS ENTREGADOS SPRINT 3	126
Pantalla del Historial de Reservas para el Usuario Cliente.....	126
Proceso de Cancelación de una Reserva.....	126
Control para evitar reservas fuera del horario establecido.....	127
Confirmación de Pago al momento de completar una reserva	128
Resumen administrativo (dueño de sala) sobre la actividad de las salas	128
ANEXO 13: ACTA DE REVISIÓN USUARIO CLIENTE SPRINT 4	129
ANEXO 14: ACTA DE REVISIÓN USUARIO ADMINISTRADOR SPRINT 4	130
ANEXO 15: PRODUCTOS ENTREGADOS SPRINT 4.	131
Pantalla Perfil (usuario cliente),.....	131
Pantalla Perfil (usuario administrador)	132
Screenshots sobre las notificaciones que se reciben al realizar una reserva	132
ANEXO 16: DICCIONARIO DE DATOS.	133
Tabla ‘usuario’	133
Tabla ‘saladeensayo’	134
Tabla ‘horario’	135
Tabla ‘reserva’	135

Tabla ‘pago’	136
Tabla ‘jamdollars’	136
Tabla ‘resena’	136
Tabla ‘favorito’	137
ANEXO 17: ACTIVIDADES PARA LAS PRUEBAS DE USABILIDAD (USUARIO CLIENTE):	138
ANEXO 18: ACTIVIDADES PARA LAS PRUEBAS DE USABILIDAD (USUARIO ADMINISTRADOR):.....	140
ANEXO 19: ACTA DE PARTICIPACIÓN – PRUEBAS DE USABILIDAD.....	143
ANEXO 20: PRUEBA PRE-TEST:	145
ANEXO 21: PRUEBA POST-TEST:.....	147
ANEXO 22: RESULTADOS FINALES PROTOTIPO	154
Botón para cambiar de rol sin perder la información del usuario.....	154
Acción al presionar el botón de eliminación:	155
Lista de detalles sobre equipos específicos:.....	155
Botón para eliminar imágenes:	156
UI Mejorada en la pantalla de Gestión de Horarios:.....	156
Botón para acceder a los detalles de la sala desde ‘Mis Reservas’:.....	157

ÍNDICE DE FIGURAS

Figura 1.	Actividad Cultural por cantón.....	2
Figura 2.	Actividad Cultural más Registrada en el cantón Quito.....	2
Figura 3.	Metodología SCRUM.....	14
Figura 4.	Tabla ‘usuario’.....	51
Figura 5.	Diagrama de Flujo Registro/Login.....	53
Figura 6.	Tabla ‘saladeensayo’.....	59
Figura 7.	Tabla ‘horario’.....	60
Figura 8.	Tabla ‘reserva’.....	60
Figura 9.	Diagrama de Flujo ‘Reserva de una Sala de Ensayo’.....	63
Figura 10.	Diagrama de Flujo ‘Creación y Gestión de una Sala de Ensayo.....	64
Figura 11.	Búsqueda de Salas por Geolocalización.....	64
Figura 12.	Tabla ‘pago’.....	69
Figura 13.	Diagrama de flujo para la cancelación de reservas.....	72
Figura 14.	Diagrama de flujo para la visualización de reservas dueño de sala.....	72
Figura 15.	Tabla ‘favoritos’.....	76
Figura 16.	Tabla ‘resena’.....	77
Figura 17.	Tabla ‘jamdollars’.....	77
Figura 18.	Diagrama de Flujo Acumulación de JamDollars.....	80
Figura 19.	Flujo 2 Envío de notificaciones por correo electrónico.....	80
Figura 20.	Diagrama de Clases de la Aplicación ‘QuitoJam’.....	82

ÍNDICE DE TABLAS

Tabla 1.	Comparativa entre Supabase y Firebase.	22
Tabla 2.	Comparativa entre Flutter Web y React Native.	24
Tabla 3.	Comparativa entre Visual Studio Code y React Native.	25
Tabla 4.	Comparativa entre Servicios de Hosting Estático.	27
Tabla 5.	Comparativa entre las metodologías SCRUM y Kanban	28
Tabla 6.	Requerimientos Funcionales y No Funcionales.	33
Tabla 7.	Roles de SCRUM definidos:.....	36
Tabla 8.	Historias de Usuario.....	36
Tabla 9.	Product Backlog.....	42
Tabla 10.	División del Product Backlog en Sprints	47
Tabla 11.	Historias de Usuario Asociadas al Sprint.....	48
Tabla 12.	Tareas del Sprint 1	50
Tabla 13.	Historias de Usuario Asociadas al Sprint 2.....	56
Tabla 14.	Tareas del Sprint 2	57
Tabla 15.	Historias de Usuario Asociadas al Sprint 3.....	67
Tabla 16.	Tareas del Sprint 3	68
Tabla 17.	Historias de Usuario Asociadas al Sprint 4.....	75
Tabla 18.	Tareas Asociadas al Sprint 4.....	76
Tabla 19.	Actividades Cliente – Promedio Tiempos	86
Tabla 20.	Actividades Administrador – Promedio Tiempos.....	87
Tabla 21.	Resumen Resultados Pre-Test.....	88
Tabla 22.	Resumen Resultados Post-Test.....	90
Tabla 23.	Nuevas Historias de Usuario.....	93

Tabla 24. Product Backlog Complementario.....95

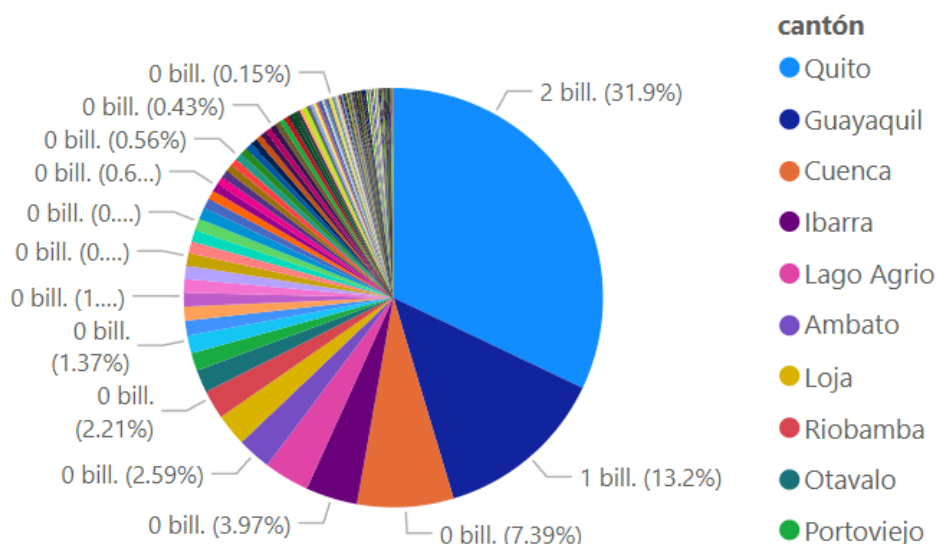
CAPÍTULO I: INTRODUCCIÓN

1.1 JUSTIFICACIÓN

Al momento de iniciar un proyecto musical, más allá de ensayar en un domicilio o practicar diariamente, el músico promedio desea escalar, no simplemente formando una banda, sino buscando un espacio en donde pueda crear música. En muchos casos, incluyéndome como músico aficionado, cuando se realiza una búsqueda dentro de aplicaciones como Google Maps por “salas de ensayo disponibles”, se obtiene resultados que, en su mayoría carecen de información detallada, como actualizada. Lo que dificulta al usuario encontrar buenas opciones dentro de la ciudad de Quito para la reserva o uso de sala de ensayo. Además, la mayoría de las salas de ensayo no cuentan con una aplicación web que les permita gestionar su disponibilidad, recibir reservas, observar información sobre el flujo de reservas o promocionar su espacio hacia la comunidad de músicos locales en la ciudad de Quito. Esta problemática puede influir directamente en la confianza tanto de los usuarios quienes publican un espacio, como de quienes buscan alquilarlo. De esta manera, cobra importancia el desarrollo de una aplicación con un diseño orientado a la experiencia del usuario (UI/UX), la cual brinde información clara, además de generar mayor seguridad.

Por otro lado, surge la pregunta: ¿Por qué concentrarse únicamente en la ciudad de Quito? Para responder esta pregunta se debe considerar la información proveniente del Registro Único de Artistas y Gestores Culturales (RUAC), cuyos datos son proporcionados por el INEC, gestionados por el Ministerio de Cultura y Patrimonio del Ecuador, actualizados el 7 de mayo del 2025. En la siguiente figura, podemos observar al cantón de Quito el cual concentra el 31.9% del total de actividades culturales registradas oficialmente a nivel nacional (ver figura 1), consolidándose como la ciudad con mayor movimiento artístico del país. En consecuencia, es la razón principal por la que el proyecto presente se concentre en la ciudad de Quito, al tener el mayor movimiento cultural del país lo que convierte a Quito como el epicentro ideal para el punto de partida. En este contexto, el presente proyecto no solo responde a una necesidad latente en la comunidad artística más grande del país, sino que maximiza su impacto potencial desde una fase inicial.

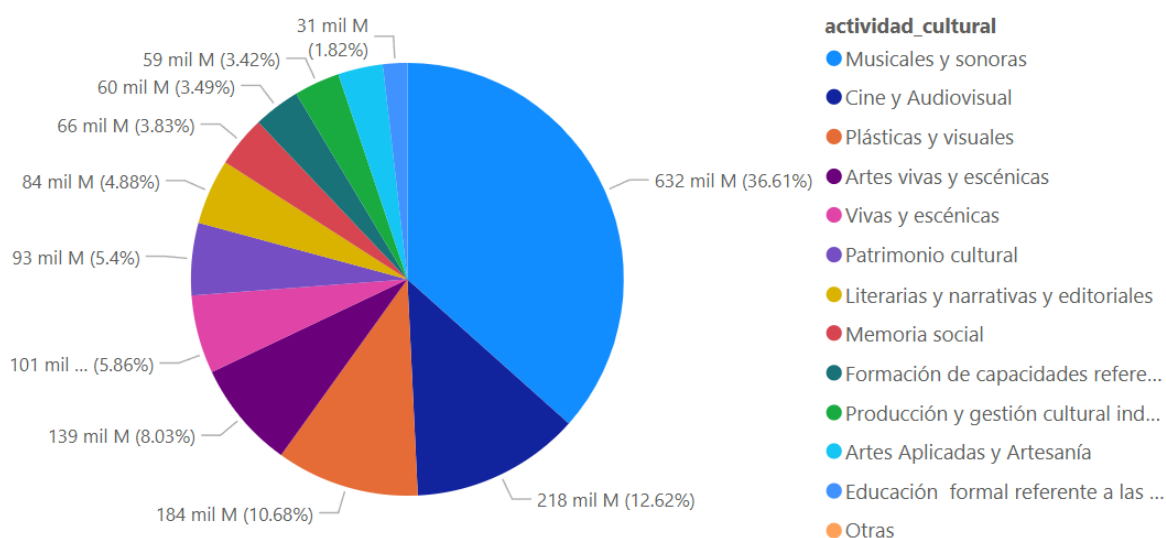
Figura 1. Actividad Cultural por cantón



Nota. Gráfico estadístico realizado por el Autor, con datos del RUAC, mediante PowerBI

Mediante un análisis más detallado por el tipo de actividad cultural dentro del cantón de Quito, se evidencia que las actividades más registradas corresponden a ‘Musicales y sonoras’ que representa el 36.61% (ver figura 2). En pocas palabras, se refleja mediante estos datos que Quito no solo posee una comunidad cultural activa, sino que la ciudad se destaca principalmente por su ámbito música.

Figura 2. Actividad Cultural más Registrada en el cantón Quito



Nota. Gráfico estadístico realizado por el Autor, con datos del RUAC, mediante PowerBI

En la anterior figura nos podemos dar cuenta la existencia de un mercado objetivo, lo que incrementa la viabilidad para poder dar una solución digital. Por lo que, el presente proyecto no trata únicamente de mejorar la búsqueda de las salas de ensayo, este proyecto trata de conectar musicalmente a los usuarios en cualquier parte de la ciudad. Esto se respalda tanto por las entrevistas realizadas (como el testimonio del propietario de “Liveroom Sessions” sobre el movimiento musical local), además de investigaciones académicas recientes. Titusunta Arias (2023), en su trabajo *“La construcción de la identidad en los jóvenes del sur de Quito en torno a la música extrema”*, sostiene que la música tiene una amplia capacidad de generar emociones. Los autores resaltan que, dentro de la juventud quiteña, la creación de música sea individual o en bandas, responden a una necesidad expresiva y colectiva, que va más allá de los géneros musicales. Lo que produce, especialmente en la ciudad de Quito, la necesidad de las personas, en especial de los jóvenes, en seguir creando música, sea de manera individual, como grupal. Como se observó en la figura 2, al existir una gran cantidad de artistas registrados oficialmente dentro de la ciudad, en consecuencia y de manera indirecta al existir una buena influencia musical, motiva a muchas personas en la creación de salas de ensayo, como de proyectos musicales tanto individuales, como de grupo.

Por tal motivo, este proyecto lo que trata es poder brindar una solución, a través de una aplicación web, la cual pueda unir a la comunidad musical de la ciudad. Desde esta perspectiva, la implementación de herramientas de búsqueda, como filtros, que puedan personalizar la información de búsqueda en temas concretos pretende optimizar significativamente la localización de estas salas de ensayo, mejorando así la experiencia del usuario. Dicha afirmación, se puede ver respalda por aplicaciones con gran éxito, como por ejemplo ‘*Airbnb*’, en la manera en cómo ellos brindan al usuario la posibilidad de personalizar la búsqueda de opciones, además de tener una mejor visión de los alojamientos disponibles, como la posibilidad de reservar y pagar desde la aplicación. En conclusión, se debe considerar una buena implementación de estas herramientas, presentando un producto 100% enfocado en brindar al usuario final la mejor experiencia posible en la búsqueda y reserva de salas de ensayo en la ciudad de Quito.

Por último, al desarrollar este proyecto podemos dar una oportunidad hacia el crecimiento musical en Quito, al permitir tener una mejor opción de búsqueda para los usuarios interesados en iniciar una carrera musical, como encontrar un espacio para crear música, o inclusive para los profesionales que tengan los equipos necesarios puedan sacar provecho de estos mismos,

de este modo poder generar un ingreso extra por sus espacios, como equipos. De esta manera, poder contribuir y brindar un fortalecimiento al ecosistema cultural de la ciudad.

1.2 PLANTEAMIENTO DEL PROBLEMA

Actualmente los músicos que requieren espacios para ensayar, enfrentándose a obstáculos en cuanto a la gestión eficiente de la información relacionada con la disponibilidad y reserva de salas de ensayo, plataformas como Google Maps o Facebook Marketplace no ofrecen resultados confiables, actualizados, o ajustados a las necesidades del usuario, lo que provoca la mala planificación de sus actividades, además de aumentar el tiempo invertido en la búsqueda. Por lo que, una de las problemáticas más importantes para el presente caso es la dificultad en el acceso a información pertinente en plataformas conocidas, aunque funcionales, resultan poco eficientes para la consulta de horarios y características de las salas disponibles. Además, de observar la interacción que tienen los músicos locales de la ciudad, siendo estos nuestros usuarios finales, sobre el prototipo brindado, para observar posibles mejoras hacia el proyecto, como la aceptación de este y el apoyo que tendría este producto si saliera al mercado. Considerando esta realidad, con el objetivo de evaluar la interacción de los usuarios finales (músicos apasionados, como profesionales) con el prototipo propuesto, así como su grado de aceptación y la viabilidad para su uso real, se plantearon las siguientes preguntas:

- **¿De qué forma se puede mejorar la búsqueda y reserva de las salas de ensayo disponibles en Quito, respondiendo a las necesidades principales de los músicos locales?**

Y las siguientes preguntas secundarias:

- ¿Existe actualmente una plataforma centralizada que gestione el uso de tecnología en la reserva de salas de ensayo en Quito?
- ¿Qué clase de datos tienen mayor impacto en la elección de los usuarios al reservar un espacio de ensayo, para optimizar la experiencia del usuario?
- ¿Cuáles son las principales dificultades que enfrentan los músicos y propietarios al consultar, como gestionar la información de las salas de ensayo?
- ¿Qué impacto tiene la usabilidad a la toma de decisiones de los músicos durante el proceso de reserva?

1.3 OBJETIVOS

1.3.1 Objetivo General

Diseñar y desarrollar el prototipo funcional de una aplicación web que facilite la reserva, como la gestión administrativa básica sobre las salas de ensayo musicales en el Distrito Metropolitano de Quito, respondiendo a las necesidades en específico de los músicos locales, como de los dueños de salas.

1.3.2 Objetivos Específicos

- Definir los requerimientos funcionales y no funcionales para el desarrollo de la aplicación web, para garantizar que se cumpla las necesidades de los músicos, como de los propietarios de las salas de ensayo en Quito.
- Implementar un sistema de búsqueda que permita a los usuarios localizar salas de ensayo según criterios como la ubicación, precio, entre otras características disponibles.
- Diseñar un sistema de reservas el cual sea funcional, como seguro, el cual permita a los usuarios seleccionar parámetros como fecha y horario.
- Implementar un apartado administrativo básico para que los dueños de las salas puedan crear, como editar sus salas de ensayo publicadas, además de gestionar los horarios de estas mismas.
- Evaluar la usabilidad y funcionalidad del prototipo mediante pruebas piloto con al menos 10 usuarios reales, utilizando métricas como tasa de éxito en tareas, y nivel de satisfacción.

1.4 ANTECEDENTES

Evidentemente, existen opciones de salas profesionales disponibles, por ejemplo, ‘Liveroom Sessions’ que es una sala de ensayo – grabación, la cual procura mantenerse actualizada en los medios. Con 13 años en el alquiler de espacios enfocados en la grabación y composición musical, si dispone de una información actualizada, pero se limitan a no tener un alcance mayor en la ciudad, como al tener muchos usuarios y no poder atenderlos a todos, lo que provoca que en algunos casos se crucen horarios, sin olvidarnos de la inseguridad que puede existir al momento de recibir una reserva de una persona la cual no conoce. Esto produce que muchos espacios no sean publicados en los medios sociales, siendo difundidos por métodos antiguos, como por ejemplo del boca a boca, y de recomendaciones. Limitando a que usuarios que están

iniciando en la música, como fue mi caso con mi banda, no puedan encontrar un espacio cercano para poder ensayar y grabar música, lo que produce que muchos dejen sus proyectos musicales de lado. Por otro lado, servicios ampliamente utilizados para la búsqueda de espacios o lugares en específico, como lo es *'Google Maps'*, carecen de funcionalidades como la disponibilidad en tiempo real, la búsqueda por filtros, características específicas con respecto a la sala de ensayo, como grabación musical, o un sistema el cual permita manejar reserva integrados limita su utilidad para los usuarios con necesidades particulares. Por consiguiente, esta necesidad se vuelve más relevante en ciudades las cuales tienen un alto nivel de movimiento artístico emergente, como lo es la ciudad de Quito, en donde existe un marcado interés por espacios que fomentan la creación musical.

Frente a este tipo de situaciones, resulta interesante observar modelos exitosos de aplicaciones como *'Airbnb'*, que permiten al usuario tener una mejor búsqueda personalizada, basándose en las necesidades para la búsqueda de un espacio de alojamiento. Además, de permitir a las personas las cuales constan con un espacio para ser habitable por algunos días, les brinda la oportunidad de poder publicar sus espacios y generar ingresos de estos. Así pues, se demuestra que una aplicación móvil con una alta usabilidad, personalización mejora la experiencia de uso y la retención de usuarios. Calle Garcia et al. (2024), destacan que en un 85% de los usuarios consideran a la usabilidad como un factor clave en su decisión de utilizar una aplicación de forma continua, o que esta sea su opción número 1 en utilizarla. Por lo que, dentro de la *'AppStore'* o *'GooglePlay'*, aplicaciones que superen más de las 4 estrellas, tienen una tasa de retención de usuarios más alta, superando un 20% de aquellas que no superan de las 4 estrellas, si observamos nos daremos cuenta de que aplicaciones como *'Airbnb'* y *'WhatSpot'* cuentan superan ese número de estrellas, lo que representa un éxito de la aplicación en países como los Estados Unidos.

A pesar de estos avances en soluciones digitales en otros países, dentro de la ciudad de Quito no existe una plataforma especializada la cual permita atender las necesidades de los músicos locales en cuanto a la reserva, como gestión de salas de ensayo-grabación, motivo por el cual existe una carencia limitada en la visibilidad de espacios para la creación de música. Esto justifica el desarrollo de este proyecto para observar el comportamiento de la aplicación en los usuarios finales, con el fin de ver como en la actualidad las personas buscan soluciones que puedan cubrir necesidades en específico, con personalización y ajustándose a las necesidades de los usuarios.

1.5 ALCANCE

El presente proyecto tiene como finalidad el desarrollo de un prototipo funcional, sobre una aplicación web orientada hacia la búsqueda personalizada, como la gestión de salas musicales (ensayo/grabación), dentro del Distrito Metropolitano de Quito. De este modo, para lograr con el objetivo general el presente prototipo debe constar con las funcionalidades mínimas viables para asegurar desde su operatividad, usabilidad y la posibilidad de evaluación durante el proceso de validación, hasta alcanzar con un prototipo final. Como consecuencia, para lograr con lo mencionado anteriormente se tomaron en cuenta los siguientes puntos:

- Un registro o inicio de sesión, para los usuarios músicos y para los propietarios de las salas de ensayo/grabación. Permitiendo identificar roles, con los cuales podemos manejar los distintos accesos en la aplicación web.
- Constar con una búsqueda de salas por ubicación y características. Lo que nos permite brindar al usuario brindar un enfoque más personalizado, ajustándose a las necesidades principales.
- Visualización de disponibilidad por fecha. De esta manera, permitir al usuario planificar sus reservas con anticipación.
- Reserva de salas con confirmación. Permitiendo al usuario el contar con una experiencia clara, como funcional sobre los procesos que se llevan a cabo dentro de la aplicación web.
- Panel básico, para que el propietario pueda gestionar la disponibilidad y la información del local.

Con las funcionalidades presentes se garantizará que la solución sea funcional, como usable y por último evaluable en el tiempo establecido. En otras palabras, necesitamos que la aplicación permita a los músicos acceder fácilmente a información relevante sobre estos espacios, optimizando el proceso de búsqueda, además que los propietarios puedan gestionar su disponibilidad de la sala de manera eficiente.

En conclusión, el presente prototipo está diseñado para dar una solución inicial, escalable, la cual pueda evolucionar hacia una aplicación de uso real a futuro. Si bien es cierto, que el presente alcance representa una versión mínima viable, es este prototipo final el cual nos puede permitir poner a prueba un flujo completo sobre la interacción entre músicos y dueños de salas.

CAPÍTULO II: MARCO TEÓRICO Y CONCEPTUAL

2.1 MARCO CONCEPTUAL

2.1.1 Sala de Ensayo/Grabación Musical

Podemos definir una sala de ensayo como un espacio diseñado para la práctica, ya sea individual o colectiva en base actividades musicales. Por eso, estos espacios están preparados acústicamente, para permitir una reproducción sonora nítida sin interferencias, lo que es importante para el desarrollo musical. Adicionalmente, se debe considerar aspectos clave de una sala de ensayo que influyen en su creación, siendo los fines con el cual se creó la respectiva sala de ensayo. Por ejemplo, en una sala diseñada para una banda de 4 personas como máximo no puede ser utilizada por una orquesta de 7 integrantes, por lo que es importante halar sobre los equipos necesarios que se utilizarán tanto para un ensayo, como para una grabación. En consecuencia, no solo se considera el espacio físico para realizar un ensayo, sino que los músicos se concentran en factores relacionados con la cantidad de equipos disponibles en la sala, tanto para un ensayo, como para una grabación.

En este sentido, además del tratamiento acústico, lo que una sala de ensayo debe contar con un equipamiento adecuado el cual pueda facilitar la práctica musical. Lo que incluiría el contar con instrumentos esenciales como guitarras, bajos, baterías y teclados, un sistema de amplificación, además de monitoreo tales como micrófonos y soportes, así como herramientas para la grabación y edición de audio. Cabe destacar, que existe una diferencia entre una sala de ensayo y un estudio de grabación, la sala de ensayo no siempre puede incluir los equipos necesarios para la grabación, como la edición, pero si debe cumplir con los instrumentos mínimos, un equipo de amplificación y por último un aislamiento acústico.

En conclusión, con este concepto definido, podemos conocer sobre los requerimientos mínimos que debería constar una sala de ensayo musical o estudio de grabación, características importantes que nos servirán para poder realizar filtrados, tales como búsqueda por equipos disponibles, por tipo de sala, inclusive por tipo de genero especializado que estaría dicha sala de ensayo.

2.1.2 Prototipo de Aplicación Web

El prototipo de una aplicación web es una representación preliminar y funcional de un sistema, el cual nos permite visualizar, como evaluar las características principales antes de su desarrollo completo y lanzamiento posterior. Dicho enfoque, nos permite identificar las mejoras en la experiencia del usuario, al igual que la estructura funcional de la aplicación. María Pascual (2024), mencionan en su artículo “*Beneficios del Prototipado en la Experiencia de Usuario*”, sobre el prototipado de aplicaciones enfocado en la experiencia de usuario (UX) es clave para la validación de ideas, como el poder de detectar errores en fases tempranas de desarrollo, además de optimizar la usabilidad dentro de la aplicación, o software en desarrollo. Dependiendo del nivel de fidelidad del prototipo con la idea final del desarrollo, los modelos de prototipo pueden representar aspectos básicos, como funcionalidades principales de la aplicación, los cuales ayudan a complementar a la mejora del producto final. El uso de estos prototipos no solo facilita la detección de problemas en fases tempranas, sino que por otro lado nos permiten mejorar significativamente la comunicación entre diseñadores, desarrolladores y usuarios finales.

Por otra parte, desde una perspectiva de diseño centrado en el usuario, el prototipado iterativo resulta importante para comprender las necesidades reales del público objetivo que tiene el desarrollo de este proyecto. En el artículo publicado en la *Revista Neuronum*, Salamanca Vega (2023), define el proceso iterativo como una metodología la cual combina una generación de prototipos, como de evaluación y por último de un refinamiento continuo, con el objetivo de una mejora progresiva en cuanto a la calidad, como la funcionalidad del producto al estar alineado hacia las necesidades reales del usuario. De esta forma, resulta fundamental el adoptar este enfoque en el desarrollo del prototipo, puesto a que ya definimos algunas de las funcionalidades clave basadas en necesidades reales, como lo es la búsqueda, la reserva, la visualización de disponibilidad, detalles de equipos, el proceso de reserva, de la creación de una sala, entre otras funcionalidades más; deben ser puestas a prueba, para ser optimizadas antes de un lanzamiento oficial. Concluyendo, que cada iteración realizada en el avance del desarrollo de este prototipo contribuye al perfeccionamiento de un producto final.

- **Evaluación de Prototipos**

Para garantizar la calidad del prototipo desarrollado, es necesario emplear la norma internacional ISO 9241-11:2018 “Ergonomía de la interacción hombre-sistema” que permitirá evaluar la usabilidad basándose en tres componentes principales: eficacia, eficiencia y satisfacción del sistema, que serán medidos a través de pruebas con usuarios reales, donde se observará el desempeño de las tareas y el tiempo que tardan en ser ejecutadas. Esta norma será útil para tareas específicas como reservar una sala y modificar preferencias, además de evaluar si la aplicación cumple con los objetivos desde el punto de vista de los usuarios finales.

2.1.3 Experiencia del Usuario (UX) & Diseño de Interfaz de Usuario (UI) mediante conceptos de Interacción Humano Computador (IHC)

La experiencia de usuario (UX), además del diseño de la interfaz de usuario (UI) son componentes clave para determinar una buena presentación y desarrollo de prototipos funcionales, sin generar errores o confusiones hacia los usuarios finales. Esto implica que, determinan la eficacia que se puede lograr en la aplicación, como la satisfacción con la que los usuarios finales interactúan con la misma. En consecuencia, necesitamos tener claro sobre la interacción humano computador (IHC) conjunto a estos conceptos de UX y UI, que se unen para el diseño y el desarrollo de interfaces las cuales sean intuitivas para el usuario. Además, de brindar accesibilidad en cuanto a su uso, para que de esta manera sea centrada en ofrecer la mejor experiencia de uso en nuestro prototipo a desarrollar.

Podemos tomar de ejemplo el caso estudiado de Rodríguez y Fernández (2024), los cuales efectuaron una investigación sobre cómo los principios de IHC permiten aumentar la interacción entre los usuarios finales y los sistemas digitales en entornos interactivos culturales. En este sentido, cabe destacar que, mediante la clarificación de las necesidades del usuario sobre el sistema, se logra facilitar los procesos, como la navegación y ejecución de tareas dentro del sistema en desarrollo, como consecuencia reforzando la comunicación entre los distintos actores (usuarios, desarrolladores, diseñadores). En conclusión, esto implica entender que una interfaz bien diseñada no solo limita al usuario a completar las actividades dentro de la aplicación sin errores, o complicaciones, sino que contribuye a la retención de los usuarios en utilizar la aplicación de manera frecuente.

A partir de lo anterior, es importante considerar los principios heurísticos de usabilidad por Jakob Nielsen, los cuales explicados por David Arezana (2022), son ampliamente utilizados para la evaluación de interfaces gráficas de usuario. Estos principios son:

- **Visibilidad del estado del sistema.** Trata sobre mantener informado al usuario de lo que está pasando dentro de la aplicación web, además de ofrecer un tiempo de respuesta breve.
- **Relación entre el sistema y el mundo real.** Trata sobre la conexión que debe tener el sistema con el usuario final, utilizando una gramática que resulte familiar para el usuario.
- **Control y libertad del usuario.** La equivocación de los usuarios es normal, por lo que debemos brindarle al usuario la posibilidad de subsanar el error.
- **Consistencia y estándares.** Al manejar colores con respecto a las acciones que se realizan, por ejemplo, asociar el rojo para eliminar, cuando hay errores, y el verde cuando todo está bien, o para indicar aprobación.
- **Prevención de errores.** Se basa en estar un paso por delante del usuario, al prevenir cualquier tipo de error que pueda cometer el usuario.
- **Reconocer antes que recordar.** Permitirle al usuario no memorizar sus acciones, sino de lo contrario hacer que las acciones o movimientos en la aplicación sean intuitivos.
- **Flexibilidad y eficiencia de uso.** Trata sobre como nuestra aplicación debe de ser utilizada tanto por usuarios nuevos, como expertos.
- **Diseño estético y minimalista.** Se refiere al contenido que en muchos casos no es necesario dentro de la aplicación, sin sobrecargar de información a los usuarios si no es necesario.
- **Ayudar a los usuarios a reconocer, diagnosticar y corregir errores.** Significa que todos los errores que pasen en nuestra aplicación sean entendibles para el usuario.
- **Ayuda y Documentación.** A pesar de que el objetivo sea que el uso de la aplicación no se necesitó un documento de ayuda, o manual. Siempre es útil brindarle al usuario un pequeño manual de funcionamiento.

2.1.4 Usabilidad

A la usabilidad la podemos entender como un atributo de calidad la cual evalúa qué tan fáciles e intuitivas pueden llegar hacer las interfaces web, como móviles. En otras palabras, se refiere al grado en el cual se evalúa como un usuario puede utilizar un producto o servicio para lograr sus objetivos específicos en un contexto de uso determinado. Esta definición destaca la importancia de diseñar interfaces las cuales permitan a los usuarios alcanzar sus metas de manera efectiva y sin mucho esfuerzo.

En nuestro ámbito de desarrollo de una aplicación web, la usabilidad es un factor muy importante que influye directamente en la experiencia del usuario y en la aceptación del producto. Según Couoh Novelo (2021), destaca que una interfaz bien diseñada puede facilitar la navegación, minimiza errores y facilita el aprendizaje del sistema, lo que contribuye a un mayor nivel de satisfacción, como una mejor experiencia al usuario.

2.2 METODOLOGIA

2.2.1 Ingeniería de requerimientos

Es una rama de la ingeniería en software relacionada con actividades de descubrimiento, documentación y mantenimiento de requerimientos para un software, consiste en identificar el propósito del software, al igual que el contexto en el que será aplicado. En este contexto, del presente proyecto se garantizará la satisfacción de los usuarios, la efectividad en el diseño e implementación del sistema, mientras que los requisitos reflejarán las necesidades funcionales (lo que debe hacer la aplicación) y las no funcionales (rendimiento y usabilidad de la aplicación).

2.2.2 Técnicas para el levantamiento de requerimientos

Un aspecto de suma importancia en el desarrollo de un sistema es el de determinar con precisión lo que el usuario espera del sistema, en base a las acción o procedimientos que espera cumplir mediante el sistema. Lo que nos permite aplicar diversas técnicas que se orientan a identificar los requerimientos reales, en este caso tanto de los músicos locales, como el de los dueños de salas. Por lo tanto, entre las técnicas más relevantes para el presente proyecto se encuentran:

- **Entrevistas**

Las entrevistas con músicos locales y propietarios de salas de ensayo, que nos permitirá conocer las necesidades, deseos y frustraciones actuales al momento de realizar una reserva, como de gestionar una sala de ensayo.

- **Historias de Usuario**

Son descripciones sencillas de las necesidades de los usuarios desde su perspectiva y tienen la siguiente estructura: Como [usuario], quiero [acción], para [lograr objetivo]; Como músico, quiero buscar salas de ensayo para poder practicar en un espacio que tenga los instrumentos y herramientas necesarias; Como músico, quiero que la aplicación me permita filtrar las salas de ensayo por ubicación/equipos/costo para seleccionar el lugar más adecuado para mi práctica, entre otras historias de usuario. Cabe mencionar que las historias de usuario son parte clave para conocer con exactitud la problemática actual la cual se necesita resolver.

- **Validación y verificación de requerimientos**

Posteriormente a que los requerimientos fueron recopilados, es necesario verificarlos. Se crearán prototipos funcionales para pruebas de usuario reales donde se verificará si se cumplen sus expectativas, lo que nos permitirá identificar errores, ambigüedades en el sistema, o inclusive mejoras antes de una implementación definitiva de un sistema.

- **Gestión de requerimientos**

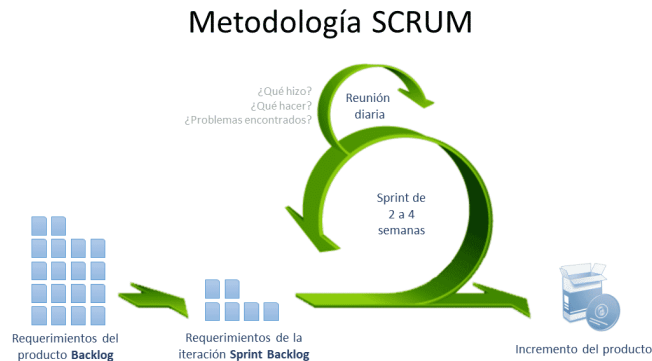
Una vez definidos y validados, se gestionará la evolución del proyecto, los requerimientos podrían cambiar a medida que se recopile más información en las entrevistas y si surgen nuevas necesidades. Por consiguiente, se obtiene una retroalimentación la cual permite que un sistema continúe siendo relevante y funcional, si se realizan los cambios sugeridos por los usuarios.

2.2.3 SCRUM

SCRUM es un marco de trabajo ágil que organiza el desarrollo de un proyecto en ciclos cortos llamados 'sprints'. Al finalizar cada sprint, se entrega una versión funcional del producto. Según el artículo sobre la Guía del Scrum, realizado por Schwaber y Sutherland (2020), esta metodología de SCRUM permite tener una facilidad en la transparencia, la inspección y la adaptación del proceso de desarrollo, promoviendo así una comunicación continua entre los desarrolladores y los usuarios finales. Por lo que, esta metodología es importante para el presente proyecto, por lo que se requiere flexibilidad para ajustar funcionalidades conforme a

retroalimentaciones periódicas. Em consecuencia de esto, cada sprint genera una versión más completa del proyecto final, en este caso del prototipo final a desarrollarse. Para entender de mejor manera esta metodología se realizará una explicación de los componentes principales de SCRUM en el presente diagrama:

Figura 3. Metodología SCRUM



Nota. Adaptado de Metodología SCRUM: Metodología ágil, por D. Calvo, s.f., DiegoCalvo.es. <https://www.diegocalvo.es/metodologia-scrum-metodologia-agil/>

Objetos clave de SRUM:

- **Product Backlog:** Se trata de una lista de tareas clasificada, en base al alza de requerimientos, en donde se prioriza todas las funcionalidades necesarias para un producto final.
- **Sprint Backlog:** Es el subconjunto de las actividades previamente mencionadas en el ‘Product Backlog’, para ser desarrolladas durante un sprint. Donde se detallan las tareas necesarias para cumplir con el sprint.
- **Incremento del Producto:** Son el resultado funcional final de cada sprint, el cual va agregando robustes al proyecto, además de ser el presentado cuando se ejecutan las reuniones de revisión.

Eventos de SCRUM:

- **Sprint Planning:** Se trata de la reunión inicial en donde todo el equipo define lo que se va a hacer y cómo se logrará hacerlo durante el sprint.
- **Daily SCRUM:** Es un encuentro entre el equipo de desarrollo de 10 a 15 min, para determinar lo que se está haciendo ese día.
- **Sprint Review:** Al final de cada sprint, se presenta el incremento hacia los usuarios finales, para obtener una retroalimentación de estos.

- **Sprint Retrospective:** Trata sobre una reunión interna, para discutir y reflexionar sobre los avances del proyecto, identificar mejoras y aplicarlas en el siguiente sprint.

A pesar de que SCRUM está diseñada para ser una metodología de rendimiento ágil, enfocada hacer aplicada a equipos de entre 5 a 10 personas, en el presente proyecto se ha optado por una implementación individual. Esta decisión responde al desarrollo académico del proyecto, el cual corresponde a una sola persona. Sin embargo, se mantendrá la esencia del marco metodológico de SCRUM mediante el uso de los ciclos de sprint, las reuniones enfocadas en generar validación, como retroalimentación en los avances de desarrollo sobre el prototipo de la aplicación web. Lo que nos permitirá validar funcionalidades clave, como adaptarse a los requerimientos emergentes, para mantener un enfoque iterativo.

2.2.4 Implementación de SCRUM, para el desarrollo del prototipo de aplicación web para la reserva y gestión sobre salas de ensayo.

La implementación de la metodología de SCRUM, en el desarrollo del prototipo de la aplicación web para la reserva y gestión de las salas de ensayo, no se limita en cuanto a su base teórica de su estructura ágil. Esta metodología se adaptó al presente proyectos por lo que es clave para garantizar flexibilidad, como adaptabilidad a los requerimientos propuestos por los usuarios finales, permitiendo una mejora continua del producto final, durante el proceso de desarrollo. Cabe mencionar, que el presente proyecto, además de brindar una posible solución al tema de la búsqueda, como la reserva de las salas de ensayo, está centrado en la experiencia del usuario final y en funcionalidades específicas, las cuales requieren una validación constante. Es por este motivo, que SCRUM nos proporciona un marco de trabajo el cual nos permita responder de manera eficaz hacia los cambios, mejoras y la retroalimentación brindada por el usuario final, a través de la ejecución de las reuniones por sprint.

Para poder implementar dicha metodología en nuestro desarrollo, durante la fase de planificación y alza de requerimientos, se identificaron las necesidades funcionales para poder establecer un primer prototipo de la aplicación. Basándonos en entrevistas para la recolección de información importante para levantar el desarrollo. En consecuencia, de esto, se identificaron elementos clave para desarrollar, como lo son la búsqueda por ubicación, la visualización de disponibilidad, la implementación de un sistema de reservas, el poder constar con reseñas y calificaciones, además del panel administrativo básico para la administración de las salas de ensayo. A partir de estos elementos identificados se estructura el ‘Sprint Backlog’,

seleccionando las tareas alcanzables para periodos de desarrollo, normalmente se manejan periodos de desarrollo de dos semanas.

Seguidamente, nosotros implementamos SCRUM dentro de nuestro proyecto, al contar con reuniones clave, que se les conoce como ‘Sprint Planning’, en donde se definen cada objetivo de cada ciclo. Lo que nos permite, constar con una buena organización de entregables por cada ‘Sprint Review’, en donde se presenta el incremento de la aplicación funcional para recibir retroalimentaciones, además de discutir sobre el futuro de esta, lo que genera una retrospectiva, la cual promueve la mejora continua. En consecuencia, estas dinámicas aseguran el flujo ágil y colaborativo, donde el proyecto en desarrollo puede adaptar nuevas ideas, cambios o errores identificados por los usuarios finales. Esta implementación de la metodología SCRUM garantiza que el desarrollo del prototipo pueda mantener un enfoque centrado en el usuario, con entregas constantes, con modificaciones en cada sprint, además de una validación continua de las decisiones en cuanto al diseño, como de la implementación del prototipo. De esta manera aseguramos que el producto final, que es el prototipo del proyecto, pueda responder hacia las necesidades reales de la comunidad musical en la ciudad de Quito.

2.2.5 Beneficios en cuanto la implementación de SCRUM para el desarrollo del prototipo de aplicación web la reserva y gestión sobre salas de ensayo.

Mediante la implementación de la metodología SCRUM para el desarrollo del prototipo final, sobre las reservas de sala de ensayo en nuestra aplicación web, representa una decisión enfocada en maximizar la eficiencia de trabajo, como la flexibilidad y calidad en cada uno de los procesos de construcción de la aplicación web. Por consecuencia, al tratarse de un proyecto el cual está enfocado en usuarios reales, tanto músicos y propietarios de salas, la capacidad de SCRUM es ideal para adaptarse a los requerimientos funcionales, solicitados por los usuarios, los cuales pueden cambiar a lo largo del desarrollo, por lo que, al realizar entregas progresivas en cada sprint, resulta beneficioso para un producto final 100% enfocado en los usuarios.

Otro de los beneficios de SCRUM, es la priorización de actividades mediante la creación del ‘Product Backlog’, el cual nos permite determinar el nivel de complejidad de la actividad, y de esa manera poder ordenar los entregables por cada sprint en base a los requerimientos plasmados en el backlog. Esto contribuye a la satisfacción por parte del usuario final, por lo que el sistema responde a sus necesidades principales. Por otro lado, desde un enfoque más técnico, SCRUM impulsa al desarrollo en la mejora continua mediante los ciclos del sprint, los cuales corresponden a ciclos con retroalimentación estructurados, esto nos permite ajustarnos

a cambios o mejoras basándonos en la opinión del usuario final. Finalmente, con la gestión y historial del proyecto, promovida por el uso de esta metodología, nos permite tener un seguimiento claro sobre el proceso, como la toma de decisiones en todo el desarrollo del prototipo final.

Podemos concluir, el conjunto de estos beneficios consolida a SCRUM como una metodología adecuada para el desarrollo del prototipo final, la cual nos permite elevar el nivel de calidad del producto final, sino también que nos permite mejorar la experiencia en cuanto al desarrollo de este tipo de aplicaciones web.

2.3 HERRAMIENTAS

Para el desarrollo del prototipo web del presente proyecto, se tomó en consideración el uso de estas herramientas eficientes, las cuales permiten una rápida implementación, escalabilidad y facilidad de mantenimiento. A continuación, se describen las principales herramientas utilizadas para el desarrollo del presente prototipo.

2.3.1 Flutter

Flutter es un equipo (kit) de desarrollo de software (SDK), creado por Google, el cual está diseñado para construir aplicaciones nativas de alta calidad, a partir de una sola base de código. Flutter, nos permite desarrollar para múltiples plataformas, incluyendo dispositivos móviles (iOS/Android), web y de escritorio, eliminando de esta forma la necesidad de escribir código en específico para cada sistema operativo o plataforma.

Seguidamente, debemos tener en cuenta que uno de los pilares fundamentales que hacen de Flutter una buena opción para el desarrollo de aplicaciones multiplataforma, es su arquitectura en capas la cual está diseñada para ofrecer un sistema escalable, además de personalizable. Este tipo de arquitectura incluye componentes como un motor de renderizado, conocido como Dart, la biblioteca de widgets, además de las herramientas de desarrollo. Por lo tanto, gracias a esta estructura modular, los desarrolladores tienen un control total sobre cada aspecto de la interfaz gráfica de la aplicación y el comportamiento de esta, esto nos permite adaptar, como extender las funcionalidades según las necesidades del proyecto en desarrollo.

Una de las características más destacadas en cuanto a la funcionalidad de ‘hot reload’, la cual nos permite a los desarrolladores ver los cambios realizados dentro del código en tiempo real, sin necesidad de reiniciar toda la aplicación. Esta capacidad nos permite acelerar el proceso de desarrollo iterativo, como mejorar la productividad dentro de entornos de prueba, sobre todo

en el prototipado web. Por otro lado, Flutter nos permite desplegar aplicaciones web responsivas utilizando la misma base de código para aplicaciones móviles, lo que optimiza el esfuerzo en cuanto a desarrollo, además de mantener una experiencia de usuario consistente. En conclusión, el SDK de Flutter, enfocado para el desarrollo de páginas web (Flutter Web) proporciona una gran variedad de widgets altamente personalizables, como interactivos y adaptados a distintos navegadores, lo que garantiza una experiencia visual coherente, la cual esta alineada con los estándares visuales de aplicaciones contemporáneas.

2.3.2 Supabase

Supabase es una plataforma de tipo BaaS (Backend-as-a-Service) de código abierto, que nos proporciona una alternativa moderna, y escalable a diferencia de servicios como ‘Firebase’, además Supabase consta de una base sólida sobre PostgreSQL. Este servicio en línea está diseñado para simplificar el desarrollo de aplicaciones web al ofrecer varios servicios integrados, desde la autenticación por correo electrónico, como el de realizar bases de datos relacionales, el almacenamiento de archivos y funciones serverless, todo gestionado desde una única interfaz. Destacando, que se puede utilizar gratuitamente para el caso de desarrollo de un prototipo, como lo es nuestro caso.

Por lo que, dentro de nuestro contexto del presente proyecto Supabase cumple un rol central para la gestión en cuanto a la persistencia de datos, la lógica del negocio, por último, la autenticación de usuarios. Destacando, su sistema de autenticación integrado nos permite registrar a los usuarios de manera segura, manejando sus distintos roles (usuario cliente – usuario dueño o admin), siendo importante para nuestra aplicación web de reservas. Al mismo tiempo, tiene una compatibilidad nativa con Flutter, permitiendo una integración fluida entre el frontend y backend de la aplicación, reduciendo la complejidad al momento de configurar, como de dar mantenimiento. Sin embargo, una de sus principales limitaciones se encuentra al utilizar su versión gratuita, la cual limita de espacio en la base de datos, esto no tiene importancia en la realización, como puesta en prueba del prototipado, pero si cuando se quiera trabajar con aplicaciones que mantengan altos niveles de flujo de usuarios y datos.

Sin embargo, el uso de PostREST se considera como una ventaja al permitirnos tener una capa que expone a la base de datos como una API RESTful automática. La cual nos permite realizar operaciones de CRUD de manera estructurada, eficiente y sobre todo segura. Además, de que

Supabase nos permite escribir, como implementar funciones personalizadas (RPC), las cuales encapsulan lógica compleja directamente en la base de datos, lo que nos permite mejorar el rendimiento cuando realizamos una consulta, como de centralizar las reglas del negocio.

Finalmente, cabe destacar que gracias al enfoque que manejan, el cual está orientado hacia el desarrollador. Supabase nos ofrece una experiencia de desarrollo intuitiva, además de una documentación extensa, control en cuanto a los logs generados automáticamente por el servicio, y contar con herramientas visuales que facilitan la administración de los datos. Lo que transforma a Supabase como una solución ideal para el desarrollo de aplicaciones web, sobre todo si se trata de un prototipo, como el que se propone en el presente proyecto.

2.3.3 Visual Studio Code

Visual Studio Code es un editor de código fuente desarrollado por Microsoft, principalmente diseñado para ser ágil, como ligero, además de ser altamente personalizable. Visual Studio a diferencia de otros editores de código, como IDE avanzadas, está enfocado a ofrecer la mejor experiencia al usuario, en términos de agilidad para la escritura y modificación de código sin la necesidad de sacrificar características avanzadas. Además, Visual Studio Code consta con una amplia variedad de extensiones, como de herramientas, las cuales nos ayudan adaptándose a cualquier tipo de desarrollo con el cual estemos trabajando, en nuestro caso si existe una compatibilidad para poder ejecutar código de tipo Dart para poder trabajar con Flutter.

Entre otras características principales de Visual Studio Code, encontramos que cuenta con el autocompletado inteligente mediante la herramienta desarrollada de 'IntelliSense', que nos ayuda a poder predecir lo que debemos escribir en nuestro código, basándose en lo que tenemos desarrollado dentro de nuestro proyecto. Por otro lado, en cuanto a la depuración integrada nos permite ejecutar nuestro código directamente dentro del editor lo que nos permite observar la ejecución de código en tiempo real, y poder agregar puntos de interrupción con el objetivo de observar la ejecución de manera más detallada, permitiéndonos encontrar distintos errores que pueden existir en nuestro código. Además, una de las características más importantes que tiene Visual Studio Code, es el de contar con compatibilidad con Git para el control de versiones desarrolladas dentro de un proyecto. Por lo que, para el presente proyecto es pertinente manejar un controlador de versiones como lo es Git, tema que hablaremos más adelante, pero que es importante mencionar que tiene una buena compatibilidad directa con el entorno de desarrollo de Visual Studio Code, llevando a poder constar con un historial de versiones y cambios durante todo el desarrollo.

Asimismo, una de las ventajas de utilizar Visual Studio Code para el presente proyecto es el hecho de que es de uso gratuito, además ser de código abierto. Lo que permite tener una amplia comunidad que puede contribuir al desarrollo de nuevas funcionalidades en Visual Studio Code, de la misma manera sobre la creación de nuevas herramientas. Esto nos asegura que esta plataforma para desarrollar consta de un alto nivel de soporte por parte de una comunidad activa. Lo que significa que hay un soporte constante con lenguajes de programación, en este caso en entornos de desarrollo con Flutter y su soporte en Dart, lo que nos brinda seguridad al momento de desarrollar nuestro proyecto utilizando esta plataforma.

2.3.4 Git & GitHub

Git es un sistema enfocado en el control de versiones de distribución gratuita y de código abierto, el cual está diseñado para gestionar varios campos con velocidad y manejando un buen grado de eficiencia. Este sistema, nos permite realizar un seguimiento de las modificaciones constantes que existen dentro del desarrollo de un proyecto, a nivel de archivos modificados del proyecto. Por lo que, nos permite tener un control especializado cuando se trata de realizar desarrollos de manera grupal, por lo que nos permite hacer un seguimiento sobre cuando se aplicaron cambios en el desarrollo del proyecto.

Por otro lado, la plataforma en la nube GitHub es la cual nos permite almacenar, compartir y trabajar en nuestros desarrollos de proyectos, junto a otros usuarios. Sirve como una red social, en la cual nosotros podemos contar con un perfil, tener nuestros repositorios de proyectos, con el propósito de poder compartirlo con otros usuarios y trabajar al mismo tiempo efectuando cambios, como modificaciones dentro de los proyectos en desarrollo. Además, para un uso individual, GitHub nos permite tener nuestros repositorios en la nube de manera privada, de esta manera conjunto a Git nosotros podemos contar con un registro o historial de las versiones desarrolladas en nuestro proyecto.

Hablando un poco más a detalle, sobre cómo funcionan Git y GitHub, podemos mencionar que, al momento de cargar archivos dentro de GitHub, estos archivos se los almacena en un repositorio de Git, lo que significa que, al momento de realizar algún cambio en los archivos del proyecto de GitHub, Git se iniciaría de manera automática para poder realizar seguimientos en cuanto a los cambios, para poder administrarlos. En otras palabras, primero se trabaja de manera local, por lo que primero deberíamos sincronizar el proyecto local hacia un repositorio en GitHub, entonces cualquier cambio que realicemos en el proyecto nosotros podemos utilizar comandos como “Git Status”, para saber el estado actual con respecto a los cambios realizados

en el proyecto, seguidamente de “Git Add .”, para inicializar el proceso de cambios en el proyecto, “Git commit -m “ para añadir algún comentario a los cambios realizados, para finalmente ejecutar “Git Push” el cual nos permite insertar las modificaciones del proyecto local hacia proyecto que está en el repositorio de GitHub. Dentro del contexto de nuestro proyecto esta herramienta es muy útil para poder mantener un registro activo sobre el historial de cambios que existieron durante el proyecto, permitiéndonos tener un control, como visualización sobre los cambios dentro del prototipo durante su desarrollo.

2.3.5 Servicios de APIs

Podemos definir a una API (Application Programming Interface), como un conjunto de protocolos los cuales son utilizados tanto para diseñar, como integrar software dentro de una aplicación. Las APIs, permiten nos permiten simplificar el desarrollo de nuestras aplicaciones, lo que nos permite ahorrar tiempo y dinero. Este tipo de software nos permite tener flexibilidad, además de simplificar el diseño, como la administración del uso de las aplicaciones.

En otros términos, una API actúa como un puente el cual nos permite comunicarnos con dos componentes de software entre sí. En nuestro proyecto aplicaremos el uso de APIs, las cuales no solo se limitarán a la conexión entre el frontend en Flutter con el backend en Supabase, sino que nos permiten extender nuestras funcionalidades mediante el uso de servicios externos. Tales servicios como los de geolocalización, mapeo, navegación, servicios de notificaciones, además de los ya integrados en Supabase para la autenticación de usuarios, subida de imágenes con almacenamiento, y finalmente el uso de funciones RPC que podremos utilizar en Supabase mediante su respectiva APIs.

En conclusión, el uso de APIs para el desarrollo actual del proyecto nos garantizará que el prototipo sea escalable, seguro, como adaptable a nuevas funciones en el futuro, basándonos en desarrollar un prototipo final con funciones y acciones, que garanticen la buena experiencia fluida hacia el usuario final.

2.3.4 Firebase Hosting

Firebase es una plataforma enfocada en el desarrollo de aplicaciones móviles, como web, desarrollada por Google. Esta nos ofrece una variedad amplia de servicios backend, por lo que podemos clasificar a Firebase como un servicio de tipo BaaS, lo que nos permite utilizar servicios de backend como el uso de bases de datos, almacenamiento de archivos y hosting.

Uno de los servicios más destacables de esta plataforma es su servicio de ‘Firabase Hosting’, por el cual podemos realizar el despliegue de aplicaciones web de manera rápida y segura.

Para el presente trabajo, de los varios servicios que se disponen en esta plataforma, ocuparemos el conocido como ‘Firestore Hosting’, el cual nos ofrece seguridad al momento de desplegar nuestra aplicación, nos ofrece de igual forma certificados SSL, además de contar una integración con CDN (Content Delivery Network) para obtener una mayor velocidad global. Agregando el hecho de que tiene compatibilidad con las páginas desarrolladas en Flutter Web. Su simple configuración hace que Firabase Hosting sea una excelente opción para desplegar un prototipo, como es el caso del presente proyecto.

A pesar de lo beneficioso que puede llegar hacer Firebase en varios aspectos, en el presente caso de este proyecto no se utilizó Firebase para el backend principal del prototipo, porque para el presente desarrollo se buscaba una solución basada en bases de datos SQL tradicionales, en este caso Supabase que utiliza PostgreSQL, como hablamos anteriormente nos proporciona un mayor control sobre las funciones de negocio mediante el uso de procedimientos almacenados (RPC). Dentro de este contexto Supabase ofrece una alternativa acorde a las necesidades de contar con un modelo de datos relacional, con la lógica que el prototipo pueda requerir. No obstante, se consideró el uso de Firestore Hosting como una opción viable para el despliegue de la aplicación web, para poder realizar simultáneamente la prueba de usuarios con el primer prototipo, puesto que Firabase cumple con los requisitos de disponibilidad, escalabilidad y seguridad para paginas construidas con Flutter Web.

2.1 COMPARATIVA DE LAS HERRAMIENTAS UTILIZADAS

Para observar más a detalles del motivo de seleccionar las presentes herramientas para el desarrollo, como para el despliegue del prototipo. En este contexto, se optó por realizar tablas comparativas sobre las posibles alternativas de las herramientas implementadas en este proyecto. Cabe destacar que las presentes tablas fueron realizadas mediante el apoyo de inteligencia artificial, en este caso de ChatGPT 4o (2025).

2.4.1 Supabase vs Firebase

Tabla 1. Comparativa entre Supabase y Firebase.

Característica	Supabase	Firebase
----------------	----------	----------

Tipo de Bases de Datos	Relacional (Utilizando PostgreSQL)	NoSQL, utilizando un modelo propio conocido como Firestore, o Realtime Database.
API Rest Automática	Si, mediante PostgresREST	No directamente, se requieren funciones Cloud o Firestore SDK.
Funciones Personalizadas	Si, mediante funciones SQL o PostgreSQL RPC.	Si, mediante Cloud Functions (Node.js)
Almacenamiento de Archivos	Sí, basados en buckets	Sí, con Firebase Storage
Dashboard Administrativa	Si, consta con editor visual de tablas, control de logs, se pueden crear vistas y funciones de SQL, mediante recursos visuales.	Completo, pero orientado más al uso de Firebase console.
Licencia	Código Abierto (MIT)	De propiedad de Google
Integración con Flutter	Más amigables, con conocimiento de SQL.	Sencillo de implementar si se trata de aplicaciones móviles.
Hosting Integrado	No	Si (Firebase Hosting)
Aporte de la Comunidad	Alto, en crecimiento, con una buena calidad, con enfoque developer-friendly.	Extensa con una comunidad activa.
Plan Gratuito	Plan gratuito generoso, pero con limitaciones en espacio.	Generoso, especialmente en funciones clave.

Nota. Tabla desarrollada con el apoyo de IA

Como podemos observar en la presente tabla, ambas opciones cumplen para el almacenamiento y gestión de datos, para el desarrollo de aplicaciones tanto móviles, como de web. Sin embargo, en nuestro proyecto, considerando que es una aplicación web de reservas con múltiples relaciones entre usuarios, salas, reservas, como reseñas. En consecuencia, de esta necesidad de manejar una base de datos relacional, se decidió el uso de Supabase. Teniendo en cuenta la cantidad de relaciones entre las entidades de nuestro prototipo. Por lo que, Supabase al estar construido sobre PostgreSQL, nos permite desarrollar, como implementar una base de datos relacional ideal para modelar con múltiples entidades relacionadas.

2.4.2 Flutter Web vs React Native

Tabla 2. *Comparativa entre Flutter Web y React Native.*

Característica	Flutter Web	React Native
Lenguaje de Programación.	Dart	JavaScript / TypeScript
Renderizado en Web.	CanvasKit / HTML renderer (control total de UI)	Depende de React DOM o React Native for Web (con bridge limitado)
Compatibilidad Móvil.	Nativa (Flutter fue primero para el desarrollo móvil)	Nativa (orientado a iOS y Android, Web es experimental)
Desempeño Web.	Alto, especialmente con CanvasKit	Variable, pueden existir limitaciones gráficas.
Reutilización de Código.	Alta, el mismo código Dart puede ser implementado para web, móvil y escritorio.	Alta, pero con ciertas adaptaciones según la plataforma
Personalización de UI.	Muy alta, Flutter contiene un propio sistema de renderizado.	Alta, pero con dependencia del estilo web nativo.
Soporte Oficial Web.	Sí, estable desde Flutter 2.0	Experimental.

Despliegue Web.	Genera aplicaciones en HTML/JS listo para hosting estático.	Se requiere de una configuración adicional si se usa React Native Web.
Documentación Oficial (Web).	Clara y bien estructurada por parte de Google.	Limitada, con soporte de comunidad en React Native Web.

Nota. Tabla desarrollada con el apoyo de IA

En la presente tabla nos podemos dar cuenta de las características principales, de cada una de las herramientas expuestas, por lo que para nuestro desarrollo del prototipo. Escogimos a Flutter Web, por su única base de código para móvil y web con resultados consistentes. Se considero, que para el presente proyecto se necesita la visualización de mapas, requiriendo un renderizado personalizado que se maneja mejor con Flutter. Además, de mantener una coherencia visual entre pantallas, sin depender del estilo nativo del navegador. Por último, a través de su integración con Supabase, como sus widgets personalizados mejora la productividad frente al ecosistema fragmentado por parte de React Native Web.

2.4.3 Visual Studio Code vs Android Studio

Tabla 3. Comparativa entre Visual Studio Code y React Native.

Característica	Visual Studio Code (VS Code)	Android Studio
Tipo de Herramienta.	Editor de texto liviano y extensible.	IDE completo, desarrollado por Google.
Peso y Rendimiento.	Ligero, rápido de instalar y de ejecutar.	Es más pesado, por lo que consume más recursos del sistema.
Lenguajes Soportados.	Es multilenguaje, compatible con Dart, JavaScript, Python, etc.	Especializado en Java, Kotlin, y ahora con Dart (mediante plugins).

Integración con Flutter	Excelente, con extensiones de Flutter y Dart oficiales.	Consta de un soporte oficial completo desde Google con Flutter plugin.
Depuración y Testing.	Soporte con consola integrada, breakpoints, y una terminal interactiva.	Depurador avanzado con visualización de UI, memory y CPU profiling.
Herramientas de Autocompletado	Muy bueno, en especial con extensiones.	Intelij IntelliSense completo por defecto.
Interfaz de Usuario	Minimalista, personalizable.	Completa, pero compleja de entender.
Extensiones Disponibles	Enorme ecosistema de desarrollo (Flutter, Git, Markdown, Docket, etc).	Mas limitadas, dependiendo del entorno de JetBrains.
Control de Versiones (GIT)	Integrado, visual y práctico.	Integrado, pero con más pasos.
Uso de Recursos	Bajo	Alto
Licencia	Gratuita (De open source)	Gratuita, pero con componentes de propiedad.

Nota. Tabla desarrollada con el apoyo de IA

Ambas opciones, representan un marco completo en cuanto a herramientas para el desarrollo de software, sin embargo, para el presente proyecto se optó por el uso de Visual Studio Code por la ligereza la cual facilita el desarrollo cuando se tienen equipos de recursos moderados. Además, de contar con una amplia compatibilidad de extensiones para Flutter y Dart, lo que nos permite tener un flujo de trabajo ágil. Consta con una integración directa con Git, esencial para manejar un control de versiones del proyecto, y por ser más sencillo de utilizar. Por otro lado, cuenta con mayor facilidad para cambios rápidos, para hacer prototipado, pruebas visuales, además de la edición de múltiples archivos de manera simultánea sin sobrecargar el

sistema. Finalmente, consideramos a Android Studio como perfecto para etapas de optimización a futuro, pero Visual Studio Code permite un desarrollo fluido, como eficiente para el prototipo.

2.4.4 Comparativa de Servicios de Hosting

Tabla 4. Comparativa entre Servicios de Hosting Estático

Característica	Firebase Hosting	Netlify Inc.	Vercel
Propietario	Google	Netlify Inc.	Vercel Inc.
Enfoque principal	Hosting para aplicaciones web, integración con Firebase	Hosting JAMstack y funciones serverless	Hosting frontend y serverless de Next.js
Soporte para Flutter Web	Si, excelente para proyectos estáticos	Si, genera hosting desde carpetas '/build'	Si, compatible con '/build/web' de Flutter
Despliegue automático desde Git	Si, con GitHub o CLI	Si, con GitHub, GitLab, Bitbucket	Si, sencillo con GitHub
Certificado SSL gratuito	Si, automático	Si, automático	Si, automático
Dominio Personalizado	Si	Si	Si
Funciones Serverless	Si (Cloud Functions, se paga aparte)	Si, funciones integradas (Netlify Functions)	Si, integradas (Serverless Functions)

Velocidad y CDN (Content Delivery Network)	Alta, CDN de Google global	Alta, CDN global de Netlify	Alta, CDN propio optimizado para frontend
Ventajas Específicas	Integración nativa con Firebase Auth, DB, etc.	Preview deploys automáticos por rama	Previsualización automática por PR, SSR listo

Nota. Tabla desarrollada con el apoyo de IA

Las tres opciones para hosting ofrecen ventajas en cuanto a rápidas, seguridad, y posibilidad para escalabilidad en el proyecto a futuro. Sin embargo, pese a las ventajas observadas en cuanto al hosting con Netlify y Vercel, se optó por utilizar Firebase Hosting por su alta compatibilidad con aplicaciones web desarrolladas por Flutter, además de su sencillas para desplegar la aplicación web desde Visual Studio Code, además de la posibilidad de actualizar la página desde la terminal del proyecto. En consideración con lo anterior dicho, Firebase Hosting es ideal para el despliegue de prototipos funcionales, contando con una integración segura, como gratuita en ambientes de desarrollo académicos.

2.4.5 Comparativa de Metodologías de Desarrollo

Tabla 5. Comparativa entre las metodologías SCRUM y Kanban

Características	SCRUM	Kanban
Tipo de Metodología	Ágil, iterativa e incremental	Ágil, flujo continuo
Organización del Trabajo	Sprints fijos (1-4 semanas)	Flujo continuo, sin iteraciones fijas.
Roles Definidos	Sí (Scrum Master, Product Owner, Equipo).	No necesariamente.
Tablero de Tareas	Si, dividió por sprints	Sí, con límites de trabajo en progreso (WIP)

Entrega de Valor	Frecuente, al final de cada Sprint	Constante
Revisión y Mejora	Reuniones de revisión y retrospectiva	Revisión informal y continua
Documentación Requerida	Moderada, centrada en backlog y tareas	Baja
Cambio de Requisitos	Se permite entre Sprints	Se adapta continuamente

Nota. Tabla desarrollada con el apoyo de IA

Para el presente proyecto se optó por el uso de la metodología SCRUM, por lo que nos permite iterar rápidamente sobre las funcionalidades, como reservas, búsqueda, gestión de las salas de ensayo, entre otras. Además, de que SCRUM cuenta con un enfoque de entregas parciales, lo que facilita mantener un desarrollo constante, como controlado por tiempos de entrega. Por otro lado, por cada sprint existe la necesidad de realizar pruebas frecuentes con usuarios reales, lo que nos permite adaptarnos al ciclo de SCRUM (Sprint – prueba – mejora). Finalmente, su estructura facilita rápidamente a la gestión del proyecto académico, donde el tiempo es limitado, por lo que se requiere una visibilidad constante del avance.

CAPÍTULO III: ALZA DE REQUERIMIENTOS

3.1 ALZA DE REQUERIMIENTOS

El alza de requerimientos se realizó en base a dos entrevistas, una de ellas fue a José Torres, el cual es dueño de una sala de ensayo al norte de la ciudad conocida como “Liveroom Session”. De igual manera, la segunda entrevista se realizó a Josué Guerra el cual tiene como hobby la música. Cabe destacar que, en ambas entrevistas se encontraron aspectos positivos, algunas sugerencias de mejora, además de un mejor entendimiento de lo que se quiere lograr con esta aplicación web.

3.1.1) Primera Entrevista:

La primera entrevista se realizó a José Torres (Anexo1), y para resumir lo que fue la entrevista nos apoyaremos de las siguientes preguntas clave:

- **¿Cuál fue el objetivo de la entrevista a José Torres?**

El principal objetivo de la entrevista realizada a José Torres fue el de entender el manejo y giro del negocio de las salas de ensayo/grabación. Además, de identificar las necesidades, como expectativas con respecto a las prestaciones de la aplicación para los usuarios que sean dueños de una sala de ensayo.

- **¿Cuál es el perfil de José Torres?**

José Torres es un sonidista profesional de Quito, el cual ha participado en más de 100 eventos a nivel nacional, como internacional con bandas ecuatorianas, como internacionales. Tiene 13 años dirigiendo su sala de ensayo, la cual se ha convertido como una de las salas más concurridas y conocidas de Quito, teniendo clientes de todo tipo, desde apasionados por la música, hasta bandas profesionales como la Rocola Bacalao, Bazca, entre otras.

- **¿Qué tipo de metodología se utilizó en la entrevista?**

La principal metodología que se utilizó fue una entrevista semiestructurada con preguntas de respuesta abierta en las cuales el entrevistado podía mencionar otros temas además del principal, aportando más información sobre las necesidades principales como el llenar espacios vacíos en horarios de atención, tener seguridad en recibir clientes que tengan un perfil de músico, entre otros temas más.

- **¿Cuáles fueron los puntos clave de la entrevista?**

Los puntos clave que se obtuvieron en la entrevista fueron los siguientes:

- El uso de la sala de ensayo va por horas, teniendo un intervalo de precios de 5\$ a 25\$). En el caso de “Liveroom Session”, se cobra un valor de 13\$ la hora y se graba la sesión realizada por los músicos como extra del servicio. Si el artista necesita grabar un disco o canción, obviamente se habla de un proyecto que podría demorarse hasta dos días, pero de igual manera se cobra por hora 13\$, en donde se incluye la producción y mezcla de lo que se grabó.
- La producción musical y el alquiler de equipos, como espacios, es un mercado o sector que no se ha tenido mucho en cuenta, a pesar de que existe un incremento en la demanda de uso de las salas de ensayo, como de grabación, y ahora en la producción de contenido, no se han podido manejar bien ciertas situaciones de crecimiento en cuanto a la cantidad de clientes. En donde José comentó, que por la alta demanda que tuvo una temporada tuvo que realizar un convenio con otras salas de ensayo, lo que le llevo administrar más de una sala de ensayo.
- José destaco que no necesariamente los músicos que utilizan una sala de ensayo son profesionales, muchas personas tienen sus proyectos, como bandas y utilizan estos espacios para poder repasar, antes de tener un concierto. Por lo que, se puede llegar a tener la idea de Quito como una ciudad llena de varios artistas ocultos los cuales no necesariamente deben ser profesionales con título, como él lo define “*Un profesional es aquel el cual ensaya frecuentemente*” (Vease Anexo 1).
- La aplicación para desarrollar debe cumplir con lo siguiente: ser intuitivo, fácil de usar en cualquier dispositivo y que permita al usuario final poder encontrar una sala de ensayo en base a sus necesidades de ubicación, como el de poder publicarla si este tiene dicha sala. También se consideró la búsqueda por filtros, en base a los equipos que necesite el usuario (como instrumentos, audio, etc).
- Temas de seguridad, como el registro de usuarios para la gestión y búsqueda de las salas de ensayo. De esta manera, resguardar la seguridad del dueño de la sala y de los equipos musicales, presentes. Por lo que se debe limitar la información personal de los administradores de las salas de ensayo, como de los usuarios.
- José destaco su interés por el proyecto, al ser una gran idea, para dar una solución hacia los usuarios que necesitan encontrar una sala de ensayo la cual se

acomode a sus necesidades principales y sea acorde a la ubicación en donde este el usuario. Además, de permitir a los usuarios dueños de una sala de ensayo el poder dar a conocer sus espacios como poder gestionarlos.

3.1.2) Segunda Entrevista:

La segunda entrevista que se realizó fue al estudiante Josué Guerra (*Anexo 2*), en donde se detalló algunos aspectos importantes para el desarrollo de la aplicación. A continuación, para resumir la entrevista tomaremos en cuenta los siguientes puntos:

- **¿Cuál fue el objetivo de la entrevista a Josué Guerra?**

El principal objetivo de la entrevista con Josué fue el de conocer tanto la experiencia, como la perspectiva del usuario final, con respecto al uso de las salas de ensayo musicales, de igual forma el identificar los aspectos que más valora, como las dificultades que ha enfrentado como músico no profesional, además de sus expectativas frente a una aplicación que pueda cambiar el giro de negocio. En pocas palabras, a través de esta entrevista se buscó validar la utilidad del proyecto desde la visión de un músico activo (no profesional) y su relación con la tecnología y los procesos actuales acerca de la reserva de una sala.

- **¿Cuál es el perfil de Josué Guerra?**

Josué Guerra es un músico aficionado de 24 años, estudiante del universitario en Quito, guitarrista y parte de una banda emergente. Tiene experiencia en cuanto a buscar salas de ensayo para poder practicar con su banda. Además, tiene un buen conocimiento, o aceptación, sobre los beneficios que pueden existir al digitalizar servicios de alquiler de espacios.

- **¿Qué tipo de metodología se utilizó en la entrevista?**

Se utilizó una entrevista semiestructurada, permitiendo que el entrevistado responda de manera libre pero dentro de un conjunto definido de preguntas enfocadas en su experiencia personal. Esta metodología permitió obtener información detallada con respecto a sus preferencias, como necesidades con respecto al alquiler de las salas de ensayo.

- **¿Cuáles fueron los puntos clave de la entrevista?**

En cuanto a los puntos clave de la entrevista se identificó los siguientes:

- La mayoría de los jóvenes músicos encuentran salas por recomendación, coloquialmente “del boca a boca”, no por plataformas especializadas. A pesar de contar herramientas como las redes sociales, o Google Maps.
- Se logro identificar que los factores clave al momento de realizar una reserva son: precio, ubicación, equipos disponibles y facilidad para el proceso de la reserva. Por lo que, se debería mostrar una disponibilidad en tiempo real de los horarios disponibles de la sala, como una descripción detallada sobre los equipos.
- Existe un gran interés por parte de Josué Guerra en cuanto al desarrollo de esta aplicación. Debido a que, mediante esta aplicación podría darse a conocer nuevas salas de ensayo que posiblemente no se conocían sobre su existencia o tenían la oportunidad de darse a conocer por la comunidad artística dentro de la ciudad.
- Añadió también que, los usuarios valoran los sistemas de confianza con calificaciones y comentarios de otros usuarios, similar a las apps de servicios y aplicaciones. Lo que permite tener una mejor decisión al momento de escoger una sala de ensayo para su reserva, o de igual manera el usuario administrador pueda observar dichos comentarios y aplicar las respectivas mejoras hacia su espacio.

3.2 REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES

Con base en los temas tratados en ambas entrevistas, se logró realizar un alza de requerimientos, los cuales nos permitieron tener una mejor visión sobre lo que queremos llegar con el proyecto. Dando como resultado, la respectiva creación de una tabla con los requerimientos funcionales y no funcionales, para la presentación de un prototipo del proyecto llamado ‘QuitoJam’. Nombre el cual tiene la palabra ‘Quito’ como el espacio en donde podemos encontrar un espacio para la creación de música viviente o contemporánea, seguidamente de la palabra ‘Jam’, que es un término en Estados Unidos el cual en un contexto musical se refiere a mezclar sonidos o canciones.

3.2.1 Tabla de los Requerimientos Funcionales y No Funcionales

Tabla 6. Requerimientos Funcionales y No Funcionales

Requerimientos Funcionales y No Funcionales	
Funcionales	No Funcionales

Búsqueda de salas por nombre, ubicación y características.	Interfaz responsiva (móvil - web)
Visualización de un calendario, con horarios disponibles para reservar salas.	La plataforma debe cargar resultados en menos de 3 segundos.
Realización y cancelación de reservas.	Autenticación segura vía Supabase, garantizando la seguridad de datos personales.
Vista detallada de las salas de ensayo, con fotos, descripción, ubicación en el mapa).	Soporte para la autenticación por correo electrónico.
Registro/Inicio de sesión para usuarios clientes (músicos) y administradores (dueños de la sala).	Uso de lenguaje claro, directo y accesible para el público general.
Panel de gestión para administradores (CRUD de salas, horarios, reservas).	
Uso de geolocalización para autocompletar direcciones.	
Subir imágenes de las salas de ensayo.	
Posibilidad de añadir calificaciones y comentarios.	
Sistema de Pagos (simulado)	

Nota. Tabla desarrollada por el Autor.

Como podemos observar en la presente tabla, se deben desarrollar las funcionalidades clave, como lo son; la búsqueda de salas, la visualización de los detalles de las salas, la posibilidad de crear y gestionar dichas salas, como la selección de horarios para la reserva, entre las otras

funcionalidades descritas. Dichas características no solo facilitan la experiencia del usuario, sino que aseguran la flexibilidad, como la eficiencia para su uso en casos reales que pueda enfrentar la aplicación.

De igual manera, los requerimientos no-funcionales, tienen gran relevancia al asegurar el rendimiento óptimo, la seguridad, la accesibilidad, como la fiabilidad del proyecto. En donde, características como la carga rápida de la plataforma, la autenticación segura, además del uso de una interfaz adaptiva, permite que los usuarios puedan interactuar con la aplicación de manera fluida y sin interrupciones, mientras que la accesibilidad del lenguaje garantice una experiencia inclusiva aquellos usuarios los cuales no son técnicos o conocedores del tema.

La priorización se realizó tomando en cuenta el impacto directo en la experiencia del usuario, validado durante entrevistas y con base en los riesgos técnicos identificados. Posteriormente, los ajustes en los sprints se fundamentaron en la retroalimentación de los usuarios, garantizando que funcionalidades críticas como el filtrado por ubicación y la reserva por fechas se mantuvieran como prioridad frente a mejoras visuales.

Como conclusión, la realización de esta tabla nos permite tener claro el cumplimiento de los requerimientos para este prototipo, en base al alcance del proyecto y las necesidades encontradas dentro de los usuarios entrevistados, siendo esta tabla una representación sobre las mínimas funciones que debería cumplir la aplicación para su prototipo final. Además, cabe destacar que los presentes requerimientos fueron aprobados por los mismos usuarios entrevistados (ver Anexo 3), oficializando de igual manera su participación para las revisiones de los respectivos avances o sprints en la metodología SCRUM.

CAPÍTULO IV: DESARROLLO DEL PRIMER PROTOTIPO, MEDIANTE METODOLOGÍA SCRUM

4.1 DEFINICIÓN DE COMPONENTES CLAVE DE SCRUM

4.1.1 Roles de ‘SCRUM definidos:

Tabla 7. Roles de SCRUM definidos:

Nombre	Rol en SCRUM	Rol Funcional en la App
Jonathan Rosero	SCRUM Master/ Dev	Desarrollador Principal
José Torres	Product Owner	Usuario final de tipo Administrador (Dueño de la Sala)
Josué Guerra	Usuario Final	Usuario final de tipo Cliente (Músico)

Nota. Tabla desarrollada por el Autor

En la presente tabla, podemos observar los roles definidos para SCRUM, los cuales fueron asignados en base a las entrevistas anteriores y por el apoyo de los presentes para el proyecto ‘QuitoJam’. Por lo que, para los entregables se van a realizar reuniones coordinando un día de la semana, para la revisión, discusión de alguna mejora y la retroalimentación del proyecto.

4.1.2 HISTORIAS DE USUARIO

Tabla 8. Historias de Usuario

IDE	ROL	DESCRIPCIÓN	RAZÓN	CRITERIO DE ACEPTACIÓN	CONTEXTO
1	Usuario Administrador	Como dueño de una sala, quiero poder registrarme en base a un perfil especial, para publicar	Tener acceso exclusivo a herramientas las cuales permitan	Registro como administrador funcional. Formulario con campos clave.	Tener Login y Registro el cual permita la creación de un

		información, horarios y precios, de mis salas de ensayo.	gestionar y publicar mis salas de ensayo.	Apartado único de la aplicación para dueños de las salas.	usuario tipo 'Admin'.
2	Usuario Normal	Quiero poder registrarme e iniciar sesión, para poder visualizar las salas y realizar reservas.	El poder acceder a funciones básicas de la aplicación.	Formulario de registro para los usuarios de tipo 'cliente', los cuales al momento de registrarse	Tener un Login y registro el cual permita el ingreso, como registro del usuario de tipo 'cliente'.
3	Usuario Normal	Quiero poder ver una lista de salas disponibles, para decidir cuál reservar.	Explorar opciones de salas fácilmente antes de reservar.	Pantallas con tarjetas de salas, mostrando características clave	Desarrollar un contenedor (tipo tarjeta) para mostrar las características de la sala de ensayo.
4	Usuario Normal	Quiero poder buscar salas por nombre, características o ubicación.	Filtrar mejor según las preferencias.	Tener un apartado para realizar búsquedas con actualizaciones en tiempo real.	Implementación de búsquedas completas, en base a los caracteres o palabras clave.
5	Usuario Normal	Necesito ver los detalles de una sala específica, incluyendo mapa,	Conocer a detalle completo la	Contar con una Pantalla de detalles con imágenes,	Uso de herramientas de Flutter para la presentación

		horarios, restricciones si las hay y fotografías.	sala de ensayo	dirección, disponibilidad, descripción, entre otros.	de un mapa, además del diseño para la pantalla.
6	Usuario Normal	Quiero poder seleccionar fecha y hora para hacer una reserva.	Reservar una sala en un horario específico.	Tener un calendario para seleccionar la fecha de reserva, además de contar con una validación de disponibilidad antes de confirmar.	Implementar una lógica de verificación de conflictos de horarios.
7	Usuario Normal	Necesito poder ver mis reservas realizadas (pasadas y futuras). A modo de tener un historial de mi actividad.	Tener un control y seguimiento de las reservas realizadas.	Contar con una pantalla a modo de historial sobre reservas realizadas. Con algunos datos clave visibles (sala, fecha, estado).	Mediante la obtención de datos del backend. (Supabase RPC o REST).
8	Usuario Normal	Necesito el poder cancelar una reserva si ya no la necesito.	Liberar espacio y evitar confirmacion es en las reservas innecesarias.	Tener un botón para cancelar una reserva.	Requiere lógica en Supabase. Y una implementación en Flutter con un botón.

9	Usuario Administrador	Como dueño de una sala de ensayo, necesito poder crear nuevas salas de ensayo con descripción, ubicación, imágenes y precios.	Añadir nuevas ofertas fácilmente.	Contar con un formulario en donde los campos sean validados. Posibilidad de subir imágenes, y ubicación con ayuda de mapa.	Uso de Supabase storage, con servicio de conexión y funciones en Flutter con. dart
10	Usuario Administrador	Quiero editar o eliminar salas que ya no estén disponibles.	Poder mantener información actualizada de las salas.	Tener funcionalidades CRUD (actualizar y eliminar). Contar con una confirmación antes de eliminar.	Operaciones seguras con Supabase y validación.
11	Usuario Administrador	Como dueño de la sala de ensayo, quiero configurar los horarios disponibles para ensayos o grabaciones.	Para evitar reservas fuera del horario real, o de un horario no definido.	Contar con funcionalidades editables con rango de horario.	Mediante una UI para gestionar los horarios de las reservas disponibles en las salas de ensayo.
12	Usuario Administrador	Como dueño de la sala de ensayo quiero ver todas las reservas hechas en	Conocer el uso de mi espacio.	Pantalla tipo tabla o tarjetas con nombre del usuario, fecha y	Se puede usar un filtro en Supabase por

		mis salas ordenadas por fecha.		estado de la reserva.	sala_id y fecha.
13	Usuario Administrador	Como dueño, quiero subir fotos de mis salas y verlas en vista previa antes de guardar.	Mostrar mejor mi espacio y atraer clientes.	Constar con la opción de subir fotografías con vista previa.	Subida de imagen mediante un bucket de Supabase y referencia en la base de datos.
14	Usuario Normal	Quisiera poder tener personalización de mi perfil, poder editarlo cuando necesite cambiar cualquier aspecto de mi cuenta.	Poder cambiar de tipo de cuenta. Poder eliminar cuenta y todos los registros de esta misma.	Personalización de la cuenta con foto de perfil, como foto de perfil. botón para eliminar mi cuenta.	Implementación de foto de perfil. Bucket en Supabase para guardar la URL de la foto. Lógica para eliminación de la cuenta.
15	Usuario Normal	Me gustaría el poder buscar salas cercanas usando mi ubicación actual.	Poder saber que salas son las más cercanas en base a mi ubicación.	Constar con la lógica necesaria para el acceso a ubicación del usuario. Filtró por radio (ej. 5km, 10km).	Mediante la obtención de parámetros como Latitud Longitud por parte del usuario. Además de la creación de un

					apartado de búsqueda.
16	Usuario Normal	Necesito poder realizar pagos dentro de la aplicación, de esta manera la reserva esta más que segura.	Confirmar la reserva de manera segura.	Integración de un flujo visual de pago exitoso.	Este módulo se planifica para etapas avanzadas. Al entregar el prototipo solo se indicará una simulación del flujo y proceso para pagar.
17	Usuario Administrador	Quisiera que la aplicación fuera adaptable tanto para móvil, como para ordenador.	Poder utilizar la aplicación tanto en móvil, como en ordenador	Plataforma web, para la fácil adaptabilidad tanto en móviles, como en ordenadores.	Realizar el proyecto en Flutter Web, para conseguir adaptabilidad de este.
18	Usuario Administrador	Quiero recibir notificaciones, cada vez que se registre una nueva sala de ensayo.	Poder estar al pendiente de las nuevas reservas hechas dentro de la aplicación.	Notificaciones en tiempo real sobre la reserva de la sala de ensayo con detalles.	Integrar un servicio de Supabase y una API para él envió de correo electrónico, con dominio propio.
19	Usuario Cliente	Me gustaría que cada vez que pague dentro de la aplicación pueda	Para poder estar más interesado en seguir	Apartado de recompensas por fidelidad al usar la aplicación.	Utilizar el concepto de 'JamDollars', como moneda

acumular créditos para poder canjearlos para una reserva gratis, como también descuentos en las reservas.	utilizando la aplicación.	Como descuentos, reservas gratis, etc.	virtual para canjear recompensas.
---	---------------------------	--	-----------------------------------

Nota. Tabla desarrollada por el Autor.

Las 19 historias de usuario identificadas mediante las necesidades compartidas con los dos tipos de usuario, ya definidos en la tabla de roles, para el desarrollo del proyecto ‘QuitoJam’, nos permite estructurar de una mejor manera necesidades principales que deben ser cumplidas en este desarrollo de un primer prototipo. Cabe destacar que, dichas historias, son también producto del alza de requerimientos, con las entrevistas a los usuarios reales, las cuales cubren las necesidades del usuario tipo cliente el cual necesita ver opciones de espacios musicales para poder ensayar, con disponibilidad horaria. Como de igual manera, para el usuario de tipo administrador, el cual busca gestionar mejor sus salas de ensayo, como también el horario, ver la actividad de sus salas, y por último tener más alcance con usuario nuevos.

En conclusión, la inclusión de estas historias de usuario dentro del marco metodológico de SCRUM nos permite tener una mejor visión sobre el desarrollo del proyecto en base a las necesidades de los perfiles identificados para el uso de la aplicación web. Lo que nos lleva, a poder identificar la complejidad y prioridad que van a tener dichas historias de usuario en base a la creación del ‘Product Backlog’.

4.1.3 Product BackLog

El presente Backlog, representa una lista de funcionalidades necesarias para el desarrollo del proyecto ‘QuitoJam’, en cual se define a partir de las historias de usuario previamente levantadas. Por lo que, cada ítem del BackLog ha sido clasificado con un código, un nombre descriptivo, su nivel de complejidad y su prioridad asignada, permitiendo tener una mejor organización del proyecto.

Tabla 9. Product Backlog

Código	NOMBRE	COMPLEJIDAD	PRIORIDAD
---------------	---------------	--------------------	------------------

001.1	Pantalla de login/registro para el usuario cliente y administrador.	Media	Alta
001.2	Validación de credenciales con Supabase-Auth.	Alta	Alta
001.3	Pantalla de visualización de salas de ensayo con información básica.	Media	Media
001.4	Pantalla para usuarios invitados con visualización y funcionalidades limitadas.	Media	Media
002.1	Menú desplegable según tipo de usuario (invitado, cliente, administrador).	Media	Alta
002.2	CRUD completo para la administración y gestión de las salas (crear, editar, eliminar, visualizar).	Alta	Alta
002.3	Cards de presentación de salas de ensayo para clientes, invitados y administradores.	Media	Media
002.4	Pantalla para la gestión de horarios disponibles por sala, en el apartado del administrador.	Alta	Alta

003.1	Pantalla de detalles de sala: descripción, ubicación, mapa, galería de imágenes, etc.	Media	Alta
003.2	Subida de imágenes de las salas con previsualización.	Media	Media
004.1	Selección de fecha y hora para la reserva.	Alta	Alta
004.2	Validación de disponibilidad de los horarios antes de ejecutar la reserva.	Alta	Alta
004.3	Cancelación de reservas con confirmación y motivo.	Media	Alta
004.4	Vista de historial de reservas para el usuario (pasadas y futuras).	Media	Media
005.1	Panel de reservas recibidas por administrador.	Media	Alta
005.2	Resumen administrativo: reservas en curso, canceladas, reembolsadas, total.	Media	Alta

005.3	Gráficos estadísticos, para la visualización de la actividad de las salas.	Alta	Media
006.1	Autocompletado de dirección por geolocalización en formulario de creación de sala.	Alta	Media
006.2	Búsqueda de salas por nombre, ubicación o características.	Alta	Alta
006.3	Búsqueda de salas por cercanía usando la ubicación del dispositivo.	Alta	Media
007.1	Eliminación de cuenta del usuario desde pantalla de perfil.	Media	Media
007.2	Actualización de datos de perfil (foto, nombre, etc.).	Media	Media
007.3	Función automática de actualización de estado de reservas (en curso, finalizada).	Alta	Alta
008.1	Lógica de control para evitar reservas fuera del horario establecido.	Alta	Alta

008.2	Seguridad en acceso a pantallas y rutas según tipo de usuario.	Alta	Alta
008.3	Integración (simulada) de confirmación visual de pago.	Media	Media
009.1	Sistema de calificación y reseñas por sala de ensayo.	Media	Media
009.2	Notificaciones en tiempo real sobre las reservas hechas en la sala de ensayo.	Media	Alta
009.3	Implementación de JamDollars, para promociones incentivando para incentivar el uso frecuente de la App.	Media	Media

Nota. Tabla desarrollada por el Autor

El presente BackLog, nos proporciona una visión clara del alcance del proyecto, facilitando la planificación de los sprints, como la asignación progresiva de las tareas a realizar. La presente clasificación de actividades nos permite identificar las funcionalidades clave y nos permite identificar la dificultad que estas tienen para su desarrollo ágil. Asegurando que la implementación de estos módulos dentro de la aplicación web pueda responder a las necesidades reales de los usuarios, tanto músicos, como administradores de las salas de ensayo.

4.1.4 Definición de Sprints en base al Product Backlog (Sprint Planning)

Tabla 10. *División del Product Backlog en Sprints*

Sprint 1	Sprint 2	Sprint 3	Sprint 4
001.1	002.2	004.3	007.1
001.2	002.3	004.4	007.2
001.3	002.4	005.1	009.1
001.4	003.1	005.2	009.2
002.1	003.2	005.3	009.3
008.2	004.1	007.3	
	004.2	008.1	
	006.1	008.3	
	006.2		
	006.3		

Nota. Tabla desarrollada por el Autor

La presente tabla incluye la distribución de los ítems entregables del ‘Product Backlog’ organizados en 4 sprints, los cuales son definidos bajo la metodología de SCRUM, lo que se conoce como ‘Sprint Planning’. En donde, se considera tanto la complejidad técnica, como la prioridad del proyecto. Por lo que, se procuró mantener un balance progresivo de la carga de trabajo por sprint. En conclusión, dicha tabla está basada en cuanto a las necesidades presentadas por parte de los usuarios, desde las historias de usuario, hasta la definición del Product BackLog.

4.2 SPRINT 1 – Autenticación y Accesos

4.2.1 Introducción

Nombre del Sprint: Autenticación Usuarios/Administradores y exploración inicial.

Duración del Sprint: 05/05/2025 – 05/09/2025.

Equipo SCRUM:

- **SCRUM Master/Dev:** Jonathan Rosero
- **Product Owner:** José Torres.
- **Usuario Final:** Josué Guerra

4.2.2 Objetivos del Sprint 1:

- Diseñar y desarrollar la interfaz con conexión a la BDD el login/registro para los usuarios normales, como usuarios administradores, incluyendo validaciones básicas, y estableciendo la base funcional para futuros flujos de la aplicación.
- Se busca garantizar que la aplicación pueda identificar correctamente los roles de los usuarios, para que se pueda iniciar con la navegación clara para cada tipo de usuario.

4.2.3 Historias de Usuario Asociadas al Sprint 1

Tabla 11. Historias de Usuario Asociadas al Sprint

IDE	ROL	DESCRIPCIÓN	RAZÓN	CRITERIO DE ACEPTACIÓN	CONTEXTO
1	Usuario Administrador	Como dueño de una sala, quiero poder registrarme en base a un perfil especial, para publicar información, horarios y precios, de mis salas de ensayo.	Tener acceso exclusivo a herramientas las cuales permitan gestionar y	Registro como administrador funcional. Formulario con campos clave. Apartado único de la aplicación para dueños de las salas.	Tener Login y Registro el cual permita la creación de un usuario tipo 'Admin'.

			publicar mis salas de ensayo.		
2	Usuario Normal	Quiero poder registrarme e iniciar sesión, para poder visualizar las salas y realizar reservas.	El poder acceder a funciones básicas de la aplicación.	Formulario de registro para los usuarios de tipo 'cliente', los cuales al momento de registrarse	Tener un Login y registro el cual permita el ingreso, como registro del usuario de tipo 'cliente'.
3	Usuario Normal	Quiero poder ver una lista de salas disponibles, para decidir cuál reservar.	Explorar opciones de salas fácilmente e antes de reservar.	Pantallas con tarjetas de salas, mostrando características clave	Desarrollar un contenedor (tipo tarjeta) para mostrar las características de la sala de ensayo.
17	Usuario Administrador	Quisiera que la aplicación fuera adaptable tanto para móvil, como para ordenador.	Poder utilizar la aplicación tanto en móvil, como en ordenador	Plataforma web, para la fácil adaptabilidad tanto en móviles, como en ordenadores.	

Nota. Tabla desarrollada por el Autor

Para el presente sprint, se identificaron las tres primeras historias de usuario las cuales corresponden al desarrollo de las pantallas de Login y de Registro, además sirven para entender de mejor manera la principal lógica que queremos implementar dentro de la aplicación.

4.2.4 Tareas del Sprint 1:

Tabla 12. Tareas del Sprint 1

CÓDIGO	NOMBRE DE LA TAREA	TIPO DE USUARIO	ESTADO
001.1	Pantalla de login/registro para el usuario cliente y administrador.	Cliente/Admin	COMPLETADO
001.2	Validación de credenciales con Supabase-Auth.	Cliente/Admin	COMPLETADO
001.3	Pantalla de visualización de salas de ensayo con información básica.	Todos	COMPLETADO
001.4	Pantalla para usuarios invitados con visualización y funcionalidades limitadas.	Invitado	COMPLETADO
002.1	Menú desplegable según tipo de usuario (invitado, cliente, administrador)	Todos	COMPLETADO
008.2	Seguridad en acceso a pantallas y rutas según tipo de usuario	Todos	COMPLETADO

Nota. Tabla desarrollada por el Autor

En base al product backlog y a la división de las tareas por sprint, se especificó las tareas a realizar para este sprint en específico. Se maneja el tipo de usuario, por lo que en la aplicación tenemos tres tipos de usuarios:

- **Invitado:** Es aquel usuario el cual ingresa por primera vez a la pantalla, el cual solo puede observar las salas disponibles, pero si quiere realizar alguna acción de reserva este tiene que obligatoriamente registrarse.

- **Cliente:** Son aquellos usuarios los cuales no se registran como administradores, porque lo que quieren es realizar la búsqueda y la reserva de las salas de ensayo.
- **Administrador (Admin):** Es aquel usuario el cual se registra como administrador, para que pueda publicar, editar y observar la actividad de sus salas de ensayo publicadas.

Además, se realizó la verificación de las tareas en base al cumplimiento del sprint en el plazo establecido.

4.2.5 Modelo de Datos para el Sprint 1

En la base de datos, se utilizó el siguiente modelo de tabla para el almacenamiento de los datos personales de los usuarios registrados. Cumpliendo con aspectos claves para establecer relaciones con otras tablas, mediante su identificador único 'id_usuario', consta de campos los cuales serán útiles dentro del sistema, para más detalles de los campos junto al *Anexo 16*.

Figura 4. Tabla 'usuario'

Field Name	Data Type
id_usuario	uuid
nombre	varchar
cedula	varchar
correo	varchar
telefono	varchar
tipo_usuario	varchar
fecha_creacion	timestamp
foto_perfil	text

Nota. Captura de la tabla 'usuario', obtenida de Supabase por el Autor 2025

4.2.6 Actividades Técnicas

Frontend (Flutter)

En el presente desarrollo, cabe destacar el uso de la arquitectura en capas, o conocida como 'Clean Architecture', de manera simplificada al guardar los distintos archivos '.dart' en

carpetas como ‘/model’, ‘/screens’, ‘/services’, ‘/widgets’ y ‘/styles’. Permitiendo así un mejor control y orden sobre los mismos.

- Creación de las pantallas ‘screens/login_screen.dart’, ‘screens/register_screen.dart’, con integración hacia Supabase.
- Implementación de validaciones para los formularios, con mensajes de error hacia el usuario.
- Creación del archivo ‘services/supabase_service.dart’, para la conexión y realización de funciones para el backend. Configuración de este para agregar funciones de inicio sesión, validación de credenciales, y agregación de cuentas.
- Desarrollo de la lógica de navegación condicional basándose en el rol de usuario, creación del widget ‘widgets/custom_drawer.dart’ para personalizar el menú desplegable para cada tipo de sesión.
- Creación de la pantalla ‘screens/users/show_room.dart’, como pantalla de inicio para la sección de usuarios de tipo cliente, mostrando las cards por el tipo de usuario.
- Desarrollo de los archivos ‘widgets/salaCardAdmin.dart’ y ‘widget/salaCardUsuario.dart’, para armar la presentación resumida o de tipo ‘card’ de las salas de ensayo, enfocado en como serian visualizadas por cada tipo de usuario, para la presente entrega se mostró información quemada de las salas de ensayo.
- Implementación de restricciones para los usuarios que ingresan por primera vez hacia la aplicación en la pantalla de ‘screens/users/show_room.dart’, mediante el uso de validaciones de tipo de usuario.
- Creación del widget ‘widget/base_screen.dart’ para mantener un diseño uniforme con header y footer en la mayoría de las pantallas navegables.
- Creación del archivo tipo model de ‘usuario/usuario_model.dart’ para representar la estructura del usuario dentro del sistema. Se lo utiliza al momento de iniciar sesión, registrarse, al mostrar la información del perfil y, por último, proteger las rutas de navegación según el rol.

Backend (Supabase)

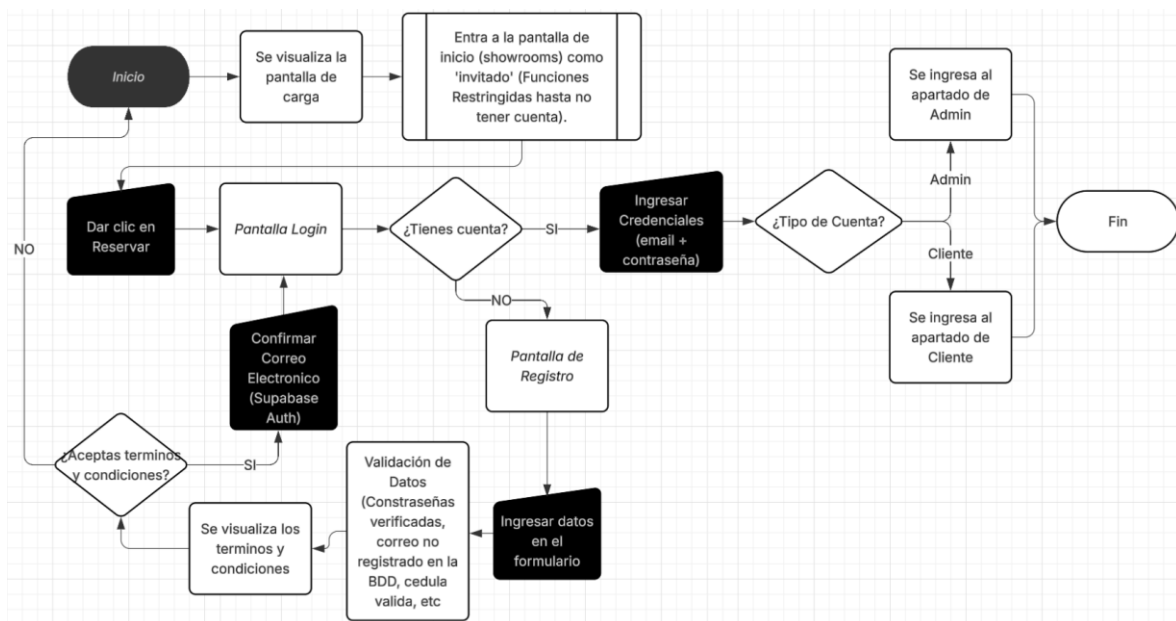
- Configurar la funcionalidad de un correo electrónico para la activación y verificación de la cuenta.
- Creación de la tabla ‘**usuario**’, verificar que los campos estén correctamente implementados.

- Utilizar tipo de dato 'uuid', para una mejor seguridad al almacenamiento de los usuarios en la BDD.

La adopción de Supabase como backend se fundamentó en su integración nativa con autenticación por roles y gestión de datos a través de PostgREST, lo que permitió reducir significativamente la complejidad arquitectónica respecto a soluciones tradicionales con NodeJS o Django REST. Esta decisión técnica facilitó un desarrollo más ágil y un modelo de seguridad robusto, esencial en este sprint donde se establecieron las bases de autenticación y separación de vistas por roles.

4.2.7 Diagramas de Flujo Correspondientes al Sprint 1

Figura 5. Diagrama de Flujo Registro/Login



Nota. Diagrama realizado por el autor en Lucidchart, 2025

El presente diagrama representa el flujo inicial al momento de interactuar por primera vez dentro de la aplicación. Por lo que, se muestra principalmente el flujo para la creación de cuenta para el usuario, como también la redirección para el apartado de cada usuario.

4.2.8 RESULTADOS OBTENIDOS

Podemos evidenciar en el *Anexo 4 y 5*, que la revisión fue aprobada del presente sprint con los miembros del equipo, por lo que se logró cumplir con todos los objetivos establecidos en las respectivas pantallas desarrolladas (*véase Anexo 6*). El presente sprint se enfocó en el flujo de

registro y autenticación para ambos tipos de usuarios, en donde se incluyó la lógica para la separación de las vistas en base a los roles. Los usuarios pudieron ingresar sin ningún tipo de errores, lograron visualizar cada una de la pantalla inicial por tipo de usuario, en este sprint se utilizaron datos quemados para poder presentar las salas de ensayo. Permitiendo, una mejor interacción y expectativa visual para los usuarios, con respecto a las pantallas a modo de que se pueda entender la diferencia en el acceso por cada tipo de usuario.

Lo que nos permite poder tener dos tipos de acceso diferenciados, los cuales nos sirven para la interacción de cada uno de los usuarios previstos que tenemos para el funcionamiento de nuestra aplicación. En conclusión, se logró cumplir con todos los objetivos planteados para el sprint. Además, de brindar algunos extras como lo fue las pantallas de confirmación para la creación de las cuentas, y algunos mensajes que permiten a la aplicación ser más intuitiva, además de las validaciones en los campos de registro, como de login.

4.2.9 Observaciones y Recomendaciones de los Usuarios (Sprint Review)

Durante la validación del presente Sprint, se realizaron comentarios positivos, por ejemplo, el usuario cliente 'Josué Guerra' expresó su satisfacción al contar con una interfaz clara y una facilidad para poder registrarse. Además, de contar con todas las validaciones disponibles en los campos del registro de la cuenta.

Por otro lado, el usuario de tipo administrador José Torres consideró que el botón para el registro de los administradores fue claro, además de que le agrado el hecho de poder confirmar la identidad del usuario mediante un mensaje de correo electrónico. Permitiendo de esta manera manejar ámbitos de seguridad al momento de validar a los usuarios. Lo que se pidió verbalmente, fue el de manejar la temática de los colores propuestos en toda la aplicación, además de manejar validación de los campos dentro del formulario.

4.2.10 Conclusiones del Sprint 1 (Sprint Retrospective)

El sprint 1 fue de mucho impacto, ya que permitió establecer la base estructural para la navegación, como la autenticación de usuarios. En donde, se cumplieron exitosamente todos los objetivos planteados desde un principio, además de obtener una retroalimentación clara por parte de los usuarios objetivo del proyecto. Por lo que, en esta etapa de desarrollo la correcta definición de los roles diferenciados permitió asegurar rutas y adaptar la experiencia a cada tipo de usuario. Por otro lado, también con la creación ya de este pequeño flujo, como la presentación previa de las pantallas para ambos tipos de usuario, se estableció ya un estilo para

estas pantallas, con el manejo de colores oscuros y de contrastes altos, diseño que llamo la atención de los usuarios por ser de cierta manera único.

Finalmente, podemos destacar que el cumplimiento de todas las actividades previstas y aprobadas por los usuarios nos permitieron avanzar hacia el desarrollo del Sprint 2 con una base estable ya definida en la aplicación.

4.3 SPRINT 2 – Búsqueda, Reserva y Gestión de las Salas de Ensayo

4.3.1 Introducción

Nombre del Sprint: Búsqueda, reserva y Gestión de Salas de Ensayo.

Duración del Sprint: 05/10/2025 – 05/23/2025

Equipo SCRUM:

- **SCRUM Master/Dev:** Jonathan Rosero
- **Product Owner:** José Torres.
- **Usuario Final:** Josué Guerra

4.3.2 Objetivos del Sprint 2:

- Crear una pantalla la cual pueda mostrar los detalles de la sala, permitiendo de esta manera observar la información completa de la misma, como la ubicación en un mapa, los precios por hora, las imágenes y la descripción.
- Realizar el flujo completo para la creación de una reserva, en donde el usuario pueda seleccionar la sala, seleccionar los horarios disponibles y finalmente confirmar y realizar el pago exitoso. incluyendo las validaciones necesarias para evitar conflictos con la disponibilidad de los horarios.
- Permitir realizar operaciones de CRUD al usuario administrador para la creación, edición, visualización y eliminación de las salas de ensayo.
- Establecer las bases para la búsqueda inteligente de las salas, por nombre, por cercanía, además de agregar filtros de precios y capacidad de personas.
- Permitir una pantalla enfocada para la búsqueda de las salas de ensayo, que disponga de filtros de búsqueda necesarios, como funcionalidades para mostrar las salas de ensayo más cercanas al usuario.

4.3.3 Historias de Usuario Asociadas al Sprint 2:

Tabla 13. Historias de Usuario Asociadas al Sprint 2

IDE	ROL	DESCRIPCIÓN	RAZÓN	CRITERIO DE ACEPTACIÓN
4	Usuario Normal	Quiero poder buscar salas por nombre, características o ubicación.	Filtrar mejor según las preferencias.	Tener un apartado para realizar búsquedas con actualizaciones en tiempo real.
5	Usuario Normal	Necesito ver los detalles de una sala específica, incluyendo mapa, horarios, restricciones si las hay y fotografías.	Conocer a detalle completo la sala de ensayo	Contar con una Pantalla de detalles con imágenes, dirección, disponibilidad, descripción, entre otros.
6	Usuario Normal	Quiero poder seleccionar fecha y hora para hacer una reserva.	Reservar una sala en un horario específico.	Tener un calendario para seleccionar la fecha de reserva, además de contar con una validación de disponibilidad antes de confirmar.
9	Usuario Administrador	Como dueño de una sala de ensayo, necesito poder crear nuevas salas de ensayo con descripción, ubicación, imágenes y precios.	Añadir nuevas ofertas fácilmente.	Contar con un formulario en donde los campos sean validados. Posibilidad de subir imágenes, y

				ubicación con ayuda de mapa.
10	Usuario Administrador	Quiero editar o eliminar salas que ya no estén disponibles.	Poder mantener información actualizada de las salas.	Tener funcionalidades CRUD (actualizar y eliminar). Contar con una confirmación antes de eliminar.
11	Usuario Administrador	Como dueño de la sala de ensayo, quiero configurar los horarios disponibles para ensayos o grabaciones.	Para evitar reservas fuera del horario real, o de un horario no definido.	Contar con funcionalidades editables con rango de horario.
15	Usuario Normal	Me gustaría el poder buscar salas cercanas usando mi ubicación actual.	Poder saber que salas son las más cercanas en base a mi ubicación.	Constar con la lógica necesaria para el acceso a ubicación del usuario. Filtró por radio (ej. 5km, 10km). Además de la creación de un apartado de búsqueda.

Nota. Tabla desarrollada por el Autor, 2025

4.3.4 Tareas del Sprint 2:

Tabla 14. Tareas del Sprint 2

CÓDIGO	NOMBRE DE LA TAREA	TIPO DE USUARIO	ESTADO
002.2	CRUD completo para la administración de salas de ensayo (crear, editar, eliminar, visualizar).	ADMIN	COMPLETADO
002.3	Cards de presentación de salas de ensayo para clientes, invitados y administradores.	CLIENTE/ADMIN	COMPLETADO
002.4	Pantalla para la gestión de horarios disponibles por sala, en el apartado del administrador.	ADMIN	COMPLETADO
003.1	Pantalla de detalles de sala: descripción, ubicación, mapa, galería de imágenes, etc.	CLIENTE/ADMIN	COMPLETADO
003.2	Subida de imágenes de las salas con previsualización.	ADMIN	COMPLETADO
004.1	Selección de fecha y hora para la reserva.	CLIENTE	COMPLETADO
004.2	Validación de disponibilidad de los horarios antes de ejecutar la reserva.	CLIENTE	COMPLETADO
006.1	Autocompletado de dirección por geolocalización en formulario de creación de sala.	ADMIN	COMPLETADO
006.2	Búsqueda de salas por nombre, ubicación o características.	CLIENTE	COMPLETADO

006.3	Búsqueda de salas por cercanía usando la ubicación del dispositivo.	CLIENTE	COMPLETADO
-------	---	---------	------------

Nota. Tabla desarrollada por el Autor, 2025

4.3.5 Modelo de Datos para el Sprint 2

Tabla ‘saladeensayo’:

Empezando con la tabla ‘saladeensayo’, la cual nos permite almacenar toda la información necesaria para desplegarla en la aplicación. Dicha tabla constituye a la estructura principal del sistema, para poder representar a cada una de las salas de ensayo disponibles dentro de la plataforma. Por lo que contiene los atributos necesarios para su visualización, ubicación geográfica, algunas características principales y su relación con el administrador de la sala, para más detalles de los atributos, mirar el *Anexo 16*.

Figura 6. Tabla ‘saladeensayo’

saladeensayo	
capacidad_maxima	int4
restricciones	text
nombre_sala	varchar
ubicacion	text
latitud	numeric
longitud	numeric
precio_por_hora	numeric
descripcion	text
id_propietario	uuid
id_sala	int4
fecha_registro	timestamp
imagenes	_text
fts	tsvector

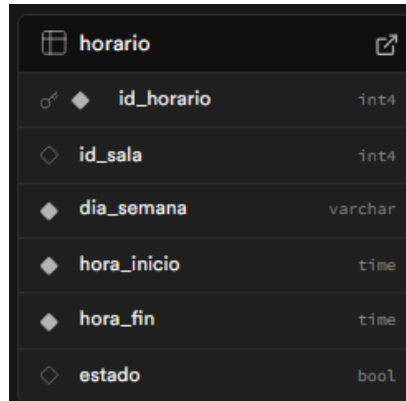
Nota. Captura obtenida de Supabase por el Autor, 2025

Tabla ‘horario’

La tabla horaria es la que permite a los usuarios administradores poder definir los días y los horarios disponibles por día, para que la sala de ensayo pueda aceptar nuevas reservas de los usuarios clientes. Por lo que, el propósito principal de la implementación de esta tabla fue para

que se permita que cada sala pueda tener múltiples registros en cuanto a sus horarios, lo que da flexibilidad a los usuarios administrador de determinar las horas que ellos quieran, en el día que ellos gusten sin romper la normalización de los datos, para más detalles de los atributos, mirar el *Anexo 16*.

Figura 7. *Tabla 'horario'*



The image shows a screenshot of a Supabase table definition for the 'horario' table. The table has the following columns and data types:

Column Name	Data Type
id_horario	int4
id_sala	int4
dia_semana	varchar
hora_inicio	time
hora_fin	time
estado	bool

Nota. Captura obtenida de Supabase por el Autor, 2025

Tabla 'reserva'

Es la que representa cada acción en la que un usuario de tipo cliente reserva una sala de ensayo, dentro de un intervalo de tiempo definido. Por lo que, este tipo de registro es clave si queremos manejar la disponibilidad, validaciones de solapamiento, además de manejar aspectos de visualización sobre la información de las reservas recibidas para el apartado administrador, como las reservas enviadas o solicitadas por parte del usuario cliente. Por último, esta entidad maneja conexión con las tablas 'usuario' (el cual es quien reserva o recibe reservas) y 'saladeensayo' (correspondiendo a la sala en la cual se realiza la reserva). Para más detalles de los atributos revisar el *Anexo 16*.

Figura 8. *Tabla 'reserva'*

Column Name	Data Type
id_reserva	int4
id_usuario	uuid
id_sala	int4
fecha_inicio	timestamp
fecha_fin	timestamp
estado	varchar
comentarios	text
fecha_reserva	timestamp
cancelada	bool

Nota. Captura obtenida de Supabase por el Autor, 2025

4.3.6 Actividades Técnicas

Frontend (Flutter)

- Desarrollo de los archivos `‘.dart’` de `‘models/sala_model.dart’`, `‘models/horario_model.dart’`, `‘models/pago_model.dart’`, `‘models/reserva_model.dart’`, para el manejo de los campos de dichas tablas en la base de datos dentro del desarrollo del frontend.
- Utilización de `‘flutter_map’`, un wrapper para mostrar y navegar por mapas dentro de la aplicación, además del uso de la API `‘Geocoding’` utilizado para obtener las coordenadas de ubicación del usuario.
- Desarrollo de la pantalla `‘screens/admins/agregar_editarSalas.dart’`, para la creación del formulario tanto para agregar una nueva sala de ensayo, como editar los campos, si esta sala esta creada. Implementando validaciones para que no se ingresen datos erróneos. Además, de agregar un botón para autocompletar la ubicación del usuario y mostrarla en el mapa.
- Desarrollo de las card para la presentación de las salas de ensayo hacia los usuarios, pese a que en el anterior sprint ya se crearon los archivos, en este sprint implementamos el modelo de la tabla, y funciones en `‘supabase_services.dart’`, para que las cards sean dinámicas con la información reflejada en la base de datos, manteniendo el estilo de presentación de estas.
- Desarrollo de la pantalla `‘screens/admins/gestionar_horarios.dart’`, para gestionar los horarios por sala, se puede acceder a esta pantalla mediante un icono de reloj en la parte inferior del card de la sala (`salaCardAdmin`).

- Desarrollo de la pantalla **'screens/users/detalle_sala.dart'**, dicha pantalla se encarga de mostrar al usuario todos los detalles disponibles de la sala de ensayo, además de incluir un botón de reservar al fondo para iniciar el proceso de reserva.
- Desarrollo de la pantalla **'screens/users/reserva_form_screen.dart'**, para la selección de horario de la reserva realizada por el usuario cliente y con la opción de agregar un comentario para detallar la reserva o cualquier requisito del usuario. En esta pantalla, se utiliza dos funciones desde **'supabase_service'** para el control de conflicto, tanto de los horarios, como el de las reservas.
- Desarrollo de la pantalla **'busqueda.dart'**, la cual permite al usuario buscar salas de ensayo en base a filtros de precio, capacidad de personas, mediante la interacción del mapa, en base a las salas más cercanas hacia el usuario y por una búsqueda en base a los caracteres de la sala, puede ser por nombre, como por detalles.
- Se crearon los archivos **'services/envservice.dart'**, el cual maneja la carga de variables desde un archivo JSON (el cual contiene al calve de la API de Google Geocoding), **'services/geocoding_service.dart'** este archivo es encargado de convertir en coordenadas geográficas mediante el uso de la API de Google Maps, lo que permite transformar una dirección escrita por el usuario en coordenadas geográficas que luego se despliegan en el mapa. Además, se creó el archivo **'services/location_service.dart'** para manejar la geolocalización en tiempo real, obteniendo la ubicación actual del usuario, mediante los servicios de ubicación del dispositivo. Por último, el archivo **'ubicación_helper.dart'** responsable de realizar geocodificación inversa utilizando 'OpenStreetMap' a través de la API 'Nominatim', el cual convierte las coordenadas (latitud y longitud) en una dirección legible para el usuario, debido a que la API de Google limitaba o no enseñaba una información legible al usuario.

Backend (Supabase)

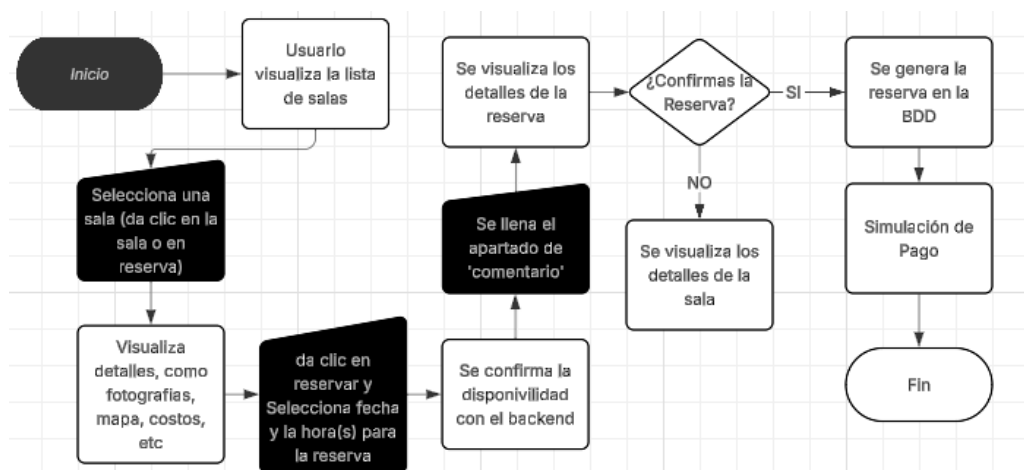
- Se realizó la creación de las tablas; **'saladeensayo'**, **'horario'**, **'reserva'**, con sus respectivas relaciones, incluyendo a la tabla **'usuario'**.
- Creación de la función **'salas_cercanas'** en Supabase, el cual devuelve una lista de salas cercanas a un usuario, dado que proporciona latitud y longitud del usuario, además de un radio de búsqueda en kilómetros. En donde se utiliza la función **'earth_distance'** para calcular la distancia entre el usuario y las salas de ensayo.

- Creación de los buckets de ‘perfil’ y ‘salas’, para almacenar y generar una ‘url’ de la imagen, dicha url se genera y guarda en la base de datos. El bucket ‘perfil’ permite almacenar una imagen (correspondiendo a la foto del perfil) y el bucket ‘salas’ permite guardar una lista de imágenes de la sala.
- Para el manejo de reservas y filtrado por geolocalización, se optó por Supabase Functions utilizando earth_distance de PostgreSQL, lo que simplificó cálculos de distancias y mantuvo la lógica del lado del servidor, reduciendo el tráfico y los tiempos de respuesta. Esta decisión técnica se justifica para mantener la consistencia y escalabilidad del sistema.

Durante este sprint, Supabase destacó al ofrecer capacidades avanzadas directamente en la base de datos PostgreSQL, como el uso de la función earth_distance para la proximidad geográfica mediante la RPC salas_cercanas. Esto permitió realizar cálculos de distancia entre el usuario y las salas en el servidor, reduciendo la transferencia de datos y evitando complejos cálculos en el cliente Flutter. Asimismo, los buckets de almacenamiento de Supabase simplificaron la gestión de imágenes de perfil y de salas, al generar URLs públicas seguras que se integraron fácilmente en los modelos del frontend, manteniendo la consistencia y reduciendo la necesidad de servidores intermediarios.

4.3.7 Diagramas de Flujo del Sprint 2

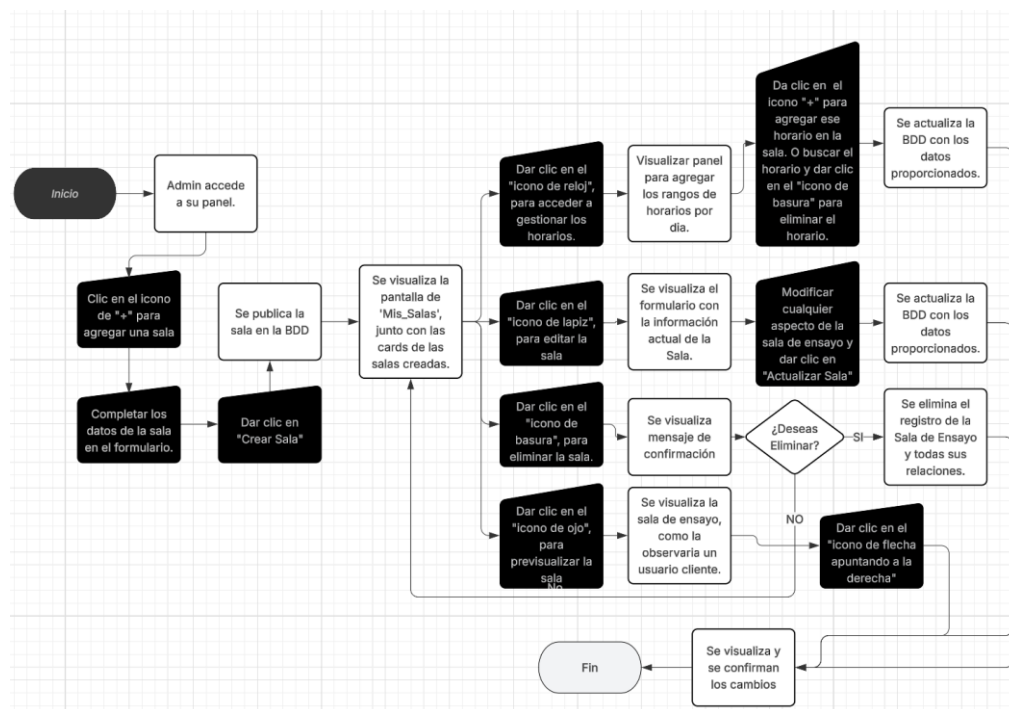
Figura 9. Diagrama de Flujo ‘Reserva de una Sala de Ensayo’



Nota. Diagrama realizado por el autor en Lucidchart, 2025

El presente diagrama de flujo representa las acciones que deben de ser realizadas por parte del usuario cliente, para poder realizar, como confirmar una reserva de una sala de ensayo. Por lo que, se representó detalladamente cada una de las acciones y sus consecuencias en el flujo.

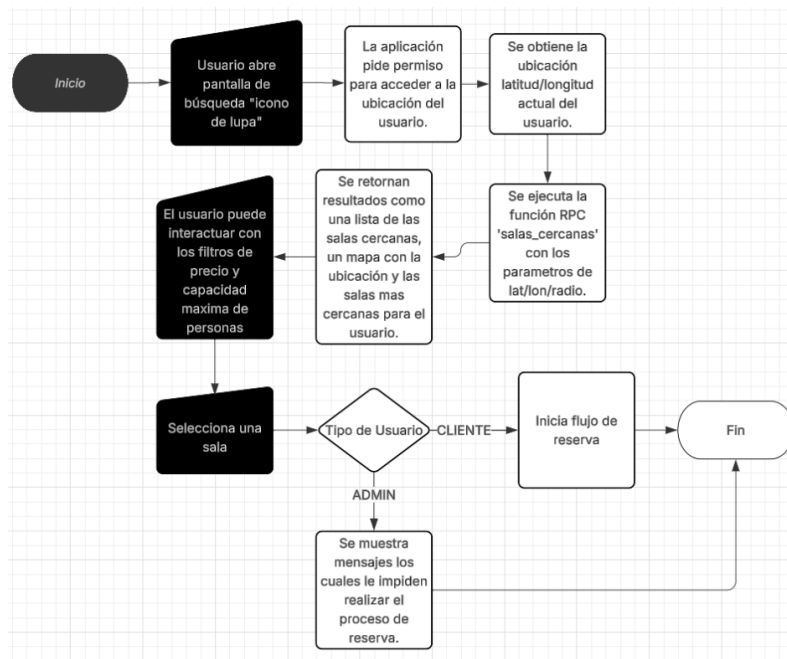
Figura 10. Diagrama de Flujo ‘Creación y Gestión de una Sala de Ensayo



Nota. Diagrama realizado por el autor en Lucidchart, 2025

En el presente flujo se enfocó en presentar las acciones requeridas y la navegación que tendría un usuario de tipo administrador para gestionar con funcionalidades CRUD, desde la creación, edición, gestión de horarios, y la visualización de la sala de ensayo. Permitiendo al usuario administrador tener una experiencia más profesional al momento de administrar su sala de ensayo dentro de la aplicación web.

Figura 11. Búsqueda de Salas por Geolocalización



Nota. Diagrama realizado por el autor en Lucidchart, 2025

El presente flujo nos ayuda a obtener un mejor panorama sobre la interacción que tiene un usuario con el apartado de búsqueda, se lo diseño como una pantalla individual para enfocarse solamente en búsqueda, a diferencia de otras pantallas que solo son para mostrar las salas de ensayo. Además, de mejorar la interacción de búsqueda hacia el usuario ofreciendo la posibilidad de filtrar las salas de ensayo.

4.3.8 Resultados Obtenidos (Sprint Review)

Dentro de la revisión de los entregables para este sprint, y de las pruebas de los respectivos usuarios, podemos observar dentro del *anexo 7* y *8*, que se marcaron todas las casillas de cumplimiento de los productos entregados. En donde, satisfactoriamente logramos cumplir con los objetivos principales de este sprint, los cuales fueron el entregar el flujo casi completo para agendar las reservas por parte del usuario cliente, como la pantalla o el apartado de la búsqueda de las salas de ensayo por proximidad y filtros. Por otro lado, al entregar el apartado administrativo para realizar acciones de tipo CRUD sobre las salas, se observó un buen desempeño aceptable al momento de ajustar la sala a la disponibilidad por parte del dueño (administrador).

En el *anexo 9*, podemos observar los productos entregados en este sprint corresponden a las pantallas (en formato móvil) las cuales se encuentran dinámicas con respecto a la información reflejada en la base de datos. Tanto para el flujo de reserva por parte del usuario cliente, además

el apartado de gestión sobre las salas de ensayo por parte del usuario tipo administrador, sin olvidarnos de la pantalla de búsqueda de las distintas salas. Por otro lado, observamos la consistencia en cuanto al diseño implementado durante toda la aplicación, manteniendo márgenes y colores similares.

4.3.9 Observaciones y Recomendaciones (Sprint Review)

Al tener todos los requerimientos solicitados aprobados por parte de ambos usuarios, en la sesión del sprint 2, no hubo muchos comentarios de mejora o cambios de la aplicación, el de completar el flujo de reserva con la integración del pago que se lo realizará en la entrega del siguiente sprint. En donde, por parte de José Torres se mencionó el tema de los datos personales del administrador, los cuales no deben ser visibles con facilidad, se explicó que solamente se comparten como dato personal del administrador el número telefónico del dueño de la sala, esto será mejor contemplado al momento de realizar el siguiente sprint. Josué Guerra, destaco lo bien implementado que están los mapas dentro de las pantallas, además de los filtros de búsqueda.

4.3.10 Conclusión del Sprint (Sprint Restrospective)

Como conclusión del sprint 2, podemos destacar que se consolidaron los objetivos principales, los cuales abarcan dos flujos de suma importancia para el desarrollo del sistema, cumpliendo satisfactoriamente con las historias de usuario identificadas para este sprint. Por otro lado, las pantallas para la reserva, la carga de los detalles de la sala, la gestión de los horarios, como la de las salas en sí, demostraron una correcta integración con la base de datos en Supabase, permitiendo que los flujos sean dinámicos con la información real. Finalmente, cabe destacar que la retroalimentación de los usuarios fue positiva, sin muchos comentarios escritos en las observaciones del sprint, lo que demuestra una claridad en cuanto a la información que se muestra en las pantallas, como la facilidad para hacer una reserva, escoger los horarios, el poder agregar y editar las salas de ensayo, como los horarios de estas, y la pantalla para realizar las búsquedas más detalladas mediante filtros. Evidentemente, se consideró la observación acerca de la protección de datos y la implementación del pago dentro de la aplicación para el futuro sprint. Sin más que agregar, este sprint dejó una base robusta para el avance hacia el siguiente sprint abarcando temas de personalización, historial de reservas, estadísticas administrativas, y la complementación del flujo de reserva con el pago.

4.4 SPRINT 3 – Flujo de Pago, Gestión de Reservas y Apartado Administrativo

4.4.1 Introducción

Nombre del Sprint: Flujo de Pago, Gestión de Reservas y Apartado Administrativo.

Duración del Sprint: 05/24/2025 – 05/30/2025

Equipo SCRUM:

- **SCRUM Master/Dev:** Jonathan Rosero
- **Product Owner:** José Torres.
- **Usuario Final:** Josué Guerra

4.4.2 Objetivos del Sprint 3:

- Implementar funcionalidades clave para la visualización y la gestión de reservas, tanto para el usuario de tipo cliente, como para el dueño de la sala (administrador).
- Completar el flujo de la reserva de las salas, mediante la integración de una pantalla con los detalles de cobro hacia el usuario cliente.
- Desarrollar funcionalidades las cuales permitan al usuario cliente el poder cancelar las reservas creadas, poder ver su historial de reservas.
- Para el panel del dueño de la sala (administrador), implementar una pantalla para poder observar el historial de las reservas recibidas por cada sala de ensayo, además de presentar resúmenes visuales sobre las salas de ensayo.

4.4.3 Historias de Usuario Asociadas al Sprint 3:

Tabla 15. Historias de Usuario Asociadas al Sprint 3

IDE	ROL	DESCRIPCIÓN	RAZÓN	CRITERIO DE ACEPTACIÓN
7	Usuario Normal	Necesito poder ver mis reservas realizadas (pasadas y futuras). A modo de tener un	Tener un control y seguimiento de las reservas realizadas.	Contar con una pantalla a modo de historial sobre reservas realizadas. Con algunos datos

		historial de mi actividad.		clave visibles (sala, fecha, estado).
8	Usuario Normal	Necesito el poder cancelar una reserva si ya no la necesito.	Liberar espacio y evitar confirmaciones en las reservas innecesarias.	Tener un botón para cancelar una reserva.
12	Usuario Administrador	Como dueño de la sala de ensayo quiero ver todas las reservas hechas en mis salas ordenadas por fecha.	Conocer el uso de mi espacio.	Pantalla tipo tabla o tarjetas con nombre del usuario, fecha y estado de la reserva.
16	Usuario Normal	Necesito poder realizar pagos dentro de la aplicación, de esta manera la reserva esta más que segura.	Confirmar la reserva de manera segura.	Integración de un flujo visual de pago exitoso.

Nota. Tabla desarrollada por el Autor, 2025

4.4.4 Tareas del Sprint 3

Tabla 16. Tareas del Sprint 3

CÓDIGO	NOMBRE DE LA TAREA	TIPO DE USUARIO	ESTADO
004.3	Cancelación de reservas con confirmación y motivo.	CLIENTE	COMPLETADO

004.4	Vista de historial de reservas para el usuario (pasadas y futuras).	CLIENTE/ADMIN	COMPLETADO
005.1	Panel de reservas recibidas por administrador.	ADMIN	COMPLETADO
005.2	Resumen administrativo: reservas en curso, canceladas, reembolsadas, total.	ADMIN	COMPLETADO
007.3	Función automática de actualización de estado de reservas (en curso, finalizada).	CLIENTE/ADMIN	COMPLETADO
008.2	Lógica de control para evitar reservas fuera del horario establecido.	CLIENTE/ADMIN	COMPLETADO
008.3	Integración (simulada) de confirmación visual de pago.	CLIENTE	COMPLETADO

Nota. Tabla desarrollada por el Autor, 2025

4.4.5 Modelo de Datos para el Sprint 3

Tabla ‘pago’

Dicha tabla es la encargada de gestionar la información asociada al registro del pago de una reserva. Si bien es cierto que le presente prototipo de aplicación web solo simula un flujo de pago, esta estructura nos permite dejar constancia del pago como tal, de esta manera saber si el pago fue confirmado y en qué fecha se lo realizó. Por otro lado, sirve como una base si a futuro se desea integrar pasarelas de pago reales. Revisar el Anexo 16 para más detalles de los atributos.

Figura 12. Tabla ‘pago’

pago	
id_pago	int4
id_reserva	int4
metodo_pago	text
referencia_pago	text
pagado	bool
fecha_pago	timestamp
monto	numeric

Nota. Captura obtenida de Supabase por el Autor, 2025

4.4.6 Actividades Técnicas

Frontend (Flutter)

- Creación de la pantalla ‘**screens/users/mis_reservas.dart**’, pantalla en donde se muestra y se carga todas las reservas confirmadas (ya pagadas) del usuario, información como el número telefónico del dueño de la sala, el inicio y fin de la reserva. Además, de que en la misma pantalla se puede cancelar la reserva.
- Creación de la pantalla ‘**screens/admins/reservas_recibidas.dart**’, con el objetivo de mostrar al usuario dueño de la sala de ensayo (administrador) la lista de las reservas recibida con información clave con el nombre, el número telefónico, si la reserva está confirmada, cancelada, como otras cosas para tener un historial con el usuario. Por último, se implementó tres gráficos estadísticos para mejorar aspectos visuales sobre las reservas y el uso de las salas dentro de la aplicación.
- Creación de funciones en Supabase como ‘**existeConflictoHorario**’, la cual verifica si la hora de inicio y fin de una reserva se cruza con el horario disponible, lo que permite evitar que los usuarios reserven fuera del horario disponible. De igual manera la creación de la función ‘**actualizarEstadosAutomaticos**’, la cual se invoca al crear la reserva, al cancelar la reserva y cuando esta reserva está en curso según corresponda. Y por último se manejó una lógica al momento de utilizar la función ‘**cancelarReservaYActualizarPago**’, la cual aparte de cancelar y cambiar el estado de la reserva, actualiza el modo de pago. Entonces, cuando la reserva se cancela antes de

4 horas de la reserva, se puede generar un reembolso, si se cancela la reserva dentro de esas 4 horas, no se puede generar un reembolso en la cancelación de la reserva.

- Creación de la pantalla **'screens/users/reserva_form_screen.dart'**, para completar con el flujo del pago, a través de un resumen presentado al usuario acerca de la reserva que va a realizar. Una vez que se confirma el pago, se genera el historial de la reserva, se bloquea el horario seleccionado de la reserva y ya no se muestra para los usuarios que deseen reservar ese día y hora.

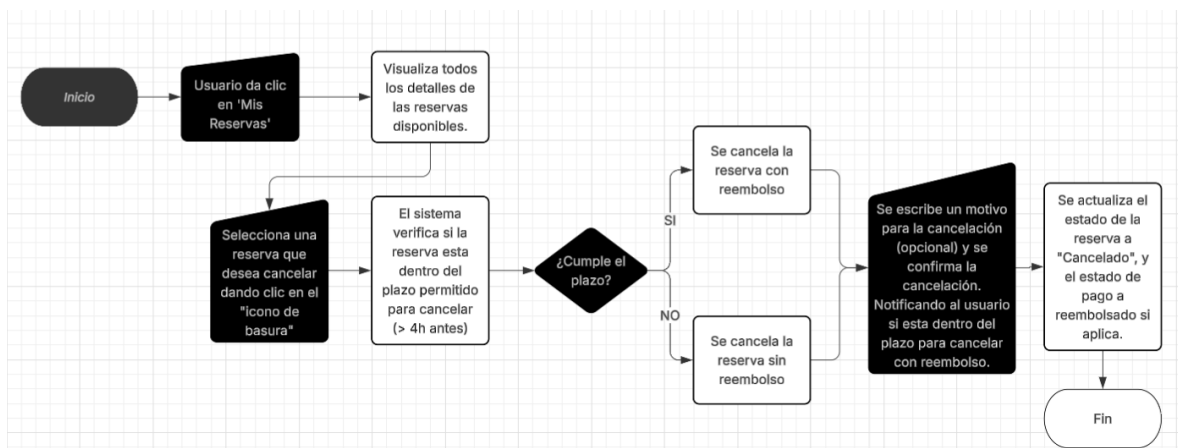
Backend (Supabase)

- Creación de la tabla **'pago'**, para almacenar y tener registro de las salas pagadas – reembolsadas.
- Función **'actualizar_estado_reservas'**, se encarga de cambiar el estado de la reserva, después de ejecutar una acción en específico, por ejemplo, cuando se cancela la reserva, se ejecuta junto al
- Función **'reservas_admin_por_propietario'**, nos permite obtener una lista detallada de todas las reservas para un administrador. Por lo que, nos brinda información completa desde los nombres, hasta los detalles del usuario cliente quien realizó la reserva.
- Función **'reservas_usuario_actual'**, es la encargada de obtener una lista de reservas de un usuario específico. Por lo que, nos muestra una tabla con información esencial para el usuario cliente.
- Función **'resumen_reservas_admin'**, es la encargada de brindar datos como resumen sobre el flujo de las reservas realizadas dentro de la aplicación. Brindando al usuario dueño de la sala de ensayo (administrador), detalles sobre la cantidad de usuarios que hayan realizado una reserva dentro de las salas disponibles del dueño.

La elección de Supabase como backend fue crucial en este sprint, dado que permitió aprovechar funciones RPC personalizadas como **'existeConflictoHorario'** y **'actualizarEstadosAutomaticos'** directamente desde PostgreSQL, reduciendo la complejidad de manejar lógicas críticas (como reembolsos o validación de horarios) en el frontend. Esto optimizó el rendimiento y minimizó errores por inconsistencias. Además, la integración nativa con PostgREST facilitó exponer las consultas complejas (por ejemplo, **reservas_admin_por_propietario** o **resumen_reservas_admin**), como endpoints RESTful, asegurando un consumo sencillo desde Flutter.

4.4.7 Diagrama de Flujo

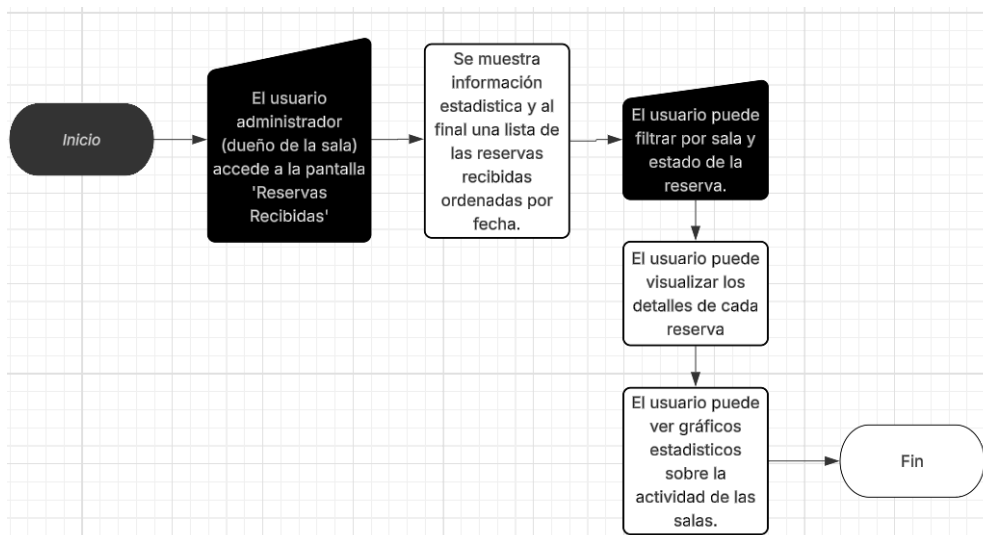
Figura 13. Diagrama de flujo para la cancelación de reservas



Nota. Diagrama realizado por el Autor en Lucidchar, 2025

En el presente diagrama de flujo, muestra la posibilidad del usuario al cancelar las reservas por cualquier motivo, en donde existe la restricción del sistema al momento de realizar la cancelación de la reserva si no es dentro de las 4 horas antes de la hora establecida para la reserva. Optimizando el tiempo para actualizar los datos de la reserva, librando a la sala para una mayor optimización y comprometiendo al usuario a no cancelar a última hora.

Figura 14. Diagrama de flujo para la visualización de reservas dueño de sala



Nota. Diagrama realizado por el Autor en Lucidchart, 2025

El presente diagrama resume la interacción que el dueño de una sala de ensayo puede llegar a tener en el panel administrativo, en este caso dentro de la pantalla de ‘Reservas Recibidas’.

4.4.8 Resultados Obtenidos

Los resultados obtenidos dentro de este sprint fueron complementarios para ambas partes de la aplicación, tanto para usuarios clientes, como usuarios dueños (administradores), por lo que se procuró brindar una solución enfocada en el aspecto del control sobre los estados de las reservas generadas. Permitiendo al dueño (usuario administrador) tener una visualización resumida sobre el movimiento de reservas existente en su sala de ensayo. Por otro lado, permitir al usuario cliente poder cancelar su reserva si este lo desea, recordándolo que existe una penalización al mismo por cancelar la reserva si esta fuera del rango permitido definido. Evidentemente, esta información debe estar actualizada en las pantallas de aceptación de términos, además de esto, se procuró de brindar la mejor experiencia visual posible hacia los usuarios. Podemos observar los productos desarrollados obtenidos del presente sprint dentro del anexo 12.

4.4.9 Observaciones y Comentarios (Sprint Review)

Los principales comentarios de los usuarios al momento de realizar la revisión del sprint 3 fueron muchas. Por un lado, José Torres destaco el panel administrativo sobre el cual le sirvió y aprobó los gráficos estadísticos, como los resúmenes de la información proporcionada acerca del flujo de reservas de las salas, como el resumen, los días más ocupados, y el porcentaje de uso, menciono que son cosas las cuales ayudan a tener una mejor visión acerca del comportamiento de la sala, sobre todo de los usuarios. Por otro lado, Josué Guerra destaco la facilidad para cancelar una sala, sin mucho apuro, ni complejidad, sin comentarios extras se aprobó la información que se muestra hacia el usuario sobre la reserva, de igual manera el apartado del pago con la información detallada. Se realizaron pruebas para verificar que los horarios no se crucen, de esta manera se permite un mejor control en cuanto a los usuarios que van reservando en la sala.

4.4.10 Conclusiones del Sprint 3 (Sprint Restrospective)

Se logro el cumplimiento de los objetivos establecidos para este sprint, lo que incluyo la implementación de funcionalidades clave, como la visualización de las reservas, permitiendo un mejor control para los usuarios en cuanto gestionar sus reservas, y para los dueños de las salas de ensayo tener una mejor visión sobre la actividad de sus salas o sala de ensayo. Además, se completó el flujo de la reserva, incluyendo la pantalla de pago con los detalles de la reserva realizada, permitiendo al usuario cliente tener una mejor experiencia de uso de la aplicación.

Por otro lado, la funcionalidad para cancelar la reserva de la sala de ensayo tuvo una correcta implementación, permitiendo a los usuarios poder gestionar su tiempo si necesitan cancelar la reserva, incluyendo una opción de reembolso si cancela antes de las 4 horas restantes para la reserva garantizando un compromiso de cumplimiento al usuario y dar la oportunidad a otros usuarios de poder reservar en ese horario.

Dentro del panel administrativo, se permite que el dueño de la sala tenga acceso a una mejor experiencia de visualización con información clave mostrada en graficas estadísticas lo que mejora el entendimiento de la actividad de sus salas con la aplicación, sin olvidar que las reservas recibidas se actualizan en base a las acciones del usuario cliente, lo que facilita y automatiza la gestión de estas, debido a que cuando se cancela una sala o se termina el tiempo de la reserva de la sala, se actualiza la disponibilidad de la sala, lo que facilita al administrador a no estar pendiente en cuanto a la actualización de disponibilidad de su sala. A pesar del cumplimiento de este sprint, todavía queda mejorar aspectos visuales, como funcionalidades complementarias para la entrega de un primer prototipo funcional de la aplicación, sin embargo, se ha dado un paso importante hacia la creación de una plataforma completa, a pesar de que faltan detalles, el presente equipo está bien posicionado para avanzar hacia el siguiente sprint.

4.5 SPRINT 4 – Personalización, Reseñas y Fidelidad.

4.5.1 Introducción

Nombre del Sprint: Personalización de Perfil, reseñas y fidelidad en la aplicación.

Duración del Sprint: 05/31/2025 – 06/05/2025

Equipo SCRUM:

- **SCRUM Master/Dev:** Jonathan Rosero
- **Product Owner:** José Torres.
- **Usuario Final:** Josué Guerra

4.5.2 Objetivos del Sprint:

- Implementar funcionalidades las cuales permitan la personalización y ajustes de cuenta hacia el perfil.
- Agregar comentarios y calificaciones para las salas de ensayo.

- Realizar un sistema de recompensas, manejando una moneda virtual “JamDollars”, para mantener fidelidad en el uso constante de la aplicación para la reserva de salas de ensayo.

4.5.3 Historias de Usuario Asociadas al Sprint 4

Tabla 17. Historias de Usuario Asociadas al Sprint 4

IDE	ROL	DESCRIPCIÓN	RAZÓN	CRITERIO DE ACEPTACIÓN
14	Usuario Normal	Quisiera poder tener personalización de mi perfil, poder editarlo cuando necesite cambiar cualquier aspecto de mi cuenta.	Poder eliminar la cuenta y todos los registros de esta misma.	Personalización de la cuenta con foto de perfil. Botón para eliminar mi cuenta.
18	Usuario Administrador	Quiero recibir notificaciones, cada vez que se registre una nueva sala de ensayo.	Poder estar al pendiente de las nuevas reservas hechas dentro de la aplicación.	Notificaciones en tiempo real sobre la reserva de la sala de ensayo con detalles.
19	Usuario Cliente	Me gustaría que cada vez que pague dentro de la aplicación pueda acumular créditos para poder canjearlos para una reserva gratis, como también descuentos en las reservas.	Para poder estar más interesado en seguir utilizando la aplicación.	Apartado de recompensas por fidelidad al usar la aplicación. Como descuentos, reservas gratis, etc.

Nota. Tabla desarrollada por el Autor, 2025

4.5.4 Tareas del Sprint 4

Tabla 18. Tareas Asociadas al Sprint 4

CÓDIGO	NOMBRE DE LA TAREA	TIPO DE USUARIO	ESTADO
007.1	Eliminación de cuenta del usuario desde pantalla de perfil.	CLIENTE/ADMIN	COMPLETADO
007.2	Actualización de datos de perfil (foto, nombre, etc.).	CLIENTE/ADMIN	COMPLETADO
009.1	Sistema de calificación y reseñas por sala de ensayo.	CLIENTE	COMPLETADO
009.2	Notificaciones en tiempo real sobre las reservas hechas en la sala de ensayo.	CLIENTE/ADMIN	COMPLETADO
009.3	Implementación de JamDollars, para promociones incentivando para incentivar el uso frecuente de la App.	CLIENTE	COMPLETADO

Nota. Tabla desarrollada por el Autor, 2025

4.5.5 Modelo de Datos Sprint 4

Tabla 'favorito'

Figura 15. Tabla 'favoritos'

Column Name	Data Type	Constraints
id_usuario	uuid	
id_sala	int4	
id_favorito	int4	Primary Key
fecha	timestamp	

Nota. Captura obtenida de Supabase por el Autor, 2025

Esta tabla nos permite almacenar información sobre la preferencia de los usuarios en cuanto a sus salas de ensayo favoritas, para guardarlas y acceder a ellas de manera fácil en una sección personalizada (ej: “Salas Favoritas”).

Tabla ‘resena’

Figura 16. Tabla ‘resena’

Column Name	Data Type	Constraints
id_sala	int4	
id_usuario	uuid	
comentario	text	
calificacion	int4	
id	uuid	Primary Key
creado_en	timestamp	

Nota. Captura obtenida de Supabase por el Autor, 2025

Se creo con el propósito de manejar un sistema de reseñas, en donde cada usuario puede dejar una calificación, y un comentario para una sala de ensayo. Por lo que solo se aceptarían comentarios de usuarios de tipo clientes. Se debe validar que un mismo usuario no pueda comentar dos veces en la misma sala.

Tabla ‘jamdollars’

Figura 17. Tabla ‘jamdollars’

Column Name	Data Type
id_usuario	uuid
monto	numeric
descripcion	text
id	int4
fecha	timestamp

Nota. Captura obtenida de Supabase por el Autor, 2025

La presente tabla nos permite el poder implementar un sistema de fidelidad, premiando al usuario los cuales utilizan más frecuente la aplicación web. En la mayoría de los casos, los usuarios se sienten más animados por utilizar la aplicación web, con el objetivo de ganar más premios. Por lo que, con esta tabla cada acción queda registrada para implementar trazabilidad dentro de las transacciones las cuales permitan sumar JamDollars.

4.5.6 Actividades Técnicas del Sprint 4

Frontend (Flutter)

- Se desarrollo la pantalla de **'screens/perfil_screen.dart'**, para implementar funcionalidades para la edición del perfil, el cambio de fotografía de perfil, además del botón para eliminar la cuenta con verificación de contraseña.
- Se realizó la creación de los archivos **'models/resena_model.dart'**, para definir un modelo de datos con los campos de la tabla 'resena', para temas de los comentarios hacia la sala de ensayo. Junto a **'widget/resenas/agregar_resena_widget.dart'**, para crear un formulario con campo de 'comentario' y una selección de estrellas en la parte inferior para brindar una calificación hacia la sala de ensayo. Por último, se implementó el archivo **'widget/resenas/lista_resenas_widget.dart'**, para la lista visual de los comentarios por parte de los usuarios en las salas de ensayo.
- Se realizo la creación de **'services/jamdollars_service.dart'**, para la gestión de la lógica sobre el sistema de fidelidad dentro de la aplicación. De esta manera se desarrollaron funciones para obtener el total de los jamdollars por usuario, además de registrar las transacciones realizadas, y el poder canjear dichos jamdollars por cada sala de ensayo reservada dentro de la aplicación. Además, se muestra notificaciones en la

pantalla justo después de realizar el pago de la reserva, a modo de confirmar la reserva y enseñar las recompensas al usuario por usar la aplicación.

- Se creó la pantalla `'screens/users/favoritosScreen.dart'`, para mostrar al usuario las salas favoritas marcada por él. Mediante la estrella que hay en cada una de las cards de las salas de ensayo, por lo que cada vez que el usuario daba clic o tocaba esa estrella la sala de ensayo se iba añadir a sus favoritas en esta pantalla.

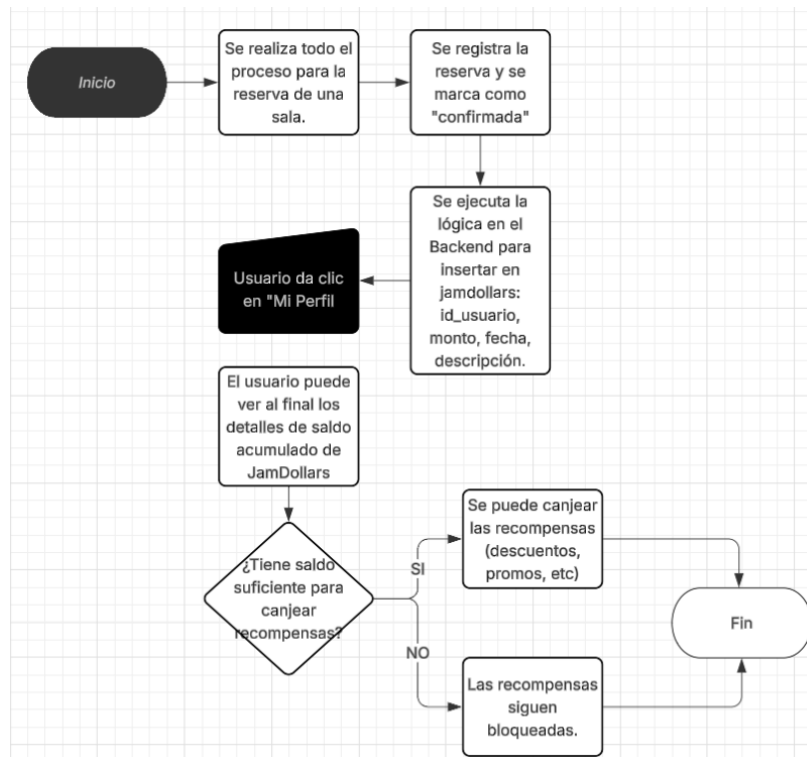
Backend (Supabase)

- Creación y conexión de las tablas `'resena'`, `'favoritos'`, `'jamdollars'` para implementar más dinámicas en la aplicación.
- Uso de la API Resend con la creación de un typescript para definir los mensajes de correo electrónicos a enviarse hacia los usuarios, en este caso para notificar las reservas creadas y canceladas. En donde, se conectó el archivo typescript hacia Supabase, y este en el front dentro de la función `'_enviarCorreo'`, que se llama cada vez que un usuario realiza una reserva y cada vez que el mismo la cancela. Cabe destacar, que se tuvo que utilizar un dominio a través de `'Go Daddy'`, el cual fue `'@QuitoJam.xyz'` para este tipo de pruebas.
- Se añadió una restricción, `'UNIQUE (id_usuario, id_sala)'` a modo de evitar que un usuario realice más de una reseña en la misma sala.
- Se añadió una validación lógica que permite canjear jamdollars, únicamente si existe un pago confirmado.

En este sprint, las decisiones técnicas estuvieron orientadas a enriquecer la experiencia del usuario mediante nuevas entidades (reseña, favoritos y jamdollars), así como la integración con servicios externos. Se optó por utilizar Supabase en conjunto con la API de Resend a través de TypeScript, lo cual permitió gestionar notificaciones por correo directamente desde triggers en la base de datos, manteniendo desacoplado al cliente Flutter. Asimismo, se aplicaron restricciones UNIQUE y validaciones en RPCs para garantizar integridad referencial, como en el caso de las reseñas por usuario y el canje de puntos JamDollars únicamente con pagos confirmados.

4.5.7 Diagrama de Flujo

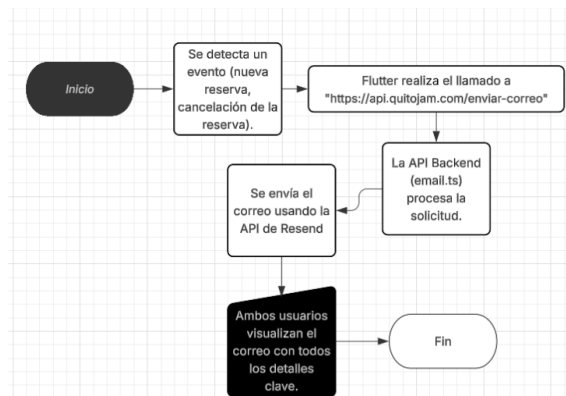
Figura 18. Diagrama de Flujo Acumulación de JamDollars.



Nota. Diagrama realizado por el Autor en Lucidchart, 2025

En el presente flujo podemos observar cómo se vincula la reserva confirmada con el sistema de puntos de fidelidad 'JamDollars', lo que motiva al usuario a utilizar con frecuencia la aplicación, para la reservar, como realizar los pagos dentro de la aplicación web.

Figura 19. Flujo 2 Envío de notificaciones por correo electrónico



Nota. Diagrama realizado por el autor en Lucidchart

El presente diagrama nos ayuda a entender de manera visual el cómo se dispara un correo electrónico desde Flutter usando una API por intermedio (Resend), de esta manera entregamos una notificación al usuario en tiempo real de lo que está pasando con sus reservas.

4.5.8 Resultados Obtenidos

Durante el sprint 4, se logró implementar una serie de funcionalidades las cuales están orientadas a mejorar la experiencia del usuario en cuanto al uso de la aplicación web, como la fidelidad a la misma. Cabe destacar que se logró cumplir con los objetivos establecidos para el sprint, entre los resultados principales se destacan la posibilidad de poder publicar reseñas o comentarios, permitiendo a los usuarios el poder calificar a las salas de ensayo. Por otro lado, el poder personalizar el perfil permite al usuario tener una mejor experiencia de uso de la aplicación, además de poder contar con la opción de eliminar su cuenta. En este sprint, también se optó por la implementación de un sistema para mantener a los usuarios dentro de la aplicación, siendo el sistema de puntos JamDollars, el cual permite al usuario poder acumular puntos o saldo, para poder canjearlo en futuras promociones como recibir distintos descuentos en la reserva de las salas, incluyendo la posibilidad de ganar una reserva totalmente gratuita de la sala. Finalmente, se logró implementar una solución para mandar notificaciones hacia los usuarios, tanto dueño como cliente, sobre el estado en tiempo real de la reserva, aspectos clave como la creación y la cancelación de una reserva son cosas que siempre deben ser notificadas en ambas partes. Sin nada más que añadir, se pueden observar los productos entregados de la aplicación en el *anexo 15*.

4.5.9 Observaciones y Comentarios (Sprint Review)

En la reunión para la revisión de los productos se destacó la implementación de las nuevas funcionalidades para la aplicación, una de ellas el hecho de agregar un incentivo virtual pudo responder a la necesidad anteriormente comentada de José Torres, con respecto a su temor de que el usuario ya no vuelva a utilizar la aplicación una vez que ya tenga el contacto del dueño de la sala. Por lo que, mediante la implementación del sistema de JamDollars, logramos cubrir ese problema, permitiendo así un motivo más para el uso de la aplicación de manera constante. En ambas partes, tanto Josué, como José se destacó la importancia de manejar las notificaciones de la aplicación, por más que estas fueran por correo electrónico, las notificaciones en tiempo real de estas aplicaciones tienen gran importancia para estar a los pendientes de las reservas. Finalmente, con respecto a la personalización del perfil y de poder agregar como favoritas a las

salas de ensayo, fue recibido de manera positiva, destacando la mejora sobre la experiencia de uso del usuario con respecto a la aplicación. No se agregaron más comentarios, por lo que se terminó el proceso de revisiones de sprints para este primer prototipo, con mucho positivismo y agradecimientos sobre el desarrollo de este proyecto.

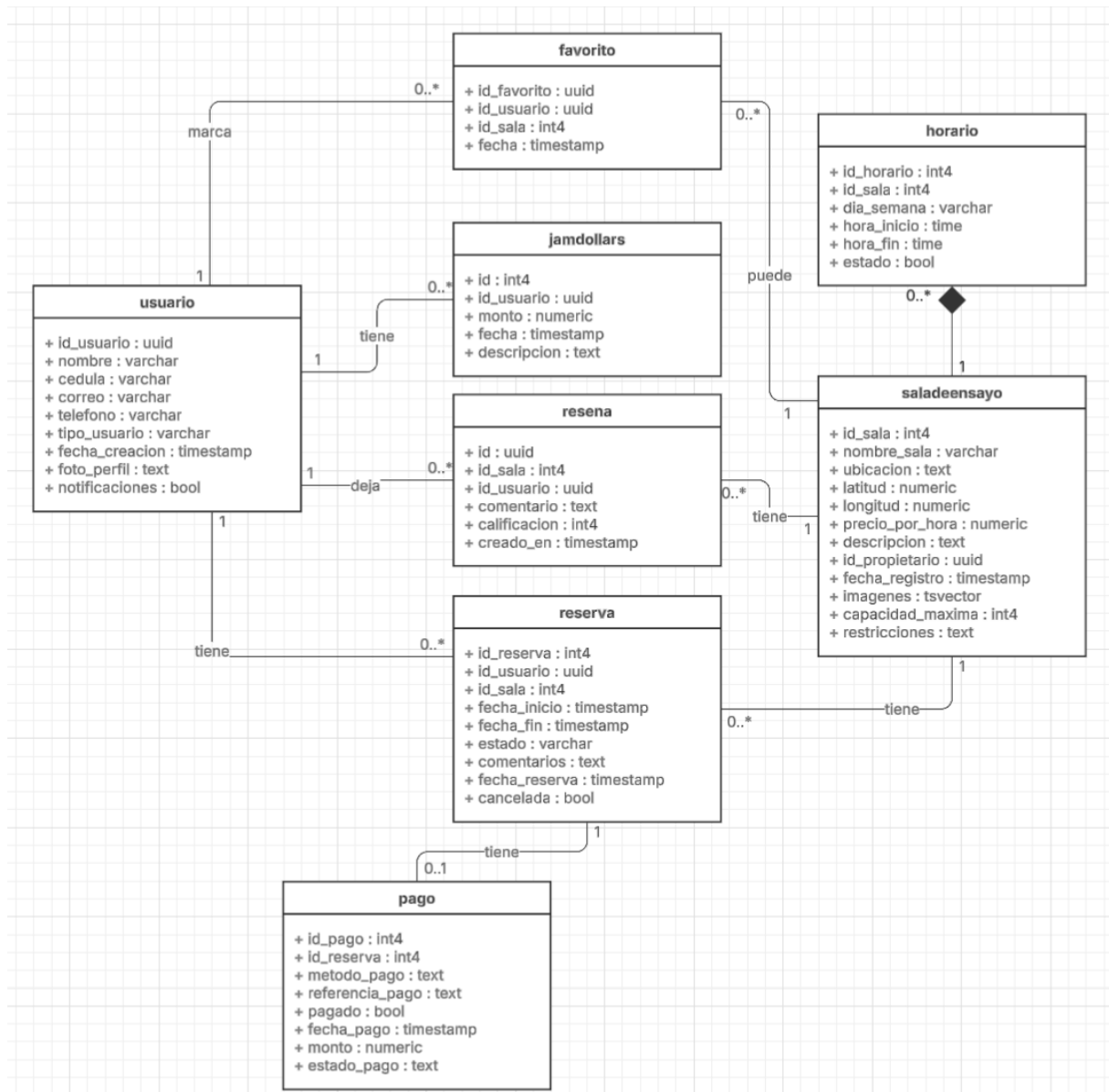
4.5.10 Conclusión del Sprint 4 (Sprint Restrospective)

Podemos mencionar que el sprint 4 representó un buen avance para la aplicación web, en términos de personalización del usuario, la interacción social mediante los comentarios y creación de un sistema de recompensas, motivando al usuario a utilizar la aplicación de manera continua. Mejorando, de igual manera los otros aspectos finales de la aplicación, por lo que con estos desarrollos se consolidaron las funcionalidades clave previamente definidas al inicio de este desarrollo. En conclusión, se completó con el desarrollo de un primer prototipo para ser lanzado hacia las pruebas de usuario, a modo de observar el comportamiento de la aplicación con los usuarios finales. Con este sprint damos como terminado una primera etapa de desarrollo, en donde se alcanzaron los objetivos establecidos tanto de este sprint, como del proyecto hasta el momento.

4.6 DIAGRAMA DE CLASES DEL SISTEMA

Para complementar el entendimiento técnico del prototipo desarrollado, a continuación, se presenta el diagrama de clases que ilustra de manera estructurada la arquitectura general del sistema. Este diagrama expone las principales entidades involucradas —tales como Usuario, Sala, Reserva, Reseña, Horario, Pago, Favorito y JamDollars— junto con sus respectivas relaciones. De esta forma, se refleja el modelo de datos implementado directamente en Supabase y su vinculación funcional con el cliente Flutter, permitiendo visualizar cómo se integran cada uno de los componentes para garantizar la operatividad del flujo completo de la aplicación.

Figura 20. Diagrama de Clases de la Aplicación ‘QuitoJam’



Nota. Diagrama de clases UML, realizado por el autor en Lucidchart

En el siguiente diagrama de clases se representan las entidades principales del sistema, sus atributos clave y las relaciones que garantizan la integridad del flujo de información en la aplicación QuitoJam. Se observa cómo un Usuario puede generar múltiples Reservas, dejar Reseñas, marcar Favoritos y acumular JamDollars como parte del sistema de fidelización. Las SalasDeEnsayo están vinculadas directamente a Horarios disponibles, reciben Reservas y pueden ser evaluadas mediante Reseñas o destacadas como Favoritos. Finalmente, el esquema muestra cómo las Reservas pueden generar registros en la tabla Pago, asegurando un control estructurado del proceso de cobro.

CAPÍTULO V: PROCESO DE PRUEBAS DE USUARIO – RESULTADOS CON RETROALIMENTACIÓN

Una vez finalizado el desarrollo del primer prototipo funcional, el presente proyecto no finaliza simplemente con la implementación de este con sus funcionalidades principales, se necesita de un proceso de validación con usuarios reales. En consecuencia, de lo mencionado, para poder tener una mayor retroalimentación en cuanto al proyecto desarrollado, se realizó una prueba de usabilidad hacia 15 personas.

Con el principal objetivo de evaluar principalmente métricas como la ‘Efectividad’, ‘Eficiencia’, y la ‘Satisfacción del Sistema’, tomadas de la norma ISO 9241-11:20218 como base principal para la evaluación de la usabilidad del prototipo. Como consecuencia, nos permitirá comprender cómo los usuarios interactúan con el prototipo, además de ver qué tan intuitivo resulta el uso de esta solución dentro de un contexto real.

5.1 CRITERIOS DE EVALUACIÓN

Para realizar las respectivas pruebas de usabilidad, primero que todo se deben establecer los principales criterios de evaluación, a modo de observar el comportamiento sobre el uso de la aplicación en los usuarios. Además, de evaluar las métricas anteriormente mencionadas de la ISO 9241-11:2018, tomaremos en cuenta el tiempo en el que los usuarios se demoran en realizar distintas actividades, por las tareas deberán estar clasificadas en base a los roles principales de la aplicación web.

Iniciamos entonces, con las principales actividades a realizar por parte del usuario tipo cliente y las metas en tiempos para completar cada una de las actividades.

- Registrarse e iniciar sesión, menor a 3 min.
- Realizar una búsqueda de una sala de ensayo, aplicando filtros de búsqueda, menor a 2 min.
- Realizar una reserva, menor a 3 min.
- Observar y gestionar reservas, menor a 2 min.
- Evaluar la sala de ensayo, menor a 2 min.

Seguidamente, con las actividades a realizar por parte del usuario tipo dueño de la sala de ensayo, con el objetivo de tiempo hacer ejecutada esa tarea.

- Registrarse e iniciar sesión, menor a 3 min.
- Creación de una sala de ensayo, con su horario, menor a 4 min.
- Consulta sobre las reservas recibidas, menor a 1 min.
- Editar información de la sala, menor a 1 min.
- Interpretar la información del panel estadístico, menor a 0:30s.

En conclusión, cada una de las presentes actividades deben tener un enfoque hacer realizadas en el menor tiempo posible, sin presionar al usuario, con el objetivo de observar el comportamiento real sobre el usuario final en situaciones reales que se presenten dentro de la aplicación. Por lo tanto, se realizó la creación de las tareas del usuario con enfoque en realizar las actividades descritas. Esto nos permitió, desarrollar un documento por roles enfocado en las tareas a realizar para los usuarios puestos a prueba. Estos dos documentos comprenden el *Anexo 17 y 18*, en los cuales se detallan un breve contexto de situación para simular un evento que se puede ocasionar que pueda involucrar a nuestra aplicación, con una ligera lista de tareas a realizar, sin ayudar ni especificar nada a los usuarios.

5.2 PRUEBAS REALIZADAS

El día 19 de junio del 2025, se realizaron las pruebas hacia los usuarios finales, para poner a prueba la usabilidad de la aplicación con un total de 15 participantes, los cuales firmaron un documento para constar con su participación, para más detalles observar el *Anexo 19*. Teniendo en cuenta que el principal perfil para poder realizar estas pruebas fue de buscar músicos de hobby, como de profesionales. Es pertinente, mencionar que las pruebas se realizaron por grupos separados, primero fue a un grupo de estudiantes de la Universidad San Francisco de Quito, seguidamente de otro grupo de estudiantes de la Pontificie Universidad Católica del Ecuador, finalmente se realizó la visita a un músico profesional para la realización de su respectiva prueba.

5.2.1 Tareas Realizadas.

El objetivo de la presente tabla es el de mostrar en resumen las actividades realizadas, como un promedio en el tiempo que se demoró en realizar la tarea. Lamentablemente, no se consta con un registro por usuario, para poder tener una información más exacta, sin embargo, se

consideró el tiempo en el que un primer usuario termino la actividad y cuando el ultimo usuario del grupo termino la actividad.

Tabla 19. Actividades Cliente – Promedio Tiempos

Actividades Realizadas	Primer Usuario en Completar	Ultimo Usuario en Completar	Meta
Registrarse e iniciar sesión	1:20 min	2:36 min	< 3 min
Realizar una búsqueda de una sala de ensayo, aplicando filtros de búsqueda	2:24 min	2:43 min	< 2 min
Realizar una reserva	0:46 min	1:50 min	< 3 min
Observar y gestionar reservas (cancelar)	0:35 min	1:25 min	< 2 min
Evaluar la sala de ensayo	0:53 min	1:30 min	< 2 min

Nota: Tabla realizada por el Autor, 2025

Como podemos observar en la presente tabla, nos podemos dar cuenta sobre los tiempos de ejecución por actividades, los cuales confirman que el prototipo mostro eficiencia para las tareas clave, como las de registrarse, de realizar, como gestionar reservas y dejar una reseña, dichas actividades fueron cumplidas con éxito en un tiempo menor al del propuesto en el de la meta. Lo que demuestra que la mayoría de los flujos principales propuestos son compresibles, además de rápidos inclusive para el usuario más lento.

Sin embargo, existe una excepción significativa al momento de realizar las búsquedas con filtros, donde no se cumplió la meta de realizar la tarea en menos de 2 minutos. El presente resultado, nos indica que existe un punto crítico en la mejora de la interfaz de ‘busqueda.dart’.

En conclusión, dentro de las actividades realizadas para el usuario de tipo ‘cliente’ el sistema cumple los objetivos de eficiencia en 4 de las 5 actividades evaluadas, por lo que se debe optimizar el proceso de búsqueda, a fin de garantizar una experiencia fluida en este apartado.

Tabla 20. Actividades Administrador – Promedio Tiempos

Actividades Realizadas	Primer Usuario en Completar	Ultimo Usuario en Completar	Meta
Registrarse e iniciar sesión	1:34 min	2:20 min	< 3 min
Creación de una sala de ensayo, con su horario,	3:35 min	4:49 min	< 4 min
Consulta sobre las reservas recibidas	0:35 min	0.50 min	< 1 min
Editar información de la sala	1:35 min	1:52 min	< 1 min
Interpretar la información del panel estadístico	0:15 min	0:23 min	< 0.30 min

Nota. Tabla realizada por el Autor, 2025

La presente tabla nos muestra que la mayoría de los flujos administrativos alcanzaron la meta propuesta del tiempo en ser realizadas. Aspectos, como el registro e inicio de sesión, además de la consulta, como la interpretación del panel estadístico, fueron completados superando la meta establecida. Destacando la interpretación del panel estadístico, la cual fue la tarea más rápida en ser ejecutada, sin embargo, existen dos tareas las cuales no pudieron cumplir con la meta. Estos valores, indican que posiblemente el formulario es muy extenso o existen confusiones al momento de completar el formulario de creación de las salas, como el de

edición, a pesar de que se cuenta con herramientas de geolocalización, como la posibilidad de poder agregar varios horarios al mismo tiempo.

5.2.2 Prueba Pre-Test

Continuando con la evaluación de las pruebas realizadas, tenemos en consideración las pruebas Pre y Post test que se realizó a los usuarios finales. Por lo que, a continuación, se presentará una tabla

Tabla 21. Resumen Resultados Pre-Test

Métrica	Resultados Clave	Detalles Adicionales
Rango de Edad de los Encuestados	La mayoría de los participantes (73%) se encuentra en el rango de edad entre 21 a 25 años.	Se encuestaron a 15 personas en total.
Uso Previo de Apps de Reserva	El 67% de los encuestados no ha utilizado o conoce una aplicación para reservar espacios físicos.	El 33% sí ha utilizado alguna aplicación de este tipo, siendo la más conocida Airbnb, además de una mención de Microsoft Booking.
Familiaridad con Aplicaciones Web/Móviles	El 60% se considera "Muy familiarizado" con el uso de aplicaciones en general.	33% se considera "Algo familiarizado", mientras que un 7% se considera "Poco familiarizado".
Funcionalidades Esperadas en una App de Reserva	Las funcionalidades más esperadas son "reserva", "disponibilidad", "ubicación", "horario" y "equipos".	Por lo que, los usuarios esperan poder encontrar resultados en base a sus necesidades personales.
Importancia de la Navegación Simple y Clara	Se le otorga una alta importancia, con una puntuación <i>Net Promoter Score</i> (NPS) de 60.	10 promotores, 4 pasivos y 1 detractor. Lo que demuestra una importancia en temas de navegación de la aplicación.

Importancia de un Proceso de Reserva Rápido	Se considera muy importante, con una puntuación NPS de 67.	10 promotores, 5 pasivos y 0 detractores. Demostrando que existe una alta valoración en los temas de servicio de reserva rápido.
Uso de Filtros de Búsqueda	El 60% de los usuarios "siempre" utiliza filtros (precio, ubicación, horario) al buscar opciones en una app, y el 40% lo hace "a menudo".	Esto nos permite darnos cuenta sobre el impacto que tiene utilizar estos filtros, en cualquier aplicación que involucre un panel de búsqueda.
Facilidad para Encontrar Salas de Ensayo/Grabación	La experiencia general para encontrar una sala ha sido difícil, resultando en un NPS de -40.	3 promotores, 3 pasivos y 9 detractores. Nos indica sobre la necesidad, como la problemática inicial de este proyecto.
Disponibilidad de Salas de Ensayo Cercanas	El 73% de los encuestados ha tenido que viajar y movilizarse por la ciudad para encontrar una sala.	Solo el 27% ha encontrado salas de ensayo cerca de su domicilio.
Problemas Comunes al Buscar Salas	Los problemas más comunes mencionados son la "poca disponibilidad", "poca accesibilidad horaria", "falta de detalles de equipos" y "cancelaciones arbitrarias".	Otros inconvenientes incluyen la ubicación y la falta de opciones. Esto nos ayuda a identificar más problemáticas al tratar de reservar una sala de ensayo sin utilizar nuestra solución.

Nota. Tabla Realizada por el Autor en base al Anexo 20, 2025

Mediante la implementación de las pruebas ‘Pre-Test’, nos podemos dar cuenta sobre como en la actualidad si existe una problemática relacionada al tema de la búsqueda, como reserva de las salas de ensayo en quito. Cabe destacar, que dichas preguntas tuvieron un enfoque más hacia el uso previo de aplicaciones de esta índole, como la necesidad de los usuarios músicos

en encontrar salas de ensayo cercanas, enfocándose más en la principal problemática de este proyecto.

5.2.3 Prueba Post-Test

Tabla 22. Resumen Resultados Post-Test

Métrica de Evaluación	Resultados Clave para el Rol "Cliente"	Resultados Clave para el Rol "Administrador"
Facilidad de Uso y Experiencia General	El 87% de los usuarios calificó la experiencia como "Fácil" (40%) o "Muy fácil" (47%). La experiencia general con el proceso de reserva y uso de la app obtuvo un Net Promoter Score (NPS) de 75.	La facilidad para crear y editar una sala de ensayo obtuvo un NPS de 34. Lo que significa que fue relativamente positivo la experiencia de uso. El panel para ver detalles de reservas, cancelaciones y cambios fue muy útil, con un NPS de 86.
Navegación y Diseño	El 85% de todos los usuarios describió la navegación como "Intuitiva" (64%) o "Muy intuitiva" (21%). Además, el 87% consideró el diseño visual como "Muy Adecuado".	
Proceso de Búsqueda y Reserva	El proceso para buscar y reservar una sala fue muy claro, obteniendo un NPS de 100. En consecuencia, el tiempo para completar la reserva fue "Rápido" (63%) o "Muy rápido" (38%).	No Aplica
Claridad y Utilidad de la Información	La información sobre cada sala fue suficiente para tomar una decisión, con un NPS de 50. Por	La claridad de la información sobre el rendimiento de las salas fue máxima, con un

	otro lado, La información en el historial de reservas se calificó con un NPS de 38. El 100% encontró la confirmación de reserva por email clara y útil.	NPS de 100. El 100% encontró la confirmación de reserva por email clara y útil.
Funcionalidades Específicas	La utilidad de los filtros de búsqueda fue alta, con un NPS de 63. Por lo que, el uso de geolocalizadores para buscar salas cercanas fue muy bien calificado, con un NPS de 75.	La gestión de disponibilidad para fechas específicas resultó difícil, con un NPS de 34. En donde, el 57% pudo actualizar fechas sin problemas, pero el 43% no.
Confusión Durante el Uso	El 60% no sintió confusión, pero un 40% sí. Las áreas de confusión fueron el buscador por palabras y la sección de horarios.	No evaluado directamente con una pregunta específica para este rol.
Satisfacción y Recomendación	El 100% de los usuarios recomendaría la aplicación "QuitoJam".	El 100% de los usuarios recomendaría la aplicación "QuitoJam". Además, El 71% está "De acuerdo" o "Muy de acuerdo" con una comisión del 5% por el uso de la app, considerándola razonable.
Comentarios Adicionales y Sugerencias de Mejora	Se sugirió "Mejorar la presentación de los detalles de las salas" y "Mejorar la búsqueda por texto".	La principal sugerencia fue "Mejorar el tema para editar los horarios, sin necesidad de borrarlos". Además, de mejorar el apartado UI de la misma sección.

Nota. Tabla Realizada por el Autor en base al Anexo 21, 2025

En la presente tabla, podemos observar los resultados finales, sobre la evaluación del 1er prototipo. Donde, mayormente hay calificaciones relativamente positivas, nos podemos dar cuenta que la aplicación en términos de cumplir las acciones con las que fue diseñada desde un principio, si las cumple. Sin embargo, hay que destacar algunas calificaciones no fueron altamente satisfactorias, identificando problemáticas dentro de la UI que no permiten tener una buena experiencia en cuanto al uso de la aplicación.

5.2.4 Retroalimentación

Gracias a las pruebas de usuarios, se logró observar la forma en como QuitoJam actuaría dentro de un entorno real. A pesar de los buenos comentarios, como el apoyo en general sobre la idea de tener una aplicación web enfocada a cubrir la necesidad de búsqueda, reserva y gestión de disponibilidad de las salas de ensayo. Hubo ciertas actividades en donde los usuarios tuvieron algo de confusión, o no pudieron completar ciertas tareas solicitadas. Siendo las dos actividades relacionadas a la creación, y edición de las salas, enfocado en el apartado de los horarios. Por otro lado, los usuarios que hicieron de rol cliente no encontraron muchas dificultades en realizar las tareas solicitadas.

Cabe destacar que las pruebas pre y post fueron muy importantes para observar tanto el rendimiento de la aplicación en general, como el nivel de aceptación sobre tener una aplicación que abarque la reserva, búsqueda y gestión de salas de ensayo. No obstante, la interacción verbal directa con los usuarios permitió conocer más sobre cómo mejorar el presente prototipo de aplicación web, para poder llegar a una aplicación completa.

Hubo comentarios, los cuales reflejaban, algunas funcionalidades extra que podrían incluirse dentro de la aplicación. Siendo una de estas el poder manejar los dos tipos de usuario, habiendo casos en los cuales un usuario pueda contar con un espacio el cual quiere alquilar, y a la vez poder reservar, como buscar espacios más completos para su uso. De igual manera, al momento de crear una sala de ensayo, contar con un listado el cual pueda manejarse como checklist determinando equipo disponible, sin la necesidad de escribir todos los detalles a mano. Finalmente, hubo confusiones y sugerencias de cambiar el nombre de ‘administrador a dueño de sala’.

Varias posibles mejoras fueron identificadas al realizar las pruebas de usabilidad, por otro lado, con respecto a las métricas como ‘Efectividad’, ‘Eficiencia’, y la ‘Satisfacción del Sistema’, correspondientes a la ISO 9241-11:2018, podemos mencionar que métricas como la efectividad

no se logró cumplir en un 100%, por lo anteriormente dicho. A diferencia de la eficiencia la cual se consideraría un 90% y la satisfacción del sistema el cual se considera como exitoso, basándonos en los comentarios altamente positivos que se obtuvieron a través de estas pruebas. En conclusión, esta retroalimentación nos lleva al poder realizar las respectivas mejoras finales a modo de presentar el prototipo final de la aplicación web, considerando las mejoras mencionadas por los usuarios reales.

5.2.5 Nuevas Historias de Usuario identificadas

Estos hallazgos reflejan que existen funcionalidades, como la de agregar la disponibilidad de horarios, necesita tener una mejora visual lo que vuelve necesario para el usuario final. Por lo que, en función a los comentarios de algunos usuarios, se lograron identificar ciertas mejoras viables para el prototipo final de la aplicación web, tales mejoras las vamos a reflejar con la ayuda de la siguiente tabla.

Tabla 23. Nuevas Historias de Usuario

IDE	ROL	DESCRIPCIÓN	RAZÓN	CRITERIO DE ACEPTACIÓN
1	Usuario de Rol Cliente	Necesito poder ingresar como dueño, en el caso que disponga una sala de ensayo para publicarla.	Con el objetivo de poder rentar, como gestionar un espacio de ensayo.	Poder cambiar de tipo de cuenta. Contar con un botón para ello.
2	Usuario de Rol Administrador	Quiero tener acceso a las funciones de cliente, para poder reservar otras salas de ensayo en el caso de que necesite	Para poder tener la opción de ir a otros espacios los cuales consten con mejores equipos que mi sala.	Poder cambiar de tipo de cuenta. Poder contar con un botón para ello.

3	Usuario de Rol Cliente	Me gustaría observar detalles específicos en las salas de ensayo, como amplificadores, instrumentos, entre otras cosas.	Con el objetivo de tomar una mejor decisión o conocer más los detalles de la sala antes de la reserva.	Contar con un listado de equipos por cada sala si esta la dispone.
4	Usuario de Rol Administrador	Quisiera que el apartado de crear, modificar horarios sea más entendible.	Con el propósito de entender más el flujo de creación de las salas como de disponibilidad.	Mejorar apartados de UI para una mejor visión sobre la gestión de horarios.
5	Usuario de Rol Cliente	Quisiera que, al momento de observar mis reservas realizadas, pueda acceder a los detalles de la sala reservada.	Para recordar la ubicación, detalles de la sala, entre otras cosas.	Agregar un acceso rápido desde los detalles de las reservas hacia la sala que se reservó.
6	Usuario de Rol Cliente	Hacer los comentarios una vez que se haya finalizado mi reserva. Que no me permita hacer más de un comentario.	Con el propósito de mantener un control sobre los comentarios, así pueda tener una mejor visión de la sala.	Agregar una restricción para no agregar más de dos comentarios por reserva finalizada del usuario.
7	Usuario de Rol Administrador	Quisiera poder editar las fotografías de las salas de ensayo, además de los	Con el propósito de completar de mejor manera el apartado	Poder editar las imágenes de las salas, además de los

horarios sin tener que borrarlos.	sobre la edición de los campos.	horarios sin tener que eliminarlos
--------------------------------------	------------------------------------	---------------------------------------

Nota. Tabla Realizada por el Autor, 2025

5.3 MEJORA DEL PROTOTIPO

5.3.1 Nuevo Product Backlog

A través de las historias de usuario identificadas, y con inspiración a la metodología de SCRUM, aplicaremos el desarrollo de un Product Backlog, para la identificación, como separación de las actividades a realizar, para determinar un prototipo final.

Tabla 24. Product Backlog Complementario

Código	NOMBRE	COMPLEJIDAD	PRIORIDAD
010.1	Botón para cambiar de rol sin perder la información.	Baja	Media
010.2	Agregar lista de detalles sobre equipos en específico.	Alta	Alta
010.3	Mejorar la UI en el apartado de gestionar horarios.	Media	Media
011.1	Integrar un botón para acceder a los detalles de la sala, desde 'Mis Reservas'.	Baja	Media
011.2	Restringir a un solo comentario por usuario, además de permitir	Media	Alta

	comentarios únicamente si se finalizó la reserva.		
011.3	Permitir al usuario editar las fotografías. Además, de editar los horarios	Baja	Media

Nota. Tabla Realizada por el Autor

5.3.2 Mejoras realizadas

Frontend (Flutter):

- Se añadió el botón ‘cambiar a dueño/cliente’, para ambos roles dentro de la pantalla ‘**screens/perfil_screen.dart**’, por lo que
- Para el apartado de la gestión de los horarios, se realizó cambios en la pantalla ‘**screens/admins/gestionar_horario.dart**’, donde se implementó cambios en el título, y en el botón para agregar nuevos horarios (véase Anexo 22). Además, de contar con nuevas.
- Se modificó la función ‘obtenerHorariosDeSala’ y se agregó ‘actualizarHorario’ dentro de ‘**services/supabase_service.dart**’, para validar las salas que sean bloqueadas y las disponibles, además de agregar un apartado de editar el horario. Además, de que se ajustaron otras pantallas, para no mostrar los horarios bloqueados. Por lo que se creó la función ‘obtenerHorariosActivosDeSala’, el cual verifica si el estado del horario es true para mostrar el horario.
- Se agregó la función ‘usuarioPuedeComentar’, para validar si se completó la reserva, entonces ahí un usuario puede realizar un comentario en la sala reservada. Además, de modificaciones en otras pantallas para evitar que el usuario pueda comentar antes de completar la reserva.
- Se agregó un botón en ‘**screens/users/mis_reservas.dart**’, que redirige al usuario a la pantalla de ‘**screens/users/detalles_sala.dart**’.
- Se agregó el archivo ‘**widgets/user_rol.dart**’, para que la aplicación se actualice en tiempo real al momento de que el usuario cambie de rol.

- Dentro del atributo de descripción de la sala, se creó una sección de ‘descripción avanzada’ para guardar una lista de atributos. Por lo que, esta lista predefinida, permite al usuario poder seleccionar los equipos mediante checklists. Además, de que se guarda en el mismo campo de descripción en la base de datos.

Backend (Supabase)

- Se creó una nueva RPC ‘verificar_reserva_finalizada’, la cual valida si un usuario ya finalizó una reserva en una sala específica.

Las mejoras incorporadas tras las pruebas de usuario reforzaron el papel de Supabase como eje del proyecto, al permitir extender lógica con nuevas RPCs como verificar_reserva_finalizada, validar horarios activos y gestionar cambios en tiempo real con triggers y actualizaciones automáticas. Estas capacidades server-side redujeron la carga en el cliente Flutter, mantuvieron la integridad de datos y facilitaron que el sistema evolucione de manera consistente frente a los ajustes iterativos propuestos por los usuarios durante el proceso de validación. Los cambios realizados, se pueden observar dentro del *Anexo 22*.

CAPÍTULO VI: CONCLUSIONES & RECOMENDACIONES

6.1 CONCLUSIONES

El desarrollo del prototipo 'QuitoJam' demostró la viabilidad del uso de recursos digitales para la mejora del proceso de búsqueda y reserva de salas de ensayo musicales en la ciudad de Quito mediante la implementación de una aplicación web. En esta se integraron filtros de búsqueda, visualización de disponibilidad, un sistema de reservas, apartados de geolocalización, además de una interfaz amigable, permitiendo conectar estos espacios de creatividad musical con la comunidad musical de la ciudad.

En respuesta a las preguntas planteadas en la problemática principal, se identificó que actualmente no existe una plataforma enfocada exclusivamente en la gestión de reservas de estos espacios musicales para ensayo, lo que reafirma la pertinencia de ofrecer este tipo de solución. Como apoyo para la implementación de esta solución, la gran mayoría de los usuarios tienen claro el concepto sobre aplicaciones que permiten realizar reservas (como Airbnb), lo que ayudó a enfocar el prototipo de aplicación hacia una solución familiarizada con el usuario. Por otro lado, en el desarrollo del prototipo, así como en su evaluación, los datos más valorados por los usuarios fueron la disponibilidad horaria del espacio, ubicación exacta, precios, equipos disponibles, y por último la confianza que puede brindar una aplicación web. Es importante destacar que la recolección de información mediante entrevistas directas y evaluaciones del prototipo fueron clave para determinar las necesidades principales de los usuarios, además de realizar correcciones para poder entregar un producto final.

En este contexto, se comprobó que la usabilidad dentro de una aplicación es un factor determinante en la toma de decisiones de los músicos. En consecuencia, mediante una buena experiencia de navegación, junto con una interfaz clara y con una retroalimentación visual efectiva, se observó en los músicos una aceptación del sistema. Finalmente, cabe mencionar que con todo lo desarrollado, el prototipo logró cumplir con sus objetivos principales, al mismo tiempo de brindar una respuesta o llamado de atención para tener en cuenta al arte, no como un pasatiempo, sino como un nicho de mercado el cual no está siendo explorado al 100%, tanto en la ciudad de Quito como en el país. La realización de este proyecto permitió reconocer la capacidad y el impacto que tiene una solución digital en la actualidad.

6.2 RECOMENDACIONES

Principalmente, se recomienda continuar con el desarrollo de la presente aplicación, priorizando la implementación de funcionalidades pendientes, como la mensajería dentro de la aplicación, con el objetivo de poder crear una comunidad artística dentro de la ciudad. Por lo que la presente aplicación puede evolucionar para convertirse en una red social enfocada en las personas que tienen esta afición o ya trabajan de manera profesional en la industria musical. Nuevamente, para poder brindar más soluciones mediante el presente proyecto, o para encontrar más funcionalidades que implementar, se debe primero conocer a las personas que utilizarían este medio, de esta manera poder brindarles una solución personalizada.

Esto conduce a la segunda recomendación, que es principalmente realizar un estudio más profundo sobre las necesidades que enfrentan los músicos locales en la ciudad de Quito, así como la manera en que mediante una solución tecnológica se pueden resolver. Por ejemplo, el presente proyecto nació de una necesidad identificada al no encontrar ningún espacio disponible cercano para la producción musical, para después conocer por parte de un músico profesional que sí existen varios espacios de este tipo en el área. Sin embargo, al no tener un lugar seguro para poder publicar sus espacios, los propietarios se limitan a ser conocidos solo por medios tradicionales. Por otro lado, existen músicos que sí cuentan con un espacio apropiado para realizar un ensayo, que de hecho sí los alquilan, pero no pueden llegar a tener un mayor alcance al no disponer de una solución como 'QuitoJam'.

Finalmente, se recomienda para el presente proyecto buscar alianzas estratégicas con instituciones culturales, como universidades y espacios culturales, con el objetivo de fomentar el presente proyecto, agregando la posibilidad de identificar más funcionalidades que podría tener una versión final de esta aplicación. En consecuencia, estas colaboraciones no solo permitirían ampliar el alcance del proyecto, sino que también se puede facilitar el acceso a espacios, financiamiento, así como visibilidad institucional. De este modo, estas recomendaciones ayudarían a consolidar a QuitoJam como una herramienta clave para el desarrollo de un ecosistema cultural y musical de la ciudad de Quito. Para finalizar, uno de los desafíos más relevantes fue la integración segura entre múltiples roles, especialmente al restringir acciones administrativas. Este proceso brindó valiosos aprendizajes sobre validaciones front-end y back-end, y resalta como futura línea de trabajo la incorporación de machine learning para predecir horarios de alta demanda, incrementando el valor analítico de la aplicación.

REFERENCIAS


- Arezana, D. (2022, 25 de marzo). *Principios de usabilidad web de Jakob Nielsen y el diseño UX*. Semrush Blog. <https://es.semrush.com/blog/usabilidad-web-principios-jakob-nielsen/>
- AWS. (s.f.-a). *¿Qué es Flutter? - Explicación de la aplicación Flutter*. Amazon Web Services, Inc. Recuperado el 30 de junio de 2025, de <https://aws.amazon.com/es/what-is/flutter/>
- AWS. (s.f.-b). *¿Qué es una API? - Explicación de interfaz de programación de aplicaciones*. Amazon Web Services, Inc. Recuperado el 30 de junio de 2025, de <https://aws.amazon.com/es/what-is/api/>
- Barragán, A. (2024, 8 de febrero). *Supabase vs Firebase: Análisis de soluciones de Backend*. OpenWebinars.net. <https://openwebinars.net/blog/supabase-vs-firebase/>
- Bhamare, S. (2025, 4 de marzo). *What is Flutter: Definition, Benefits, and Limitations*. BrowserStack. <https://www.browserstack.com/guide/what-is-flutter>
- Briseño, M. V. (2024). *Ingeniería de requerimientos (IR)* [Apuntes de clase]. Universidad Autónoma del Estado de Baja California. [http://ing.ens.uabc.mx/docencia/apuntes/computacion/requerimientos\[12147\].pdf](http://ing.ens.uabc.mx/docencia/apuntes/computacion/requerimientos[12147].pdf)
- Calle García, A. J. (2024). [Título del artículo]. *Comunicaciones y Desarrollo*, 27(2). <https://doi.org/10.21503/cyd.v27i2.2602>
- Camarano, N. (2024, 12 de diciembre). *Backend de Supabase: todo lo que necesita saber*. No-Code Start-Up. <https://nocodestartup.io/es/backend-de-supabase-todo-lo-que-necesitas-saber/>
- Cuadrado, G. C. (2024, 14 de septiembre). *Visual Studio Code: Editor de código para desarrolladores*. OpenWebinars.net. <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>
- GitHub. (2025). *Acerca de GitHub y Git*. Documentación de GitHub. Recuperado el 30 de junio de 2025, de <https://docs.github.com/es/get-started/start-your-journey/about-github-and-git>

- Merino, Á. M. (2024, 19 de septiembre). *Firestore: Una plataforma integral para el desarrollo de aplicaciones web y móviles*. Sngular.
<https://www.sngular.com/es/insights/313/firebase>
- Microsoft. (2021, 3 de noviembre). *Code editing. Redefined*. Visual Studio Code.
<https://code.visualstudio.com/>
- Ministerio de Cultura y Patrimonio de Ecuador. (s.f.). *Registro único de artistas y gestores culturales - RUAC* [Conjunto de datos]. Datos Abiertos Ecuador. Recuperado el 28 de junio de 2025, de
<https://datosabiertos.gob.ec/dataset/registro-unico-de-artistas-y-gestores-culturales-ruac>
- OpenAI. (2025). *ChatGPT* (Versión GPT-4o) [Modelo de lenguaje grande].
<https://chat.openai.com>
- Red Hat. (2023, 20 de enero). *¿Qué es una API y cómo funciona?* Red Hat.
<https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>
- Rodríguez, A. & [Apellido del segundo autor, Inicial]. (2024). [Título del artículo]. *Revista Tecnologías*, 27(60). <https://doi.org/10.22430/22565337.2958>
- Salamanca Vega, L. C. (2025, 30 de mayo). Prototipado iterativo: del error a la solución. *Revista Neuronum*, 11(4), 1–12.
<https://eduneuro.com/revista/index.php/revistaneuronum/article/download/597/677>
- Supabase. (s.f.). *Supabase | the Postgres development platform*. Recuperado el 30 de junio de 2025, de <https://supabase.com>
- Titusunta Arias, E. P. (2023). *La construcción de la identidad en los jóvenes del sur de Quito en torno a la música extrema* [Tesis de pregrado, Universidad Central del Ecuador]. Repositorio Digital UCE.
<http://www.dspace.uce.edu.ec/handle/25000/31690>

ANEXOS

ANEXO 1 – ENTREVISTA A JOSÉ TORRES

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



Quito, 1 de mayo 2025

Mi nombre es **Jonathan Steve Rosero Garcia**, estudiante de la carrera de **Ingeniería de Sistemas de la Información** en la **Pontificia Universidad Católica del Ecuador**. Actualmente me encuentro desarrollando un trabajo de titulación titulado:

“Prototipo de aplicación web para la reserva y gestión de salas de ensayo musicales en Quito (QuitoJam)”, cuyo objetivo principal es diseñar una solución digital que facilite la administración y acceso a espacios de ensayo para músicos.


En este contexto, solicito su colaboración para participar en una **entrevista de carácter académico**, cuyo propósito es recopilar información valiosa relacionada con las necesidades, problemáticas y perspectivas de los potenciales usuarios o administradores de salas de ensayo.

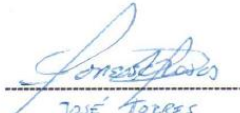
Declaro expresamente que:

- La información proporcionada será utilizada **exclusivamente con fines académicos**.
- No será compartida con terceros ni utilizada con fines comerciales o de lucro.
- Los datos serán tratados con **confidencialidad y respeto a la privacidad** del entrevistado.
- La participación es **voluntaria**, y el entrevistado/a puede desistir en cualquier momento.

Agradezco de antemano su tiempo y disposición para contribuir al desarrollo de este proyecto.

Atentamente,


Jonathan Steve Rosero Garcia


José Torres

- **Entrevista Transcrita:**

- **Entrevistador:** Hola José, como estas, gracias por el tiempo que te tomaste para esta entrevista, después de hablar un poco sobre mi proyecto, sin nada más que decir

vamos a iniciar con la entrevista. Coméntanos sobre ti quien eres y cuantos años llevas en la industria musical.

- **Entrevistado:** Buenos días, Steve, encantado de ayudarte, pues que te puedo decir, llevo trabajando con bandas, artistas y en eventos musicales por más de 20 años, eh estado en más de 100 eventos tanto nacionales como internacionales, actualmente estoy trabajando como productor musical, también como organizador de eventos, armo escenarios musicales, además de realizar grabaciones a bandas y artistas en mi sala.

- **Entrevistador:** Perfecto José, al parecer veo que tienes bastante experiencia dentro del sector musical, por lo que para iniciar quería preguntarte lo siguiente: ¿Actualmente manejas solo esta sala de ensayo hay alguna otra sala en la cual estes a cargo?

- **Entrevistado:** Actualmente si, antes teníamos otra sala aquí a unas dos cuadras, una sala aliada. Entonces, cuando no podía aquí se iban allá y viceversa. Por lo que, nos cruzábamos los clientes, debido que al principio hicimos esto con unos amigos que éramos socios. Finalmente, ellos como que se salieron del negocio y yo terminé comprando todo el equipamiento.

- **Entrevistador:** Ya. Entonces, avanzando un poco más, ¿cuál es el perfil de tus clientes en sí, tanto en género, en tal vez rangos de edad, si son músicos amateurs, si ya son profesionales? Más o menos, ¿cómo te manejas el perfil de tus clientes en sí?

- **Entrevistado:** La sala se renta para músicos que van a tener alguna presentación próximamente. Principalmente, esos son los clientes.

- **Entrevistador:** Los potenciales.

- **Entrevistado:** Ajá. Pero, más bien dicho, no los potenciales, sino los que vienen. Digamos que son los profesionales. Llámese profesionales a quienes se van a presentar ante un público próximamente. Pero también, hay otro tipo de clientes que son los que ensayan frecuentemente, por lo que ser profesional en la música es aquel el cual ensaya y no para de ensayar, sin necesidad de tener un título. Por lo general son bandas que están preparando temas para sus EP, grabaciones y todo. Entonces, a ese segmento de clientes se les ayuda con la preproducción, grabar ensayos y grabar maquetas, básicamente.

- **Entrevistador:** ¿Más o menos, los horarios que tienen más demanda?
- **Entrevistado:** En las noches, siempre en la noche.
- **Entrevistador:** ¿Existe mayor demanda de igual manera los fines de semana?
- **Entrevistado:** Ehh sí, pero es menos, porque como te digo los clientes preparan shows, y por ende los shows generalmente son viernes, sábado o jueves.
- **Entrevistador:** En el tema de las reservas generalmente, ¿Cuánto tiempo suelen durar? ¿Una hora, dos horas?
- **Entrevistado:** Por lo general son 2 horas.
- **Entrevistador:** ¿Qué incluye la hora de ensayo? y ¿Cómo influye en el costo del alquiler de la sala?
- **Entrevistado:** Incluye todo, la hora está a un precio de 13 dólares, pero incluye lo que necesites, equipo, grabación, edición, etc. Básicamente, es grabación y ensayo lo que estarías haciendo con tu grupo. Por lo general, yo grabo siempre.
- **Entrevistador:** Entonces, tú siempre los grabas, lo que se consideraría como tu plus al servicio.
- **Entrevistado:** Siempre esta microfoneado, incluso tengo equipo para transmitir en vivo, ya sea en Twitch, Facebook o Youtube. Pero, como ensayo muchas de las veces se tratan de información delicada la cual no puedes publicar. Yo lo que hago es, pues les entrego el material si es que quieren.
- **Entrevistador:** ¿Cuáles serían los servicios que te manejas? Además del alquiler de la sala para ensayos.
- **Entrevistado:** No me manejo por paquetes, por lo general son clientes frecuentes y clientes nuevos, entonces para los clientes nuevos es la tarifa de 13\$, sin embargo, para los clientes frecuentes que ocupan un mínimo de dos horas o 3 horas la sala de ensayo yo les cobro 10\$. Pero fines de semana y feriados son 13\$ la hora, incluyendo la grabación del ensayo, por lo que yo por cortesía les mando una mezcla del ensayo, ese es mi plus.
- **Entrevistador:** ¿Cómo gestionas actualmente las reservas de las salas?, además de ¿Por qué medios es que las personas llegan a conocer tu sala de ensayo? ¿Tienes alguna red social?, o es más por el boca y boca de las personas que te llegan a conocer.

- **Entrevistado:** Tengo red social para publicar lo que comparten los clientes, pero como te digo es full complicado compartir por mi cuenta los ensayos de los clientes, por lo que mantengo un perfil bajo en redes solo publicando lo que comparten mis clientes. Mayormente, me contactan y me hacen las reservas por whatsapp directamente. Por el tema de seguridad y delincuencia, siempre mi trabajo por mensajes de texto, no contesto llamadas, solo por gente que es conocida o que la tengo en mis contactos, ahora si es alguien que no conozco si acepto, pero ósea es con un poco de inseguridad. Entonces, mayormente los medios por los que me conoce la gente son más por boca a boca. Además, para contestarte lo primero que me preguntaste acerca de cómo gestiono mis reservas, es básicamente por Excel y herramientas como Google Calendar, me gusta utilizarlas.
- **Entrevistador:** Por otro lado, ¿Es frecuente que los clientes te encuentren por Google maps o por tu página dentro de las redes sociales?
- **Entrevistado:** Claro, pero al principio éramos solo una sala, pero al hacer publicidad se hizo como cuello de botella, todos querían venir aquí, se cruzaban demasiados horarios, tocaba manejar bien los horarios. Por lo que, una aplicación la cual nos permita distribuir a los clientes nos ayuda mucho, y por ejemplo si alguien quiere ensayar en determinado espacio este tiene que reservar.
- **Entrevistador:** Entiendo, justo te quería hacer una pregunta con respecto a las reservas ¿Con cuánto tiempo de anticipación sueles recibir las reservas?
- **Entrevistado:** Normalmente yo suelo pedir tres días de anticipación, dependiendo de cómo estén los horarios para organizarme. Sin embargo, si eres cliente frecuente y me dices, por ejemplo, si hoy noche tienes libre, si es que esta libre los recibo con gusto. Ósea es la disponibilidad de la sala y mío.
- **Entrevistador:** ¿Hasta qué hora suele ser el horario de atención?
- **Entrevistado:** Hasta las 11 de la noche, como manejo más reservas en la noche, a veces nos vamos hasta las 12 o 1 de la mañana, pero ya tiene que ser para un cliente frecuente así.
- **Entrevistador:** En temas de seguridad, ¿Has tratado de pedir más información del cliente para una reserva? Tal vez, como un numero de cedula, nombres completos, algún correo.

- **Entrevistado:** Con este tema de seguridad, si fuera bueno pedir esos datos, para garantizar una identidad de la persona, pero si existe como un recelo ante esa persona, la gente no da ese tipo de información, solo les eh preguntado de cómo te enteraste de nosotros y como te digo siempre ha sido así por boca a boca. Entonces, si esa es la manera de las reservas, si son gente nueva simplemente les pregunto de cómo se enteraron del sitio, si viene de algún referido y así.
- **Entrevistador:** Cambiando un poco de tema, ¿Cómo piensas que se debería manejar un control sobre la visualización de los horarios disponibles para los clientes sobre las salas de ensayo?
- **Entrevistado:** Principalmente, que los usuarios puedan ver la disponibilidad de las salas de ensayo, que puedan ver la hora, pero que no se vea el usuario que están utilizando el espacio, más bien como que el usuario pone la hora y le llegue un mensaje de si o no sobre la disponibilidad de la sala de ensayo, algo así pensaría yo, pero si existe otra manera de hacerlo estaría interesante.
- **Entrevistador:** ¿Cómo creerías que seria los pasos que un cliente podría seguir para encontrar una sala de ensayo en el proyecto a desarrollar?
- **Entrevistado:** Primeramente, que el cliente pueda buscar en base a su sector las posibles salas de ensayo disponibles, entonces se desplegarían las opciones disponibles cercanas al cliente, entonces tu podrías reservar. Yo estado pensando bastante en hacer un proyecto así de desarrollo sobre una aplicación de este tipo y sobre todo que sea para la reserva de las salas de ensayo.
- **Entrevistador:** Entonces una vez que el cliente se familiariza con la sala de ensayo, tal vez por la disponibilidad o cercanía del lugar ¿Qué otros factores consideras que permiten al usuario quedarse o preferir tu sala de ensayo a comparación de otras?
- **Entrevistado:** Una de las cosas por las cuales muchos de mis clientes vienen a mi sala de ensayo es por la comodidad que se les ofrece, la calidad del espacio es como un clic que se produce con cliente. Porque claro, hay músicos que vienen aquí, luego van a otras salas, esos serían los clientes más potenciales, porque es como que a mí me hablan directamente me dicen que baje mis precios y yo no puedo, por lo que, al hacer un sistema, o que se cobre mediante una aplicación sería más seguro que me paguen lo que está marcado ahí en la aplicación.

- **Entrevistador:** ¿Qué es lo que más te gustaría que el desarrollo de esta aplicación web pueda ayudar en tu sala de ensayo?
- **Entrevistado:** Ósea a mí me gustaría que la aplicación pueda llenarme los horarios muertos, porque yo por mi lado tengo mis clientes y esos clientes pueden o no utilizar mi aplicación.
- **Entrevistador:** Pero, por otro lado, los que están iniciando ellos no tienen mucha trayectoria como tú, por lo que ellos si pudieran sacar provecho a la aplicación.
- **Entrevistado:** Exacto porque me imagino que mediante la aplicación les permite a estas salas de ensayo que están empezando un poco a capturar a clientes los cuales por su ubicación requieran una sala de ensayo cercana a ellos.
- **Entrevistador:** José, ya que hablamos del manejo de tu sala de ensayo, ¿has considerado o te han pedido alguna vez otros métodos de pago, como transferencias o pagos en línea?
- **Entrevistado:** Sí, sí claro. Muchos músicos ya no andan con efectivo, así que prefieren pagar por transferencia. Yo no tengo problema mientras llegue el comprobante y esté claro el pago. Si la app tuviera esa opción directa de pago sería mucho más cómodo para mí poder ver quiénes son los que ya me pagaron y así.
- **Entrevistador:** Finalmente, ¿cómo ves el futuro del negocio de las salas de ensayo? ¿Crees que este tipo de herramientas digitales pueden cambiar la forma en que se gestionan?
- **Entrevistado:** Totalmente. Hoy en día todo es digital. Más aún para los músicos jóvenes, que ya están acostumbrados a hacer todo desde el celular. Si la aplicación funciona bien, podría facilitar mucho el trabajo de quienes tenemos una sala y también para los que están empezando. Por lo que, evolucionaría la actividad de reservar estas salas de ensayo, considerando la gran cantidad de personas que prestan sus espacios para hacer ensayos, sin embargo, sin llegar a muchas personas. Este tema de las salas tanto de grabación, como de ensayo siento que es como un nicho el cual no se explora mucho, pero que tiene potencial.

ANEXO 2 – ENTREVISTA A JOSUE GUERRA

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



Quito, 1 de mayo 2025

Mi nombre es **Jonathan Steve Rosero Garcia**, estudiante de la carrera de **Ingeniería de Sistemas de la Información** en la **Pontificia Universidad Católica del Ecuador**. Actualmente me encuentro desarrollando un trabajo de titulación titulado:

“Prototipo de aplicación web para la reserva y gestión de salas de ensayo musicales en Quito (QuitoJam)”, cuyo objetivo principal es diseñar una solución digital que facilite la administración y acceso a espacios de ensayo para músicos.

En este contexto, solicito su colaboración para participar en una **entrevista de carácter académico**, cuyo propósito es recopilar información valiosa relacionada con las necesidades, problemáticas y perspectivas de los potenciales usuarios o administradores de salas de ensayo.

Declaro expresamente que:

- La información proporcionada será utilizada **exclusivamente con fines académicos**.
- No será compartida con terceros ni utilizada con fines comerciales o de lucro.
- Los datos serán tratados con **confidencialidad y respeto a la privacidad** del entrevistado.
- La participación es **voluntaria**, y el entrevistado/a puede desistir en cualquier momento.

Agradezco de antemano su tiempo y disposición para contribuir al desarrollo de este proyecto.

Atentamente,



Jonathan Steve Rosero Garcia



Josue Guerra

Entrevista Transcrita:

- **Entrevistador:** Como estas, gracias por aceptar esta entrevista, vamos a comenzar un poco a presentarte, cuéntame un poco de ti ¿Hace cuánto tiempo haces música y qué tipos de proyectos musicales estás llevando actualmente?
- **Entrevistado:** Hola como estas, un gusto poder ayudarte, soy estudiante de fotografía y pues como lo sabes uno de mis hobbies favoritos es tocar la guitarra desde los 14 años. Pues ahorita tengo un proyecto con unas panas de la universidad, estamos tratando de armar una banda y estamos empezando apenas a repasar.
- **Entrevistador:** Gracias por tu presentación, vamos a iniciar con algunas preguntas de las cuales me vas a ayudar respondiendo para conocer un poco sobre tu perspectiva sobre como haces para ensayar con tu banda. Entonces empiezo con la siguiente pregunta ¿Con qué frecuencia ensayas con tu banda? De repente, ¿Ya tienen un lugar fijo donde ensayar?
- **Entrevistado:** Claro Steve mira nosotros ensayamos más o menos dos veces por semana cuando se puede, porque igual tenemos clases y trabajos, entonces a veces se complica. Sí solemos alquilar una sala, especialmente cuando ya tenemos un show cerca o queremos grabarnos algún demo. A veces también ensayamos en casa, pero no es lo mismo por el ruido y los vecinos, que siempre nos llaman la atención por el ruido, por lo que siempre tratamos de ensayar en una sala aparte.
- **Entrevistador:** Claro, es difícil ensayar en tu casa por estos temas de los vecinos y los equipos disponibles. Entonces, cuéntame ¿Cómo sueles encontrar una sala de ensayo? ¿Utilizas redes sociales, recomendaciones de amigos, o Google Maps?
- **Entrevistado:** Casi siempre por recomendación de otros músicos. O sea, nos pasamos el contacto por WhatsApp, y ahí gestiono con el dueño de la sala. Si he buscado en Google Maps, pero la mayoría ni tiene info, o el número está mal. Además, de que siempre no encuentro una sala que me quede cerca, tanto a mi como a la banda en general.
- **Entrevistador:** Interesante, porque siempre se necesita de un contacto para llegar hacia algún dueño de la sala, te cuento que igual en Google Maps, hay algunas salas, pero lo mismo la gran mayoría de salas no tiene actualizado sus datos, lo que nos dificulta el poder buscar un espacio para ensayar. En fin, ¿Cuándo reservas una

sala, ¿qué aspectos consideras más importantes? Por ejemplo, ubicación, precio, equipos, disponibilidad de horarios, etc.

- **Entrevistado:** Para mí lo principal es el precio y pues la distancia, para mi caso me interesa que exista un espacio cerca de mi ubicación, porque en cuanto a instrumentos nosotros tenemos. Después de eso, si cuenta con los equipos mínimos, por lo menos una batería completa y amplificadores, tanto para guitarras, bajo, además de los micrófonos, sería útil la verdad.

- **Entrevistador:** Esto nos lleva a la siguiente pregunta que esta algo relacionada, ¿Has tenido algún tipo de problema al momento de realizar a la reserva de tu sala de ensayo?

- **Entrevistado:** Sí, una vez llegamos y habían dado nuestra hora a otra banda porque se les cruzaron los mensajes. Como fue por WhatsApp y no confirmaron bien, paso eso. También a veces escribes y se demoran en responder, o no te dan todos los detalles. Por lo que, uno necesita saber en ese instante si ya se puede o no reservar, en el horario que yo quiero.

- **Entrevistador:** Perfecto, entonces en base a estas preguntas hasta el momento, si yo te digiera que existe una aplicación donde puedas ver salas disponibles, en donde puedes reservar directamente y buscar por zonas o características, ¿te resultaría útil? ¿Qué funciones crees que serían esenciales? Además de las que te acabo de decir.

- **Entrevistado:** Sí, la verdad sería muy útil. Lo ideal sería ver los horarios disponibles, poder reservar de manera fácil, sería chévere también poder hacer el pago directo desde la aplicación. También que se pueda ver fotos de la sala, los precios de las salas, que más podría ser, de repente el observar los equipos o alguna breve descripción sobre los equipos principales que tiene la sala.

- **Entrevistador:** Claro, información básica sobre la sala de ensayo, de igual manera ¿Te gustaría que la aplicación también permita dejar opiniones o calificaciones de las salas?, por lo que también te pregunto ¿En otras aplicaciones sueles confiar en ese tipo de referencias?

- **Entrevistado:** Claro que sí. Los comentarios me ayudan muchísimo antes de decidir, como también las estrellas de calificaciones como en Uber. Entonces en este caso de las reservas para las salas de ensayo, si una sala tiene malos

comentarios, ya uno se lo piensa dos veces. Por otro lado, también si está todo full recomendado, ves los buenos comentarios te animas más a reservar ese espacio.

- **Entrevistador:** Entonces sería muy importante para ti que se observen estos comentarios, como las calificaciones. Pero por otro tema, ¿Qué te parecería si esta aplicación te permitiera recibir notificaciones, como la confirmación de la sala de ensayo?

- **Entrevistado:** Me gustaría mucho. Dado que estas notificaciones me permiten saber que ya tengo esa reserva a mi nombre, y que ya el dueño ya me tiene guardado en su agenda para ese día.

- **Entrevistador:** En cuanto al tema de las búsquedas, ¿Te parecería útil si pudieras ver en un mapa las salas cercanas, además de que puedas agregar con filtros de búsqueda, como un rango de precios? Como lo mencionaste anteriormente que es algo importante de ver para ti.

- **Entrevistado:** Estaría muy bueno, porque de esta manera me podría ubicar mejor para ver las salas de ensayo más cercanas a mí, entonces considero que si esta super bueno que se pudiera enseñar en la sala de ensayo.

ANEXO 3: ACTA DE REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



Acta de los Requerimientos Funcionales y No Funcionales del proyecto

'QuitoJam'

Quito, 4 de mayo 2025

Proyecto: *QuitoJam – Sistema para la Reserva y Gestión de Salas de Ensayos Musicales en Quito*, elaborado por **Jonathan Steve Rosero Garcia**.

El presente documento tiene como finalidad dejar una constancia, como la validación y aceptación por parte de **José Torres** (Product Owner) y **Josué Guerra** (Usuario Final) de los requerimientos Funcionales / No – Funcionales que serán implementados en el sistema "QuitoJam", conforme el alcance establecido para el desarrollo del prototipo funcional

Tabla de Requerimientos Funcionales y No Funcionales	
Funcionales	No Funcionales
Búsqueda de salas por nombre, ubicación y características.	Interfaz responsiva (móvil - web)
Visualización de un calendario, con horarios disponibles para reservar salas.	La plataforma debe cargar resultados en menos de 3 segundos.
Realización y cancelación de reservas.	Autenticación segura vía Supabase, garantizando la seguridad de datos personales.
Vista detallada de las salas de ensayo, con fotos, descripción, ubicación en el mapa).	Soporte para la autenticación por correo electrónico.
Registro/Inicio de sesión para usuarios clientes (músicos) y administradores	Uso de lenguaje claro, directo y accesible para el público general.
Panel de gestión para administradores (CRUD de salas, horarios, reservas).	
Uso de geolocalización para autocompletar direcciones.	
Subir imágenes de las salas de ensayo.	
Posibilidad de añadir calificaciones y comentarios.	
Sistema de Pagos (simulado)	

Jonathan Rosero

José Torres

Josué Guerra

ANEXO 4: ACTA DE REVISIÓN SPRINT 1 USUARIO CLIENTE:



ACTA DE REVISIÓN DEL SPRINT 1 – USUARIO CLIENTE

Proyecto: QuitoJam

Sprint: Acceso y exploración de salas.

Usuario: Josué Guerra

Fecha: 05/09/2025

Tabla de Verificación:

CÓDIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
001.1	Pantalla de login/registro para cliente	X	
001.2	Validación de Credenciales	X	
001.3	Pantalla de visualización de Salas de Ensayo (cliente)	X	
001.4	Pantalla para usuarios invitados con visualización limitada	X	
002.1	Menú desplegable según tipo de usuario	X	
008.3	Seguridad en acceso a pantallas y rutas según tipo de usuario	X	

Observaciones/Recomendaciones:

sin observaciones

Jonathan Rosero

Josué Guerra

ANEXO 5: ACTA DE REVISIÓN SPRINT 1 USUARIO ADMINISTRADOR

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE REVISIÓN DEL SPRINT 1 – USUARIO ADMINISTRADOR

Proyecto: QuitoJam

Sprint: Acceso como administrador y gestión básica

Usuario: José Torres

Fecha: 05/09/2025

Tabla de Verificación:

CÓDIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
001.1	Pantalla de login/registro para administrador	✓	
001.2	Validación de credenciales	✓	
002.1	Menú desplegable según tipo de usuario	✓	
008.3	Seguridad en acceso a pantallas y rutas según tipo de usuario	✓	

Observaciones/Recomendaciones:

Como recomendación, manejar la información de manera privada.


Jonathan Rosero

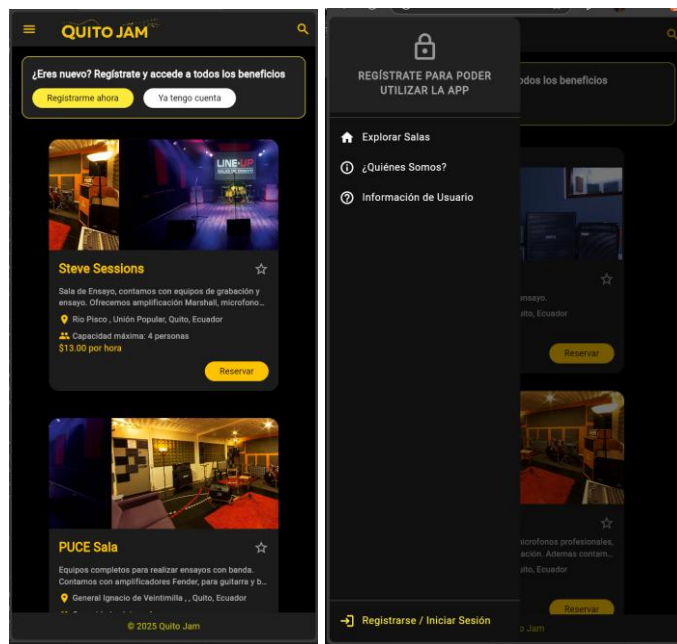

José Torres

ANEXO 6: PRODUCTOS ENTREGADOS SPRINT 1

Pantalla Welcome:



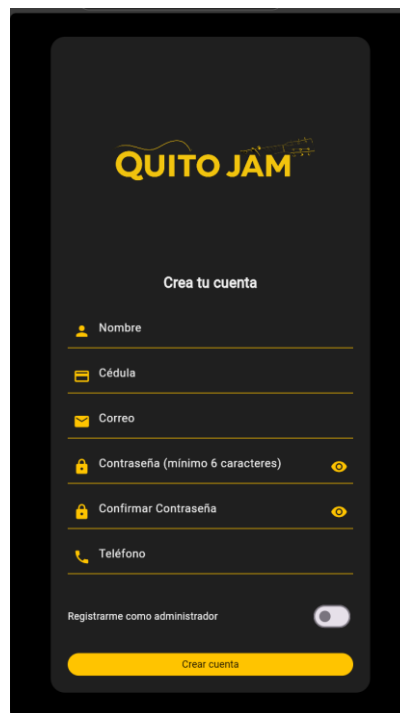
Pantalla ShowRooms – Menu (Invitado):



Pantalla Login:



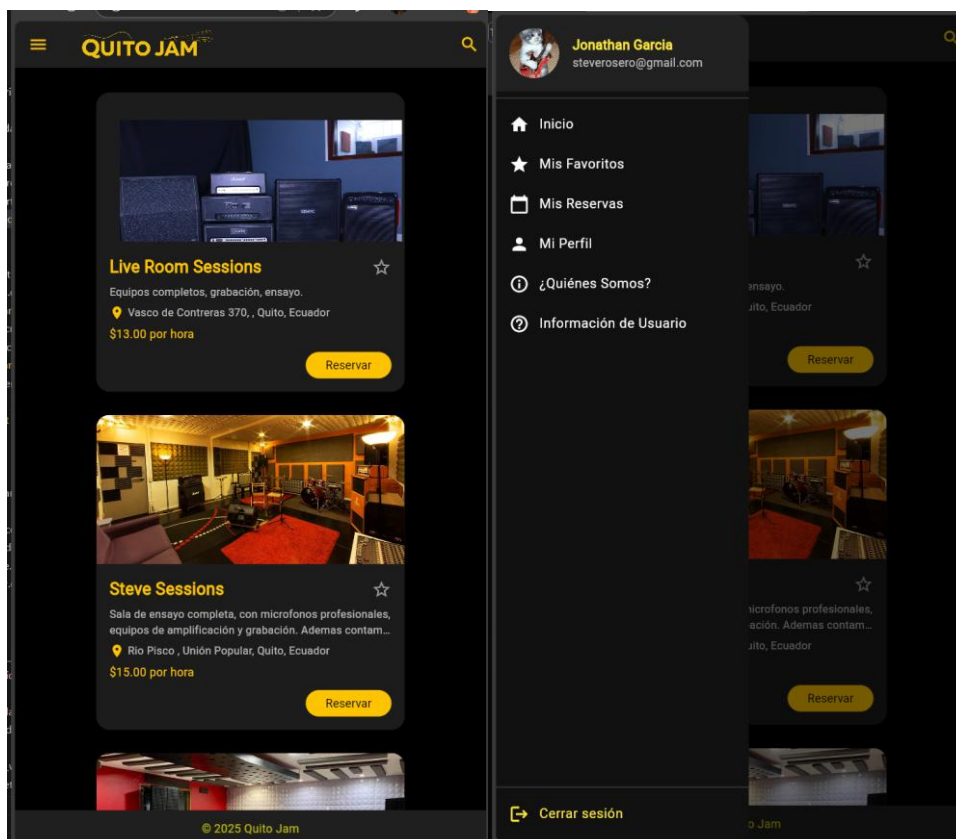
Pantalla Registro:



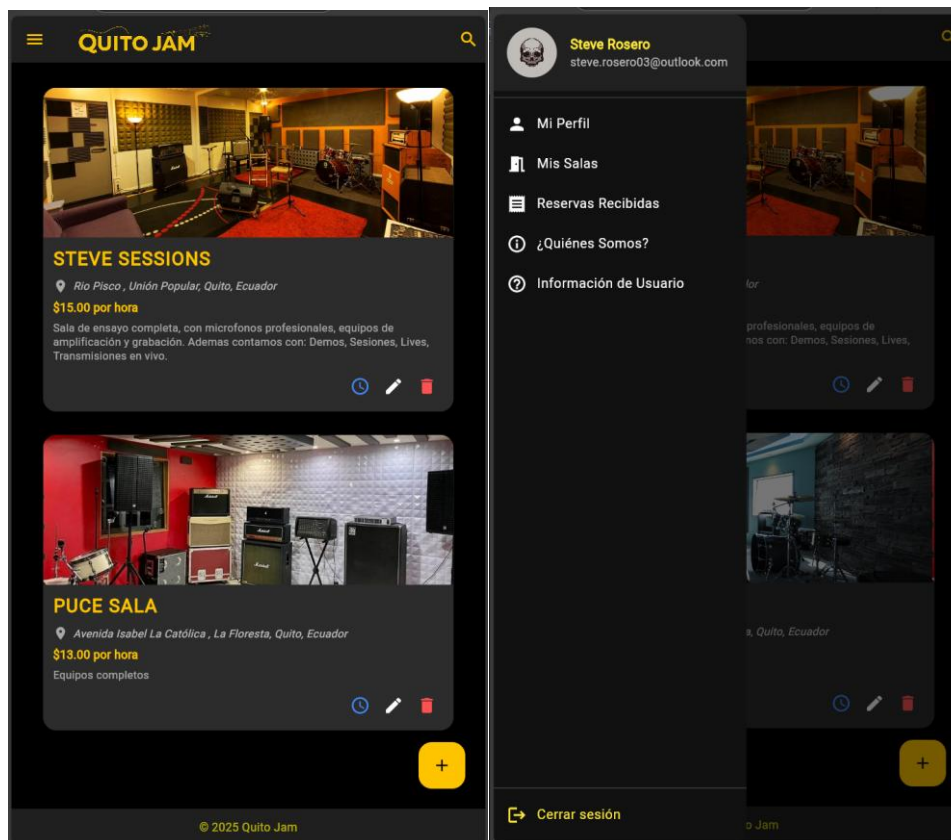
Pantalla Términos y Condiciones (Cliente – Administrador):



Pantalla Showrooms – Menu (Tipo de usuario: Cliente)



Pantalla Mis Salas – Menu (Tipo de usuario: Administrador)



ANEXO 7: ACTA DE REVISIÓN USUARIO CLIENTE SPRINT 2:

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE REVISIÓN DEL SPRINT 2 – USUARIO CLIENTE

Proyecto: QuitoJam

Sprint: Búsqueda, reserva y Gestión de Salas de Ensayo.

Usuario: Josué Guerra

Fecha: 05/23/2025

Tabla de Verificación:

CÓDIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
002.3	Cards de presentación de salas de ensayo para clientes, invitados y administradores.	X	
003.1	Pantalla de detalles de sala: descripción, ubicación, mapa, galería de imágenes, etc.	X	
004.1	Selección de fecha y hora para la reserva.	X	
004.2	Validación de disponibilidad de los horarios antes de ejecutar la reserva.	X	
006.2	Búsqueda de salas por nombre, ubicación o características.	X	
006.3	Búsqueda de salas por cercanía usando la ubicación del dispositivo.	X	

Observaciones/Recomendaciones:

Sin observaciones


Jonathan Rosero


Josué Guerra

ANEXO 8: ACTA DE REVISIÓN USUARIO ADMINISTRADOR SPRINT 2

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE REVISIÓN DEL SPRINT 2 – USUARIO ADMINISTRADOR

Proyecto: QuitoJam

Sprint: Búsqueda, reserva y Gestión de Salas de Ensayo.

Usuario: José Torres

Fecha: 05/23/2025

Tabla de Verificación:

CÓDIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
002.2	CRUD completo para la administración de salas de ensayo (crear, editar, eliminar, visualizar).	✓	
002.3	Cards de presentación de salas de ensayo para clientes, invitados y administradores.	✓	
002.4	Pantalla para la gestión de horarios disponibles por sala, en el apartado del administrador.	✓	
003.1	Pantalla de detalles de sala: descripción, ubicación, mapa, galería de imágenes, etc.	✓	
003.2	Subida de imágenes de las salas con previsualización.	✓	
006.1	Autocompletado de dirección por geolocalización en formulario de creación de sala.	✓	

Observaciones/Recomendaciones:

Solamente tener cuidado con la información pública.


Jonathan Rosero


José Torres

ANEXO 9: PRODUCTOS ENTREGADOS SPRINT 2

CRUD Completo, para la administración de las salas de ensayo / Cards de presentación de las salas de ensayo por tipo de usuario (Admin), pantalla 'Mis Salas':



Card de presentación, para los dueños de la sala.

Botón para editar la sala de ensayo.

Botón para la eliminación del registro de la sala.

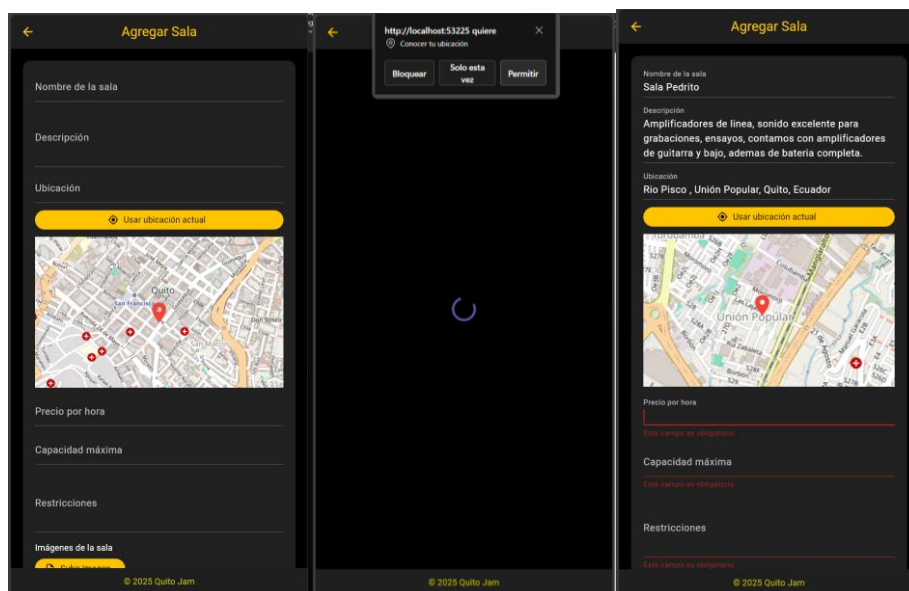
Botón para visualizar el registro de la sala

Botón para acceder a la gestión de horarios

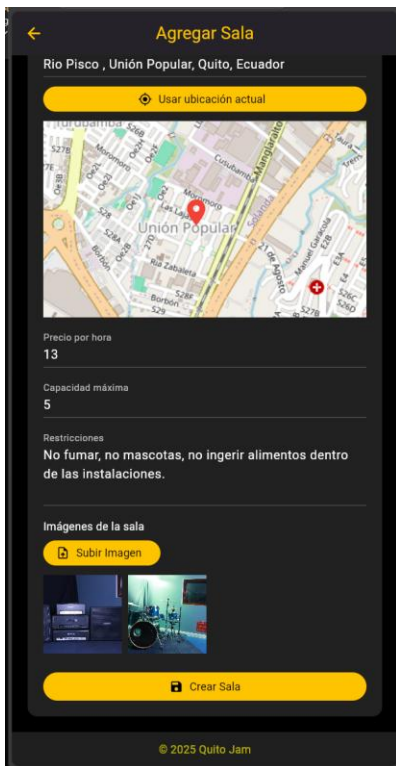
Botón para la creación de una sala de ensayo.

Pantalla para la Creación de una Sala

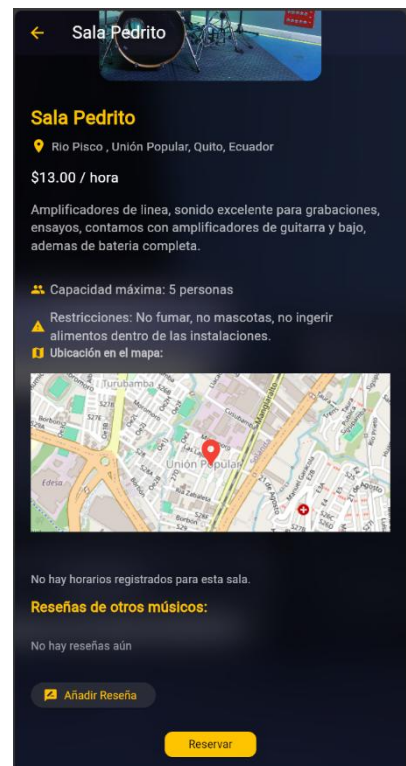
'crear_editar_sala', formulario, solicitud de llenar campo de ubicación:



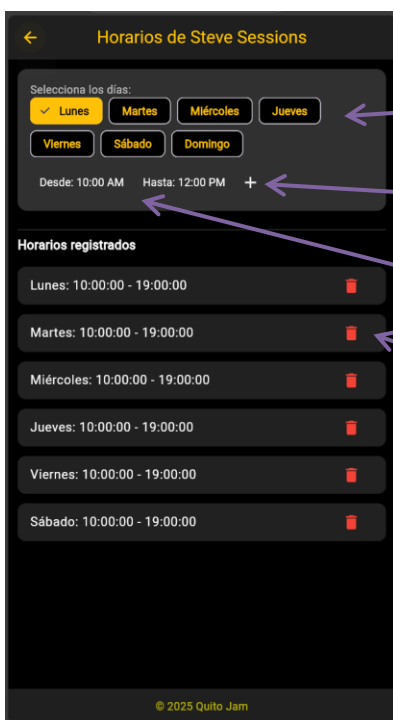
Apartado de subir imágenes con vista previa:



Pantalla con los detalles de la sala 'detalles_sala':

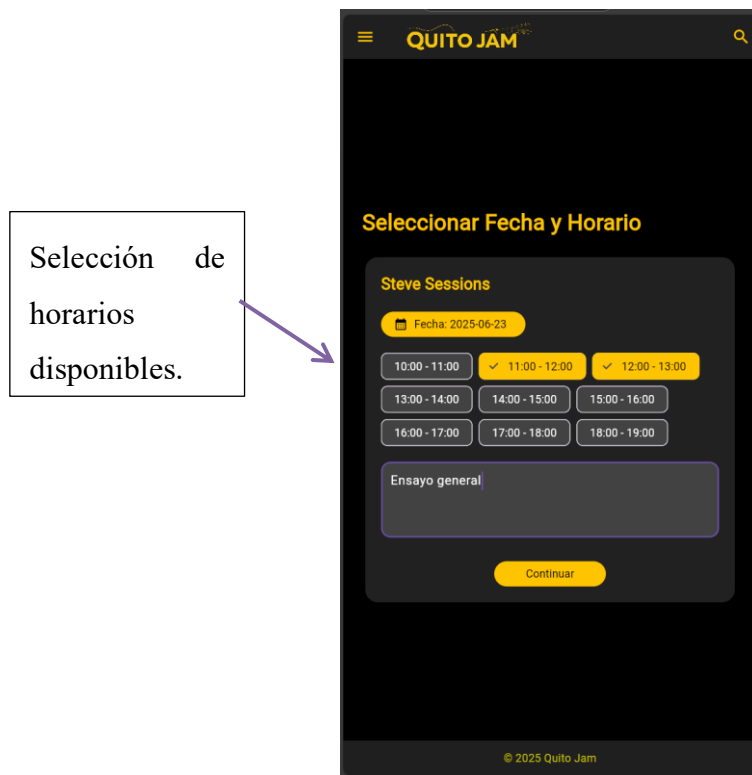
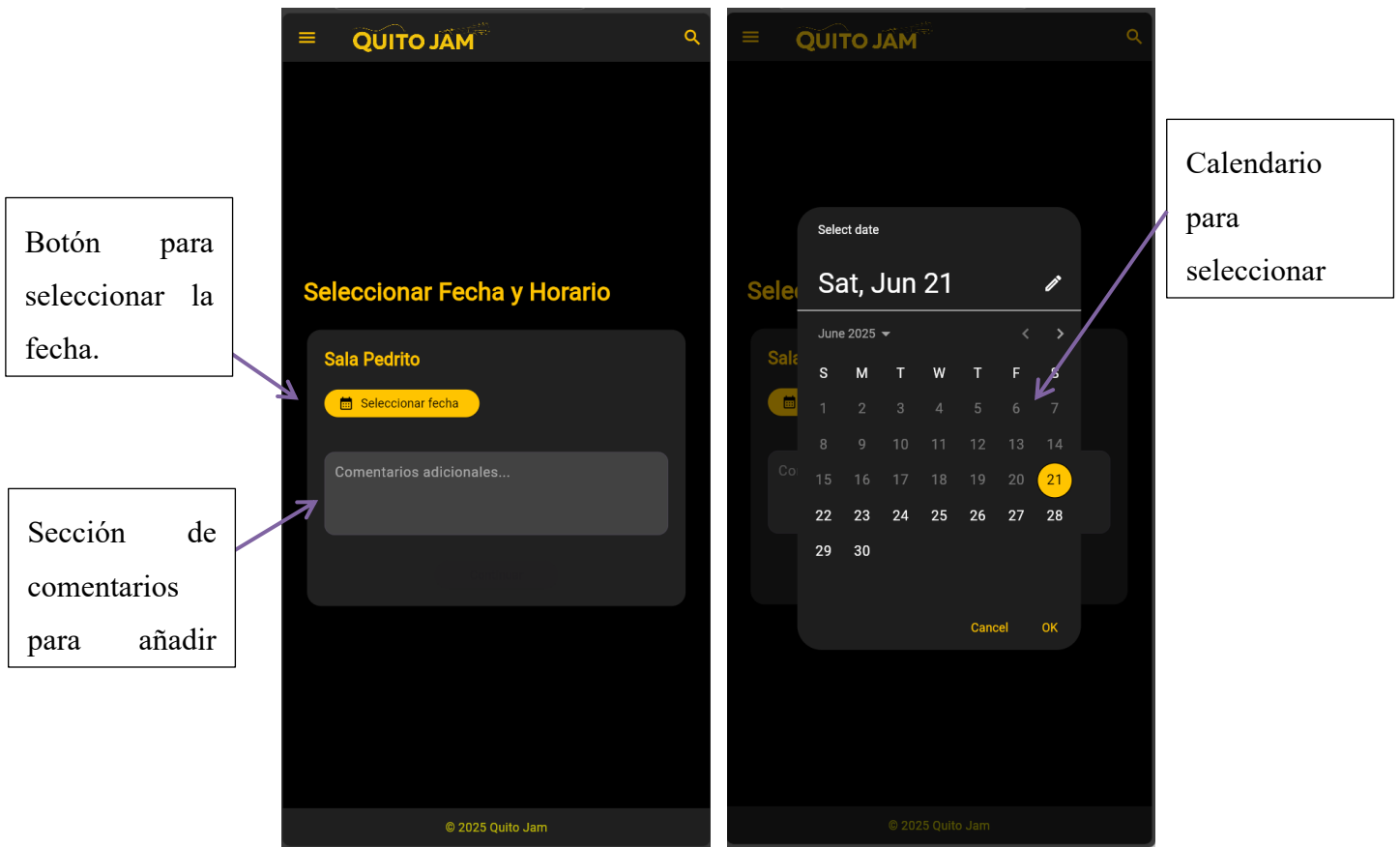


Pantalla para la Gestión de Horarios Disponibles por Sala, pantalla 'horarios'.



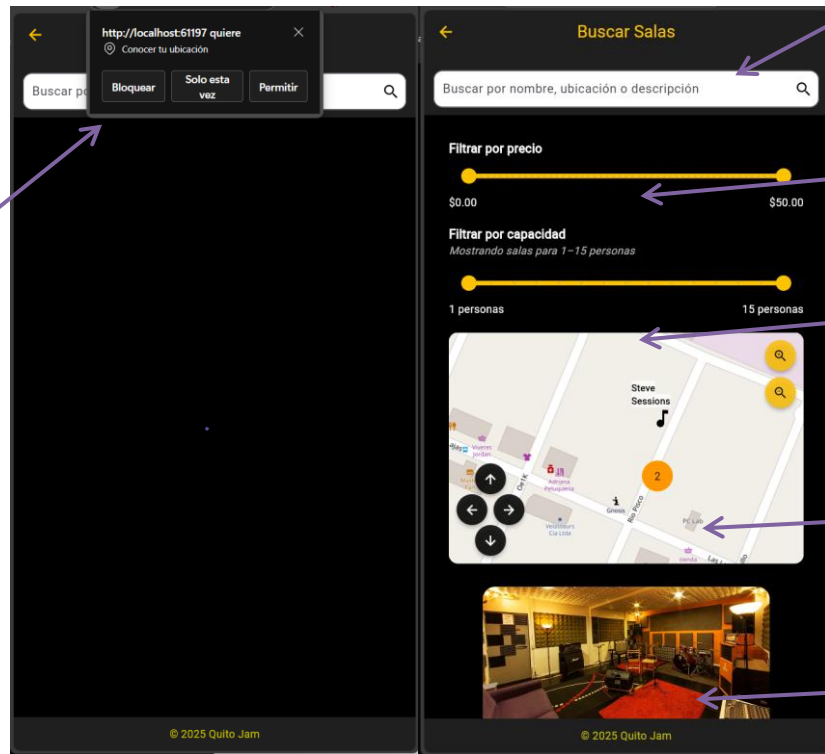
- ← Selección de día o días
- ← Botón de crear horario
- ← Rango de Horario
- ← Eliminación de horario

Pantalla para Seleccionar las Fechas de la Reserva ‘agregar_editarSalas’ y detalle final de la reserva (sin poder pagar) ‘reserva_pago_screen’:



Pantalla de Búsqueda:

Se solicita la ubicación al usuario para



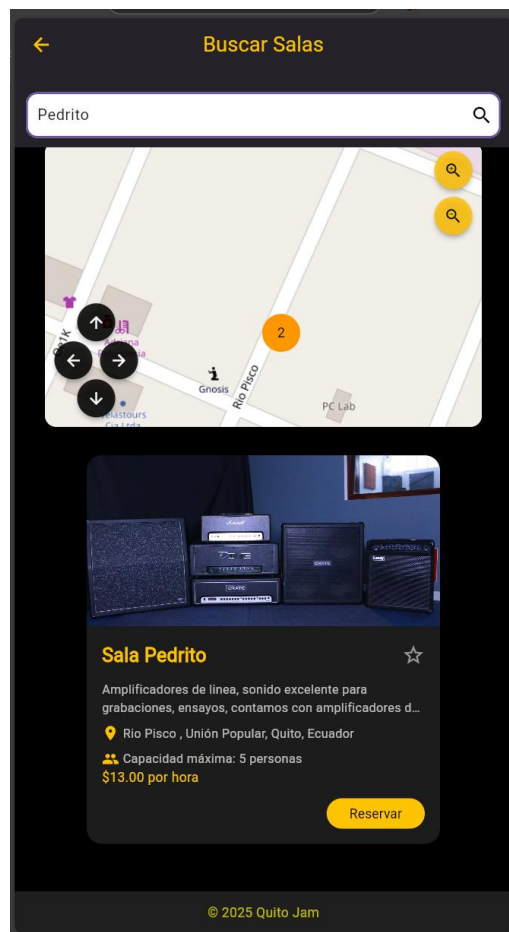
Barra de búsqueda para encontrar

Filtro por precios

Filtros por capacidad

Mapa para ubicación

Lista de las salas más cercanas.



ANEXO 10: ACTA DE REVISIÓN USUARIO CLIENTE SPRINT 3

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE REVISIÓN DEL SPRINT 3 – USUARIO CLIENTE

Proyecto: QuitoJam

Sprint: Pago, Gestión de Reservas y Apartado Administrativo.

Usuario: Josué Guerra


Fecha: 05/30/2025

Tabla de Verificación:

CÓDIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
004.3	Cancelación de reservas con confirmación y motivo	X	
004.4	Vista de historial de reservas para el usuario (pasadas y futuras).	X	
007.3	Función automática de actualización de estado de reservas (en curso, finalizada).	X	
008.2	Lógica de control para evitar reservas fuera del horario establecido.	X	
008.3	Integración (simulada) de confirmación visual de pago.	X	

Observaciones/Recomendaciones:

Sin Observaciones


Jonathan Rosero


Josué Guerra

ANEXO 11: ACTA DE REVISIÓN USUARIO ADMINISTRADOR SPRINT 3

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE REVISIÓN DEL SPRINT 3 – USUARIO ADMINISTRADOR

Proyecto: QuitoJam

Sprint: Pago, Gestión de Reservas y Apartado Administrativo.

Usuario: José Torres

Fecha: 05/30/2025

Tabla de Verificación:

CÓDIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
004.4	Vista de historial de reservas para el usuario (pasadas y futuras).	✓	
005.1	Panel de reservas recibidas por administrador.	✓	
005.2	Resumen administrativo: reservas en curso, canceladas, reembolsadas, total.	✓	
007.3	Función automática de actualización de estado de reservas (en curso, finalizada).	✓	
008.2	Lógica de control para evitar reservas fuera del horario establecido.	✓	

Observaciones/Recomendaciones:

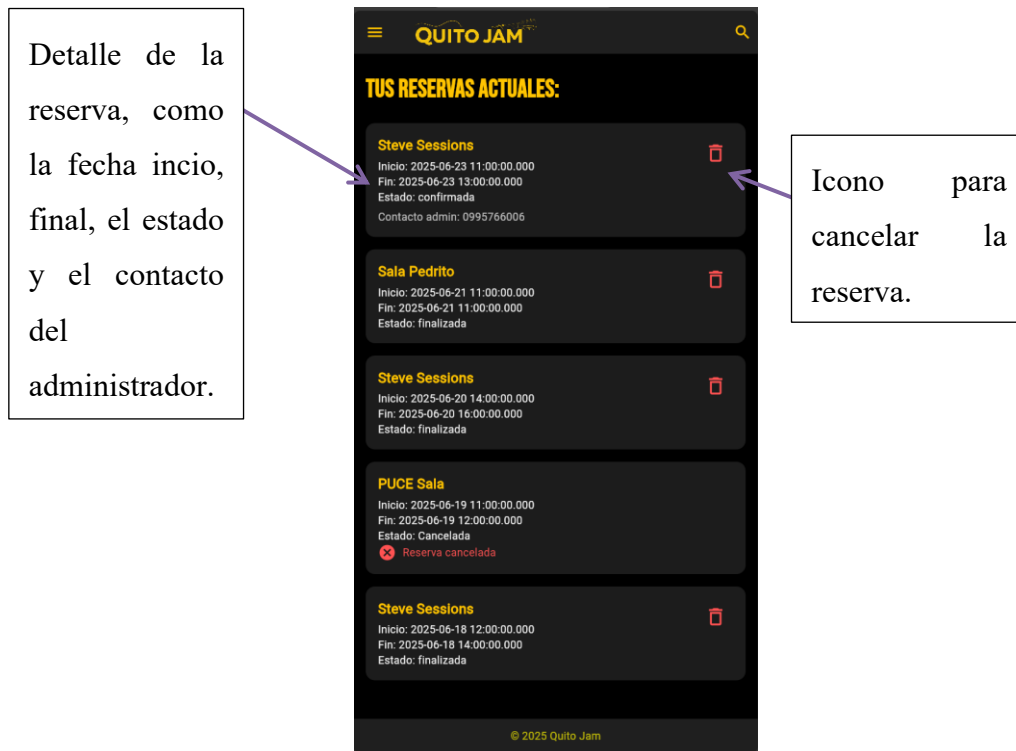
• Sería bueno que se ideen maneras de mantener al usuario usando la app y no haya fugas de usuarios al tener los contactos de las palas.

Jonathan Rosero

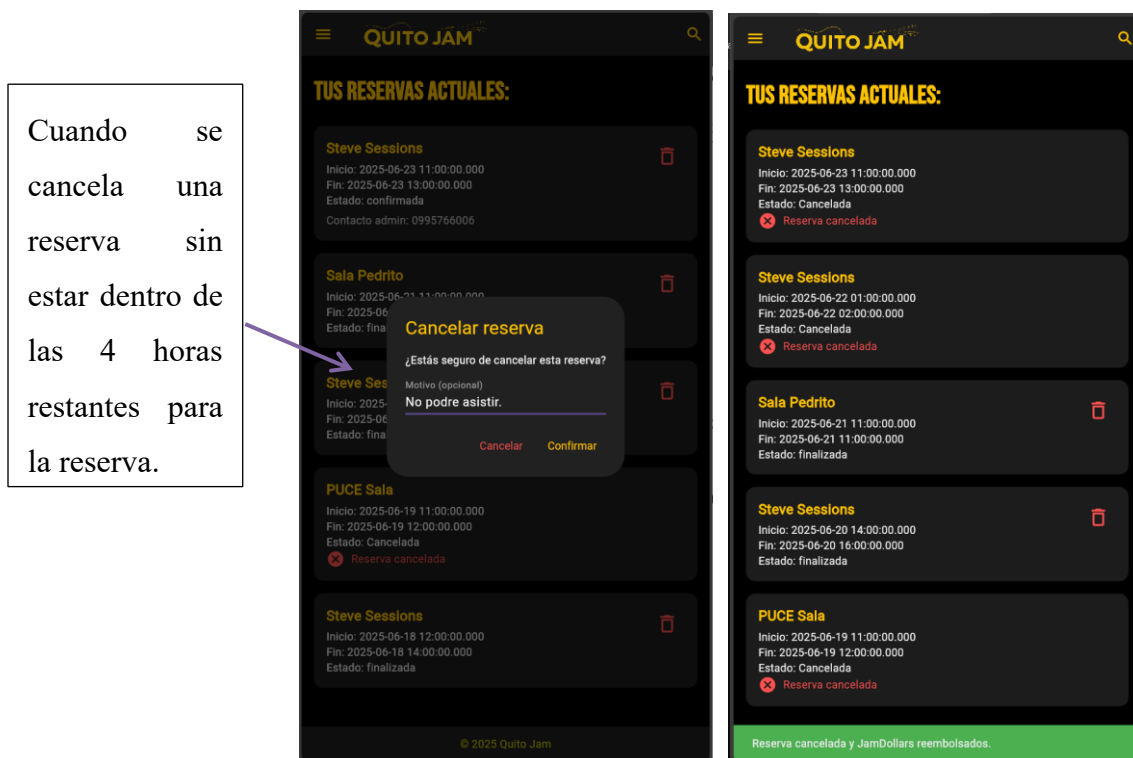
José Torres

ANEXO 12: PRODUCTOS ENTREGADOS SPRINT 3

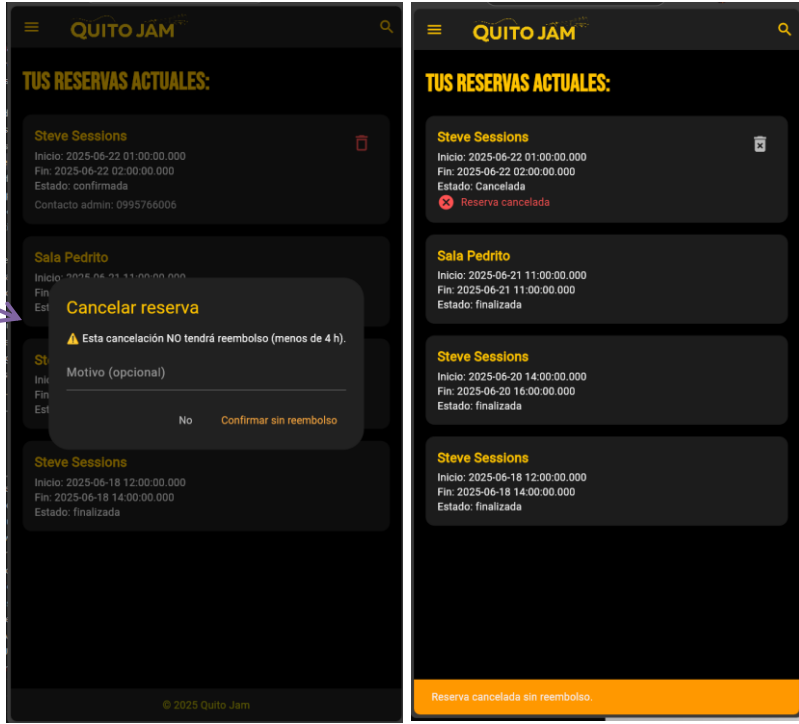
Pantalla del Historial de Reservas para el Usuario Cliente (pasadas y futuras) 'mis_reservas'.



Proceso de Cancelación de una Reserva.



Cuando se cancela una reserva estando antes de las 4 horas de la reserva.



Control para evitar reservas fuera del horario establecido.

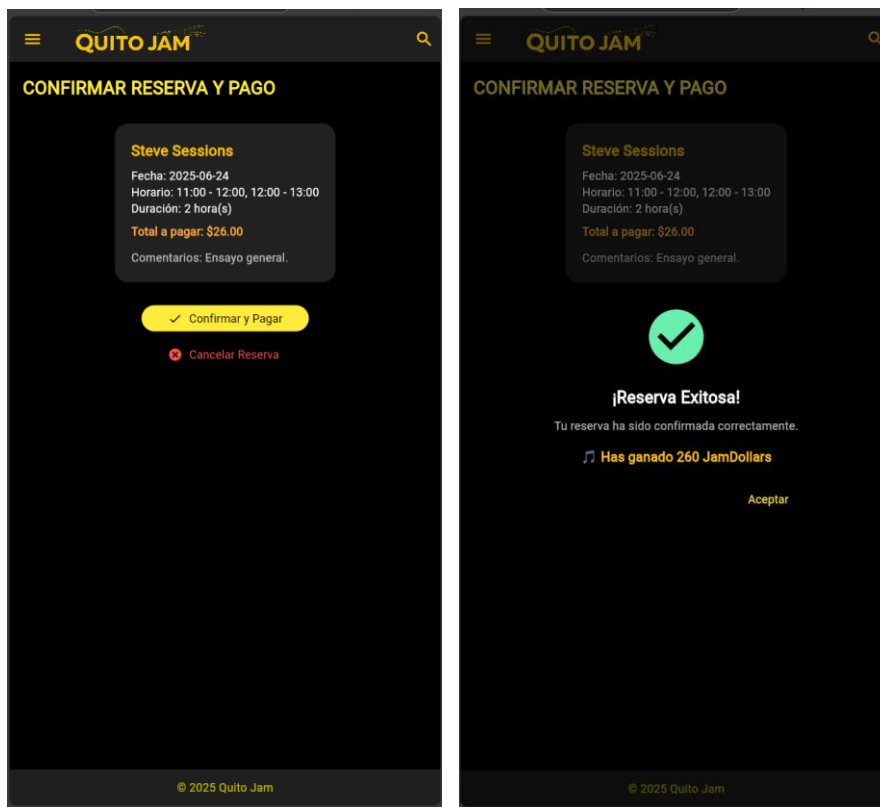
Una vez que se selecciona y se confirma la sala se bloquea



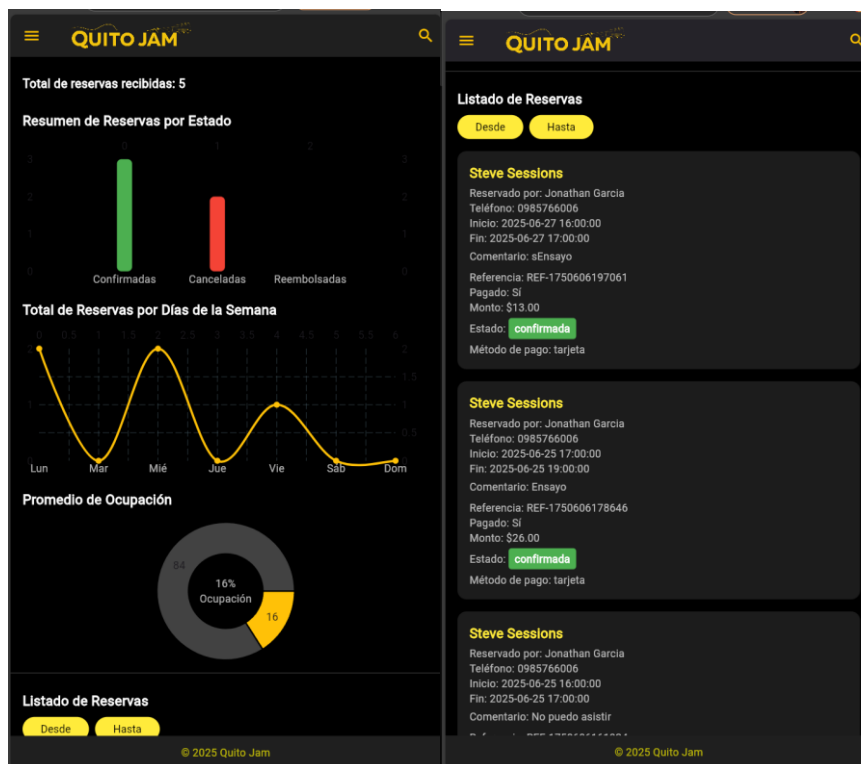
Una vez que se realice la reserva no se mostrara dichos horarios



Confirmación de Pago al momento de completar una reserva, 'reserva_pago_screen'.



Resumen administrativo (dueño de sala) sobre la actividad de las salas, 'reservas_recibidas'.



ANEXO 13: ACTA DE REVISIÓN USUARIO CLIENTE SPRINT 4

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE REVISIÓN DEL SPRINT 4 – USUARIO CLIENTE

Proyecto: QuitoJam

Sprint: Personalización, reseñas y fidelidad en la aplicación.

Usuario: Josué Guerra

Fecha: 06/05/2025

Tabla de Verificación:


CÓDIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
007.1	Eliminación de cuenta del usuario desde pantalla de perfil.	X	
007.2	Actualización de datos de perfil (foto, nombre, etc.).	X	
009.1	Sistema de calificación y reseñas por sala de ensayo.	X	
009.2	Notificaciones en tiempo real sobre las reservas hechas en la sala de ensayo.	X	
009.3	Implementación de JamDollars, para promociones incentivando para incentivar el uso frecuente de la App.	X	

Observaciones/Recomendaciones:

Sin observaciones



Jonathan Rosero



Josué Guerra

ANEXO 14: ACTA DE REVISIÓN USUARIO ADMINISTRADOR SPRINT 4

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE REVISIÓN DEL SPRINT 4 – USUARIO ADMINISTRADOR

Proyecto: QuitoJam

Sprint: Personalización, reseñas y fidelidad en la aplicación.

Usuario: José Torres

Fecha: 06/05/2025

Tabla de Verificación:

CODIGO	FUNCIÓN ENTREGADA	CUMPLE	NO CUMPLE
007.1	Eliminación de cuenta del usuario desde pantalla de perfil.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
007.2	Actualización de datos de perfil (foto, nombre, etc.).	<input checked="" type="checkbox"/>	<input type="checkbox"/>
009.2	Notificaciones en tiempo real sobre las reservas hechas en la sala de ensayo.	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Observaciones/Recomendaciones:

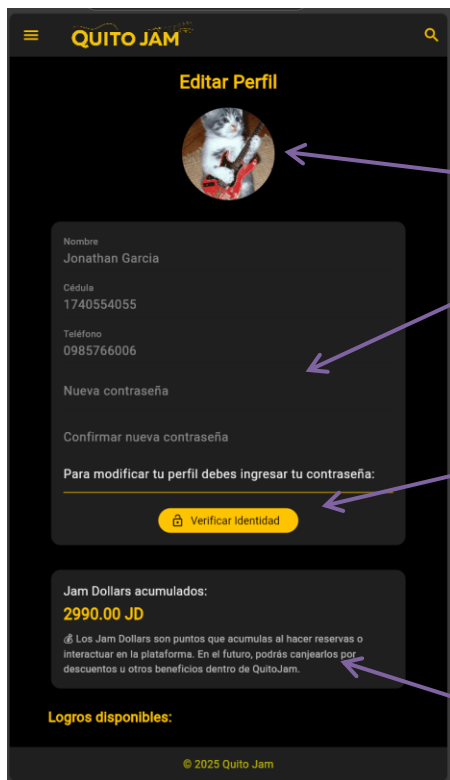
Sin comentarios
La app tiene mucho potencial.

Jonathan Rosero

José Torres

ANEXO 15: PRODUCTOS ENTREGADOS SPRINT 4.

Pantalla Perfil (usuario cliente), con personalización en aspectos de perfil y los JamDollars.



Puedes personalizar tu perfil una vez que valides tu

Primero tienes que validar tu contraseña para realizar

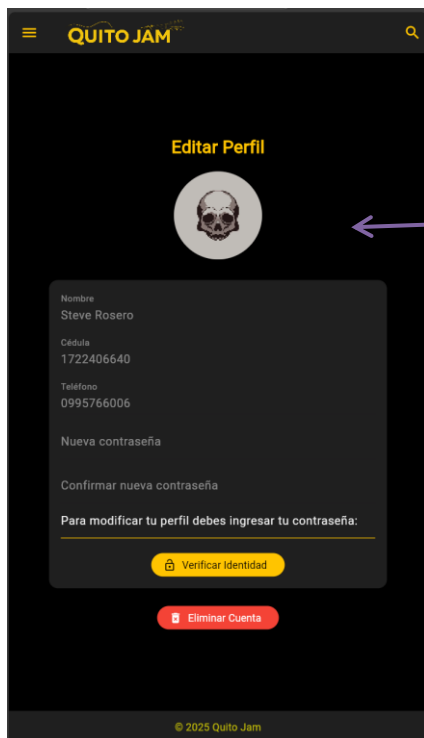
Cantidad de JamDollars recolectados por las reservas realizadas.



Recompensas canjeables con JamDollars.

Botón para la eliminación de la

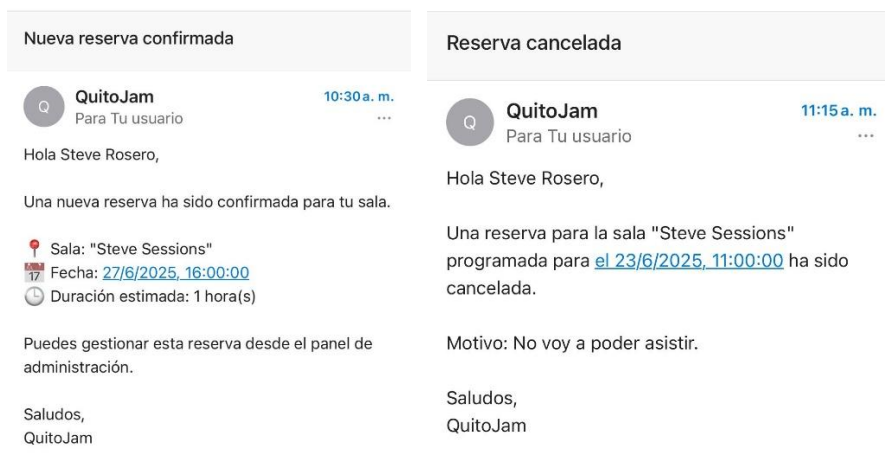
Pantalla Perfil (usuario administrador):



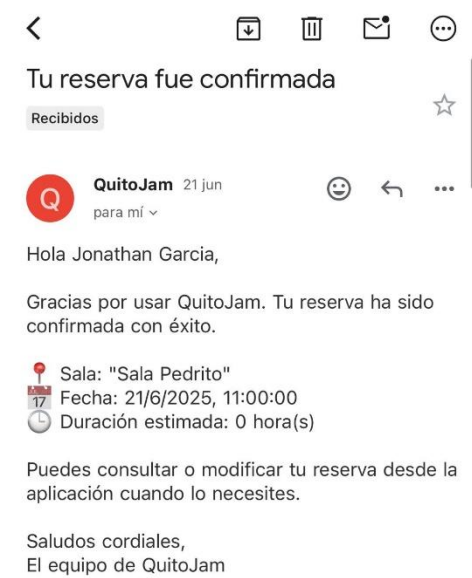
Se limitan los JamDollars únicamente hacia los usuarios de tipo cliente.

Screenshots sobre las notificaciones que se reciben al realizar una reserva

Capturas obtenidas desde Outlook, correspondiendo al correo de un usuario administrador (dueño de sala).



Capturas obtenidas desde Gmail, correspondiendo al correo de un usuario de tipo cliente.



ANEXO 16: DICCIONARIO DE DATOS.

Tabla 'usuario'

- **'id_usuario' (uuid):** Funciona como identificador único del usuario, se lo utiliza como clave primaria. Además, al ser de tipo 'uuid' se garantiza un mejor nivel de seguridad, como de privacidad, a diferencia de un int autoincrementar.
- **'nombre' (varchar):** Atributo el cual almacena el nombre del usuario. El cual se muestra en distintas partes del sistema, como en el perfil y en el panel del administrador.
- **'cedula' (varchar):** Es el campo el cual utilizamos tanto para la identificación del usuario, como de seguridad para los miembros de la aplicación.
- **'correo' (varchar):** Usado principalmente para verificar y autenticar la cuenta, además de que se necesita un correo para poder recibir notificaciones por parte de la aplicación.
- **'telefono' (varchar):** Nos permite tener un contacto directo con el usuario, como con el administrador, una vez se haya pagado por la reserva.
- **'tipo_usuario' (varchar):** Es el atributo el cual define el rol del usuario (cliente, admin) con el cual el usuario se haya registrado. Este atributo nos permite tener un control de accesos, menús de navegación y permisos, como acciones dentro de la aplicación.

- **'fecha_creación' (timestamp):** Es la fecha de creación del usuario. Funciona como un estándar para la BDD, además de que nos permite tener un registro de la creación de la cuenta del usuario.
- **'foto_perfil' (text):** Atributo el cual guarda la URL (usando Supabase Storage, mediante un bucket) de la imagen, mejora la personalización y experiencia del usuario.

Tabla 'saladeensayo'

- **'id_sala' (int4):** Es la clave primaria de la tabla, la cual nos ayuda a identificar cada una de las salas. Además, de que lo utilizamos para las relaciones con horarios y reservas.
- **'nombre_sala' (varchar):** Es el nombre de la sala, el cual utilizamos para mostrarlo en las tarjetas y búsqueda de esta.
- **'ubicacion' (text):** Es la dirección escrita completa de la sala de ensayo, la cual permite mostrar al usuario la dirección con los nombres de las calles, ciudad, etc. Se lo utiliza como tipo 'text' al ser un campo con contenido largo indefinido, lo que nos permite mayor flexibilidad.
- **'latitud' (numeric):** Corresponde a la coordenada geográfica, la cual se utiliza para la geolocalización y búsqueda por área de la sala.
- **'longitud' (numeric):** Corresponde de igual manera a la coordenada geográfica la cual complementa la latitud, para la ubicación del mapa.
- **'precio_por_hora' (numeric):** Es el valor que representa el costo de alquiler por hora, se lo establece en el formulario de creación de la sala.
- **'descripcion' (text):** Es la información adicional la cual puede mejorar la presentación de la sala hacia los usuarios clientes. Presentando equipos existentes en la sala, capacidad máxima de personas, reglas del establecimiento, etc.
- **'id_propietario' (uuid):** Es una clave foránea la cual establece una relación directa con la tabla 'usuario'. Este campo nos permite identificar el administrador dueño de la sala.
- **'fecha_registro' (timestamp):** Es el campo el cual nos ayuda a saber la fecha en la cual fue creada la sala. Útil al momento de ordenar las salas por antigüedad en la aplicación.
- **'imagenes' (text):** En este campo se almacenan las listas de URLs generadas de las imágenes guardadas en Supabase (de igual manera se utiliza otro tipo de bucket dentro

de Supabase para su almacenamiento), se manejan en listas, al ser varias las imágenes que se muestran en la presentación de cada sala de ensayo.

- **'fts' (tsvector):** Es un campo especial, enfocado para la búsqueda de texto completo (full-text search). En donde, se mejora la búsqueda por nombre, palabras clave, descripción, etc.

Tabla 'horario'

- **'id_horario' (int4):** Corresponde a la clave primaria por cada franja horaria registrada.
- **'id_sala' (int4):** Es la clave foránea que relaciona directamente con la tabla 'saladeensayo'. Por lo que nos permite definir múltiples horarios por sala.
- **'dia_semana' (varchar):** Corresponde a los días de la semana, dicho atributo facilita la visualización, como el control específico por día.
- **'hora_inicio' (time):** Corresponde a la hora de inicio sobre la disponibilidad de las salas.
- **'hora_fin' (time):** Corresponde a la hora de finalización de las reservas para las salas. Funciona, para limitar el horario por día de las salas, y de esta manera poder tener una lista de horas por día que maneja la sala de ensayo.
- **'estado' (bool):** Indica si el horario que se selecciona está disponible (true) o no (false). Esto nos ayuda a que al momento de que un usuario ya haya seleccionado ese horario en específico, no pueda aparecer como disponible en otra sesión de usuario.

Tabla 'reserva'

- **'id_reserva' (int4):** Corresponde a la clave primaria de la reserva, lo que vuelva a cada reserva como única.
- **'id_usuario' (uuid):** Es la clave foránea la cual relaciona directamente con el usuario el cual hace la reserva.
- **'id_sala' (int4):** Es otra clave foránea, que de igual manera nos permite relacionar directamente con la sala de ensayo en la cual se está gestionando la reserva.
- **'fecha_inicio' (timestamp):** Corresponde a la fecha y hora en la que comienza la reserva. Se usa para la validación y orden cronológico.
- **'fecha_fin' (timestamp):** De igual manera corresponde a la fecha y hora, pero del final de la reserva. Junto al campo de inicio nos ayuda a definir la duración de la reserva.
- **'estado' (varchar):** Corresponde al estado de la reserva (ej "pendiente", "en curso", "finalizada", "cancelada").

- **'comentarios' (text):** Campo en donde el usuario tiene la libertad de dejar observaciones adicionales para su reserva (ej “voy a hacer una grabación de una canción”, “voy a ensayar con mi banda”, etc).
- **'fecha_reserva' (timestamp):** Corresponde a la fecha y hora en la cual se realizó la reserva, útil para temas de trazabilidad.
- **'cancelada' (bool):** Indica si la reserva fue cancelada (true) o no (false). Lo que nos permite manejar una lógica sin eliminar datos.

Tabla 'pago'

- **'id_pago' (int4):** Corresponde a la clave primaria del registro de pago.
- **'id_reserva' (int4):** Es la clave foránea la cual nos permite tener una relación directa con la reserva correspondiente. Lo que nos asegura una trazabilidad directa.
- **'metodo_pago' (text):** Describe el método simulado utilizado (Directo, JamDollars).
- **'referencia_pago' (text):** Es un identificador de la transacción, el cual es útil para la simulación o registro futuro real.
- **'pagado' (bool):** Indica si el pago realizado (true) o si sigue pendiente (false).
- **'fecha_pago' (timestamp):** Fecha y hora en la cual se realizó el registro del pago. Tiene su importancia al momento de realizar un control administrativo.
- **'monto' (numeric):** Refleja el monto total que tuvo la reserva, en base a las horas seleccionadas y el precio de cada hora.

Tabla 'jamdollars'

- **'id' (int4):** Es el identificador único para cada transacción de JamDollars, hacia la cuenta del usuario.
- **'id_usuario' (uuid):** Es el identificador del usuario para que pueda acumular JamDollars, es la relación directa que tiene la tabla con el usuario, para así determinar su cantidad de JamDollars por usuario.
- **'monto' (numeric):** Corresponde al valor en JamDollars (por cada 1 USD = 10 JamDollars).
- **'descripcion' (text):** Motivo de la transacción (ej: “Reserva”, “Canje”).
- **'fecha' (timestamp):** Corresponde a la fecha y hora exacta del movimiento.

Tabla 'resena'

- **'id' (uuid):** Corresponde al identificador único del comentario.

- **'id_sala'(int4):** Corresponde a la sala, de la cual se hace la reseña o comentario.
- **'usuario' (uuid):** Es el usuario el cual realizó la reseña.
- **'comentario' (text):** Corresponde al comentario escrito por el usuario, poniendo su opinión de la sala.
- **'calificación' (int4):** Es una puntuación otorgada para la sala, tiene un ranking de 1 a 5, siendo 5 la calificación más alta de recomendación de la sala.
- **'creado_en' (timestamp):** Corresponde a la fecha y hora en que se creó el comentario sobre la sala.

Tabla 'favorito'

- **'id_favorito' (int4):** Corresponde al id único de identificación.
- **'id_usuario' (uuid):** Es el usuario el cual marco a la sala como favorita.
- **'id_sala' (int4):** Es la sala la cual es marcada como favorita.
- **'fecha' (timestamp):** Corresponde a la fecha en la cual se marcó como favorita.

ANEXO 17: ACTIVIDADES PRUEBAS DE USABILIDAD (USUARIO CLIENTE):

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

GUIA DE ACTIVIDADES QUITOJAM ‘USUARIO TIPO CLIENTE’

¡Bienvenido a QuitoJam! El lugar en donde puedes encontrar el espacio perfecto para ti y tu banda en la ciudad de Quito. En QuitoJam puedes conectar con salas de ensayo a tu alcance, descubriendo su disponibilidad, además de los equipos que necesitas para crear música. Dentro de QuitoJam, podrás reservar cualquier sala de ensayo la cual se adapte a tus necesidades, ubicación, precio y disponibilidad.

Actividades:

1. Registrarse e Iniciar Sesión

- **Contexto:** Eres un músico de la ciudad de Quito y estás buscando un espacio para ensayar, por lo que encontraste a QuitoJam. Das clic e ingresas a la pantalla de inicio, pero te diste cuenta de que no te permite ir más allá de ver las salas de ensayo ¿Qué debes hacer?
- **Tareas:**
 - Registrarse.
 - Iniciar sesión con las credenciales proporcionadas.
 - Verifica de lo que es capaz tu perfil.
 - Personaliza tu perfil.

2. Explorar Salas de Ensayo Disponibles

- **Contexto:** Tienes un proyecto musical y necesitas encontrar una sala de ensayo, la cual tenga equipos para grabación. Por lo que, empiezas a buscar un espacio el cual sea adaptable para ti.
- **Tareas:**
 - Aplica filtros de ubicación, precio, y capacidad de personas para encontrar la opción que más se adecue para ti.

- Encuentra y selecciona la opción más adecuada para ti.

3. Realizar una Reserva de Sala

- **Contexto:** Después de observar los detalles, te decides por esa sala que más te convenció, por lo que empiezas con el proceso de realizar una reserva de esa sala.

- **Tareas:**

- Ver los detalles de la Sala, e iniciar con la Reserva
- Seleccionar la fecha y hora(s) de la reserva.
- Confirmar la reserva y pagar por la misma (no es pago real).
- Verifica si te llego una notificación al correo.
- Realiza el mismo proceso, pero para otra reserva.

4. Ver y Gestionar Reservas

- **Contexto:** Después de realizar la reserva, necesitas consultar tus reservas. Además, te das cuenta de que no podrás asistir a una de las reservas que seleccionaste, por lo que decides eliminarla.

- **Tareas:**

- Acceder al historial de reservas.
- Ver detalles de una reserva futura.
- Cancelar una reserva si es necesario.

5. Evaluar la Sala de Ensayo

- **Contexto:** Después de ensayar en una sala de ensayo que reservaste, deseas dejar una reseña para que otros músicos conozcan tu experiencia, de esta manera realizas una recomendación de la sala hacia otros músicos.

- **Tareas:**

- Dejar una calificación y comentarios sobre la calidad de la sala, la atención recibida, como la experiencia en general.

ANEXO 18: ACTIVIDADES PRUEBAS DE USABILIDAD (USUARIO ADMINISTRADOR):

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

GUIA DE ACTIVIDADES QUITOJAM ‘USUARIO TIPO ADMINISTRADOR’

¡Bienvenido a QuitoJam! En nuestra aplicación web hacemos que la gestión, publicación y el alcance de tu sala se convierta en un proceso práctico. Como dueño de tu sala de ensayo, puedes darle visibilidad a tu espacio, y permitir que los músicos de todo Quito pueden encontrar tu sala de ensayo, por lo que podrás recibir reservas de las personas las cuales necesitan ocupar tu espacio, puedes ver las frecuencias de estas reservas, como las notificaciones de estas. Lo que hace de QuitoJam, la herramienta perfecta para maximizar el uso de tu sala, y conectar con la comunidad activa de músicos de la ciudad.

Usuario Administrador (Propietario de Sala)

1. Registrarse como Propietario

- **Contexto:** Ingresas por primera vez en QuitoJam, por lo que observas algunas salas publicadas, sin embargo, quieres empezar tu con la publicación de tu espacio.
- **Tareas:**
 - Completar el formulario de registro.
 - Iniciar sesión con las credenciales de propietario.
 - Observa de lo que es capaz tu perfil.
 - Modifica tu perfil (agregar foto).

2. Agregar una Nueva Sala

- **Contexto:** Tienes disponible 2 nuevas salas de ensayo y quieres añadirlas a la plataforma para que los músicos puedan ver y reservar tu sala. Por lo que necesitas agregar los detalles completos de tu sala. Recuerda que una sala puede contar, con equipos para grabación, con instrumentos, con cancelación de ruido, entre otras cosas.
- **Tareas:**

- Crear por lo menos dos salas en diferentes lugares dentro del área en donde te encuentras. (Una sala solo de grabación y otra solo de ensayos)
- Busca y completar todos los campos necesarios para crear una sala.
- Asigna precios y horarios de disponibilidad de tus salas. Agrega (sábado y domingo en una de esas salas)
- Verifica que tu sala se publicó con toda la información que pusiste.

3. **Consultar Reservas Realizadas**

- **Contexto:** Como eres dueño de varias salas, necesitas revisar todas las reservas realizadas en las últimas semanas. Deseas ver qué músicos han reservado tu espacio durante los últimos días.
- **Tareas:**
 - Acceder al panel de reservas.
 - Ver los detalles de las reservas confirmadas.
 - Observar si tienes algún tipo de cancelación y saber por qué.

4. **Editar Información de la Sala**

- **Contexto:** Quieres actualizar algunos detalles de tu sala, como el precio o agregar un nuevo equipo disponible. Debes editar la información para reflejar estos cambios.
- **Tareas:**
 - Acceder a la sección de edición.
 - Modificar la información de la sala según sea necesario.
 - Guardar los cambios.

5. **Ver Evaluaciones de la Sala**

- **Contexto:** Después de varias reservas, algunos músicos han dejado evaluaciones sobre tu sala. Por lo que, quieres revisar esas calificaciones para ver qué aspectos podrían mejorarse.

- **Tareas:**

- Ver las calificaciones y comentarios en el panel de gestión.

6. Revisar el rendimiento de tus salas

- **Contexto:** Después de un tiempo quieres ver el rendimiento de tus salas de ensayo, por lo que decides observar los días más frecuentes en los cuales recibes mayor demanda.

- **Tareas:**

- Vuelve al panel administrativo y obtén esa información.

ANEXO 19: ACTA DE PARTICIPACIÓN – PRUEBAS DE USABILIDAD.

Pontificia Universidad
Católica del Ecuador
Facultad de Ingeniería
Escuela de Sistemas



ACTA DE PARTICIPACIÓN EN LA PRUEBAS DE USUARIOS 'QUITOJAM'

Fecha: 06/19/2025

Proyecto: QuitoJam

Usted declara a través de su nombre y su firma, que participo dentro de las pruebas de usuario del proyecto 'QuitoJam'. Recuerde que toda información obtenida y su participación realizada dentro de estas pruebas de usuario será solo para fines académicos, a modo de observar el comportamiento en un entorno simulado de casos. De antemano, muchísimas gracias por participar en estas pruebas de usuario. At: **Jonathan Steve Rosero Garcia.**

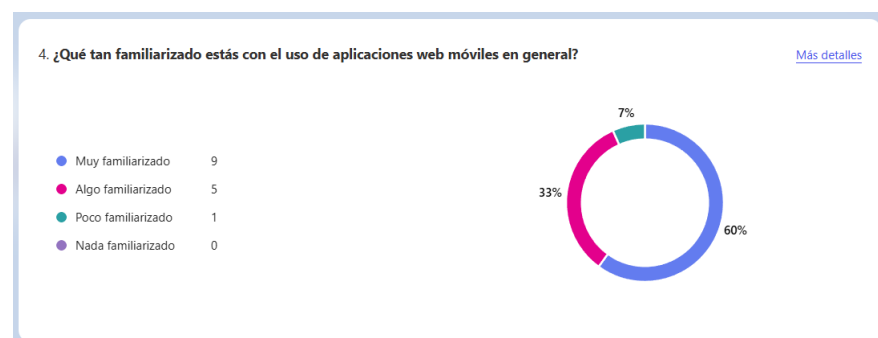
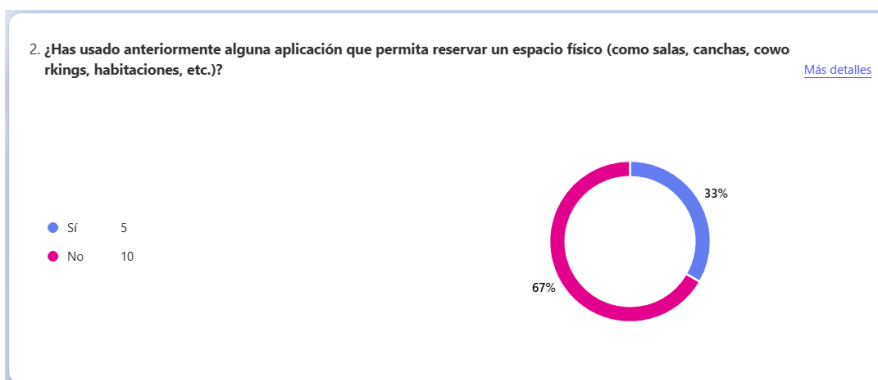
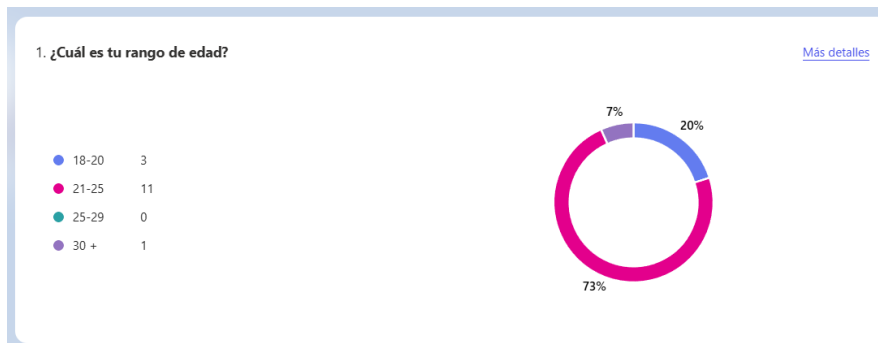
Profesión	Nombre	Firma
Ing. Sonido	Nicolas Zanches	
Composición musical	Domeica Ponguillo	
Productor Musical	Pablo Lincango	
Composición Musical	Geovanny Rivasanpica	
Composición Musical	Tomás Santander	
Estudiante Negocios Internacionales	Zulya Nuñez	



Profesión	Nombre	Firma
Psicología Clínica	Martin Pardo	
Psicología Clínica	Camila Castillo	
Psicología Clínica	Cristina Mena	
Biología Clínica	Rony Mateo Suárez	
Ing Sistemas de Info	Gabriel Carella	
Ingeniería Sistema de Información	María Astudillo	
Ing Sistemas de Información	Michael Rodriguez	
Ing Sistemas de Información	Camila Carrera	
Msc. Pedagogía Musical	Dies Gueim	

ANEXO 20: PRUEBA PRE-TEST:

Las presentes capturas representan los resultados de las pruebas pre test realizadas hacia los usuarios.



5. En una aplicación que te permita reservar un espacio físico, ¿qué funcionalidades esperarías encontrar?

[Más detalles](#)

15
Respuestas

Respuestas más recientes

"Que pueda encontrar lo que necesito."
"Que pueda encontrar una opción que mas se ajuste a mis necesidades o req..."
"Esperaría encontrarme con un espacio que pueda responder a mis necesida..."

...

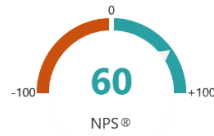
3 encuestados (20%) respondieron reserva para esta pregunta.



6. ¿Qué tan importante es para ti que una aplicación tenga una navegación simple y clara?

[Más detalles](#)

Promotores	10
Pasivos	4
Detractores	1



7. ¿Qué tan importante es para ti que un proceso de reserva sea rápido y sin pasos innecesarios?

[Más detalles](#)

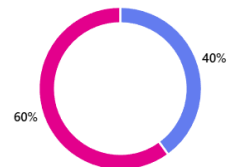
Promotores	10
Pasivos	5
Detractores	0



8. ¿Sueles utilizar filtros o criterios de búsqueda (precio, ubicación, horarios) cuando exploras opciones en una aplicación?

[Más detalles](#)

● Siempre	6
● A menudo	9
● A veces	0
● Rara vez	0
● Nunca	0



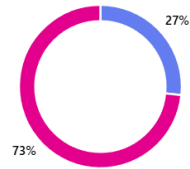
9. **Cómo músico (seas profesional o no), ¿Qué tan fácil te ha resultado el encontrar una sala de ensayo/grabación para ti o tu banda?** [Más detalles](#)

Promotores	3
Pasivos	3
Detractores	9



10. **¿Has encontrado salas de ensayo cerca de tu domicilio? O has tenido que viajar mucho para encontrar una.** [Más detalles](#)

● Si, encontrado salas de ensayo cerca.	4
● No, eh tenido que movilizarme por la ciudad.	11



11. **¿Haz tenidos problemas buscando salas de ensayo? Si es así escribe cuales han sido los más comunes.** [Más detalles](#)

15
Respuestas

Respuestas más recientes

"Últimamente no, porque cuento con mi espacio para poder ensayar, sin em..."

"Que no tengo claro en donde poder buscarlas."

"Si, la ubicación, la falta de detalles de equipos, al principio no saber que opc..."

...

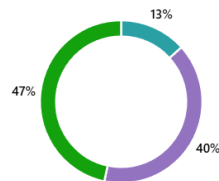
5 encuestados (33%) respondieron sala para esta pregunta.



ANEXO 21: PRUEBA POST-TEST:

1. **¿Qué tan sencilla fue tu experiencia al realizar las tareas asignadas en la aplicación?** [Más detalles](#)

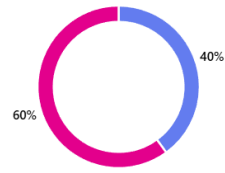
● Muy difícil	0
● Difícil	0
● Regular	2
● Fácil	6
● Muy fácil	7



2. ¿En algún momento sentiste confusión durante el uso de la aplicación?

[Más detalles](#)

- Sí 6
- No 9



3. ¿En qué parte de la aplicación sentiste confusión?

[Más detalles](#)

6
Respuestas

Respuestas más recientes

"Al momento de realizar una búsqueda por palabras el buscador no funciona..."

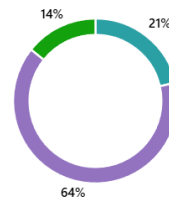
3 encuestados (50%) respondieron horarios para esta pregunta.



4. ¿Cómo describirías la navegación dentro de la aplicación?

[Más detalles](#)

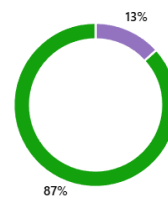
- Muy complicada 0
- Complicada 0
- Regular 3
- Intuitiva 9
- Muy intuitiva 2



5. ¿El diseño visual de la app te pareció adecuado para su propósito?

[Más detalles](#)

- Para nada adecuado 0
- Poco adecuado 0
- Neutral 0
- Algo Adecuado 2
- Muy Adecuado 13



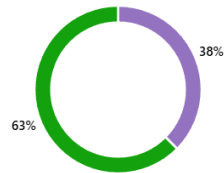
6. ¿Qué tan claro fue el proceso para buscar y reservar una sala de ensayo? (Responde si eres rol cliente) [Más detalles](#)

Promotores	8
Pasivos	0
Detractores	0



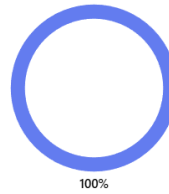
7. ¿El tiempo necesario para completar la reserva te pareció? (Responde si eres rol cliente) [Más detalles](#)

Muy lento	0
Lento	0
Adecuado	0
Rápido	3
Muy rápido	5



8. ¿La confirmación de la reserva por email fue clara y útil? (Aplica para clientes y administradores) [Más detalles](#)

Si	14
No	0



9. ¿Qué le faltó al email?

0
Respuestas

0 respuestas enviada



10. ¿La información mostrada sobre cada sala fue suficiente para tomar una decisión? (Responde si eres rol cliente) [Más detalles](#)

Promotores	6
Pasivos	0
Detractores	2



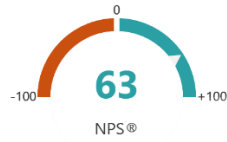
11. ¿Cómo calificarías la información que se presenta en tu historial de reservas? (Responde si eres rol cliente) [Más detalles](#)

Promotores	5
Pasivos	1
Detractores	2



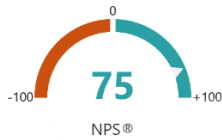
12. ¿Encontraste útil los filtros de búsqueda? (Responde si eres rol cliente) [Más detalles](#)

Promotores	5
Pasivos	3
Detractores	0



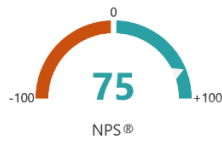
13. ¿Cómo calificarías el uso de geocalizadores para la búsqueda de salas de ensayo cercanas a ti? (Responde si eres rol cliente) [Más detalles](#)

Promotores	6
Pasivos	2
Detractores	0



14. ¿Cómo calificarías tu experiencia general con el proceso de reserva y uso de la app? (Responde si eres rol cliente) [Más detalles](#)

Promotores	6
Pasivos	2
Detractores	0



15. ¿Cómo calificarías la facilidad para poder crear, editar una sala de ensayo? (Responde si eres rol Administrador) [Más detalles](#)

Promotores	4
Pasivos	0
Detractores	2



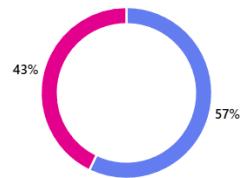
16. ¿Qué tan fácil fue gestionar la disponibilidad de tus salas para fechas específicas? (Responde si eres rol Administrador) [Más detalles](#)

Promotores	2
Pasivos	0
Detractores	4



17. ¿Pudiste actualizar las fechas de manera rápida y sin problemas? (Responde si eres rol Administrador) [Más detalles](#)

● Si	4
● No	3



18. ¿Porque no pudiste? [Más detalles](#)

3
Respuestas

Respuestas más recientes

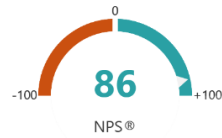
"Porque tengo que borrar la fecha, para editar una nueva."
...

1 encuestados (33%) respondieron fecha para esta pregunta.

horario
boton acción fecha información
sala momento

19. ¿Qué tan útil fue el panel para ver detalles de las reservas pasadas, cancelaciones y cambios? (Responde si eres rol Administrador) [Más detalles](#)

Promotores	6
Pasivos	1
Detractores	0



20. ¿Qué tan claro fue de interpretar, la información sobre el rendimiento de tus salas, como la demanda y ocupación? **(Responde si eres rol Administrador)** [Más detalles](#)



21. ¿Qué tan de acuerdo@ estas con descontarte el 5% del precio total de la reserva, por uso de la app para publicar y publicitar tu sala de ensayo? **(Responde si eres rol Administrador)** [Más detalles](#)



22. ¿Porque estarías de acuerdo o en desacuerdo en cobrarte una tarifa del 5% por uso de la aplicación? **(Responde si eres rol Administrador)** [Más detalles](#)

7
Respuestas

Respuestas más recientes
"No refleja un valor alto."
...

3 encuestados (43%) respondieron aplicación para esta pregunta.



23. ¿Qué funcionalidades adicionales consideras que serían útiles para mejorar la gestión de tus salas en la aplicación? **(Responde si eres rol Administrador)** [Más detalles](#)

7
Respuestas

Respuestas más recientes
"Mejorar el tema para editar los horarios, sin necesidad de borrarlos."
...

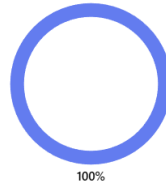
3 encuestados (43%) respondieron horarios para esta pregunta.



24. ¿Recomendarías QuitoJam ?

[Más detalles](#)

● Si 15
● No 0



26. ¿Alguna opinión o comentario extra sobre el desarrollo de este proyecto?

[Más detalles](#)

14
Respuestas

Respuestas más recientes

"Es una aplicación, que le veo futuro"
"Mejorar la presentación de los detalles de las salas"
"Mejorar la búsqueda por texto, excelente proyecto"
...

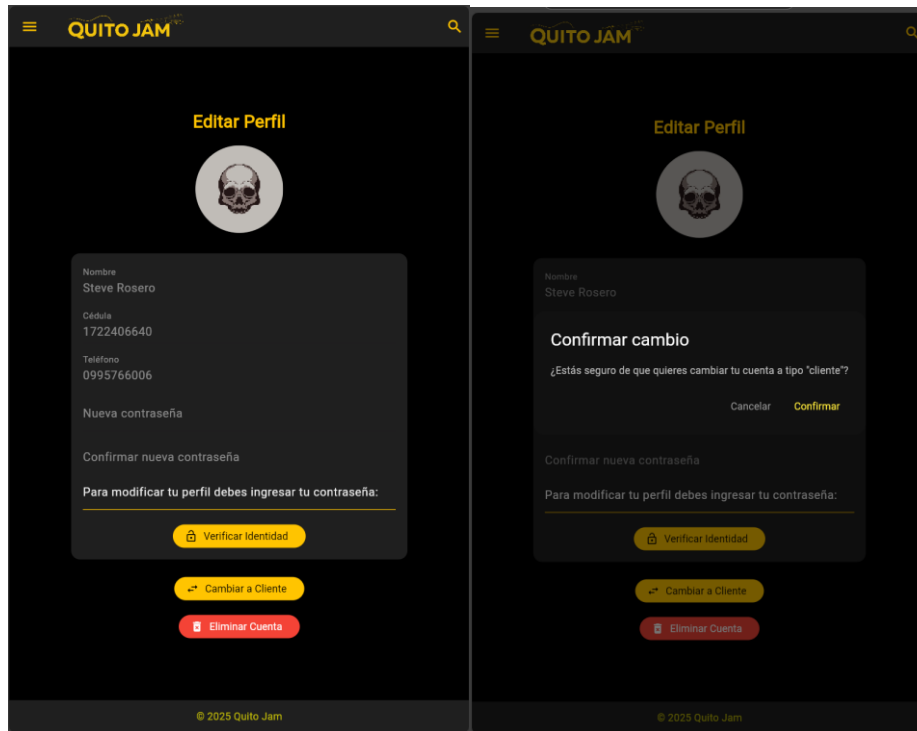
4 encuestados (29%) respondieron proyecto para esta pregunta.



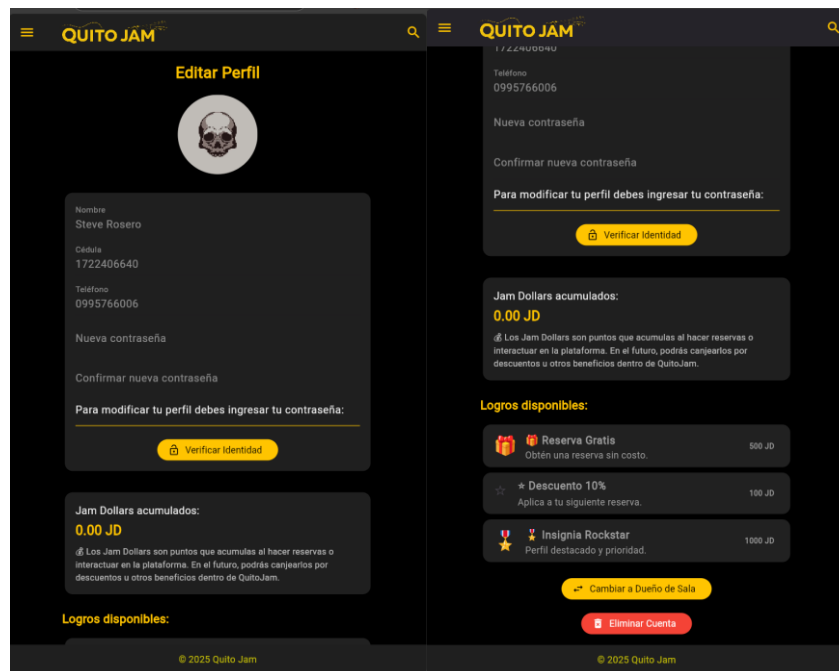
ANEXO 22: RESULTADOS FINALES PROTOTIPO

Botón para cambiar de rol sin perder la información del usuario.

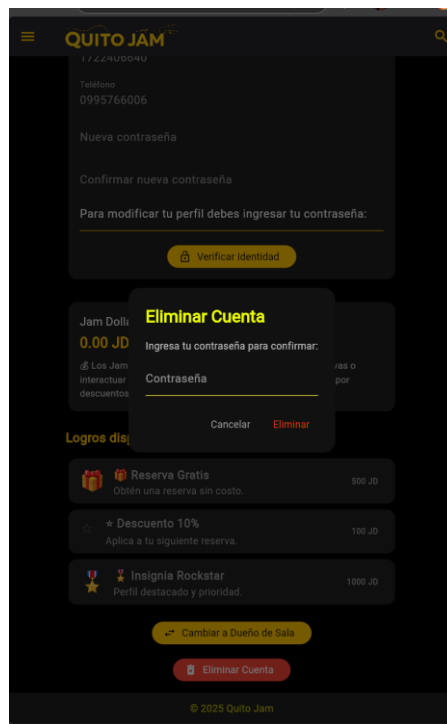
Pantalla 'Mi_Perfil' (Admin):



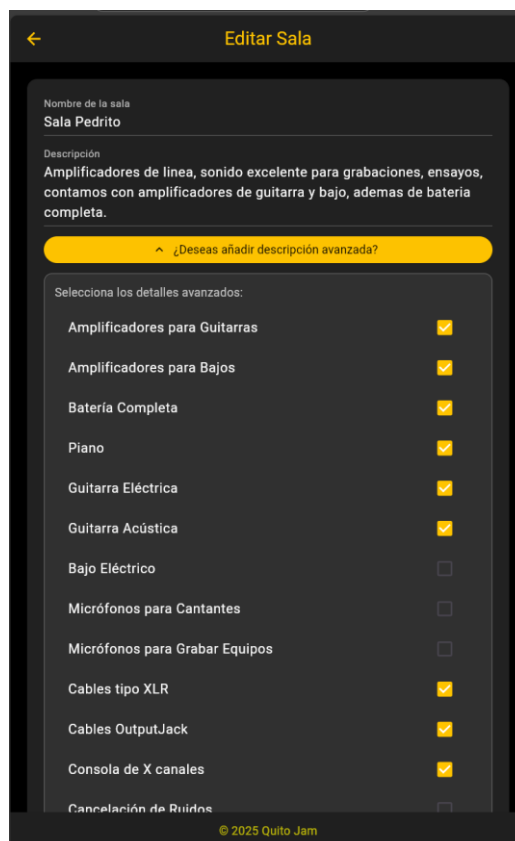
Pantalla 'Mi_Perfil' (Cliente)



Acción al presionar el botón de eliminación:

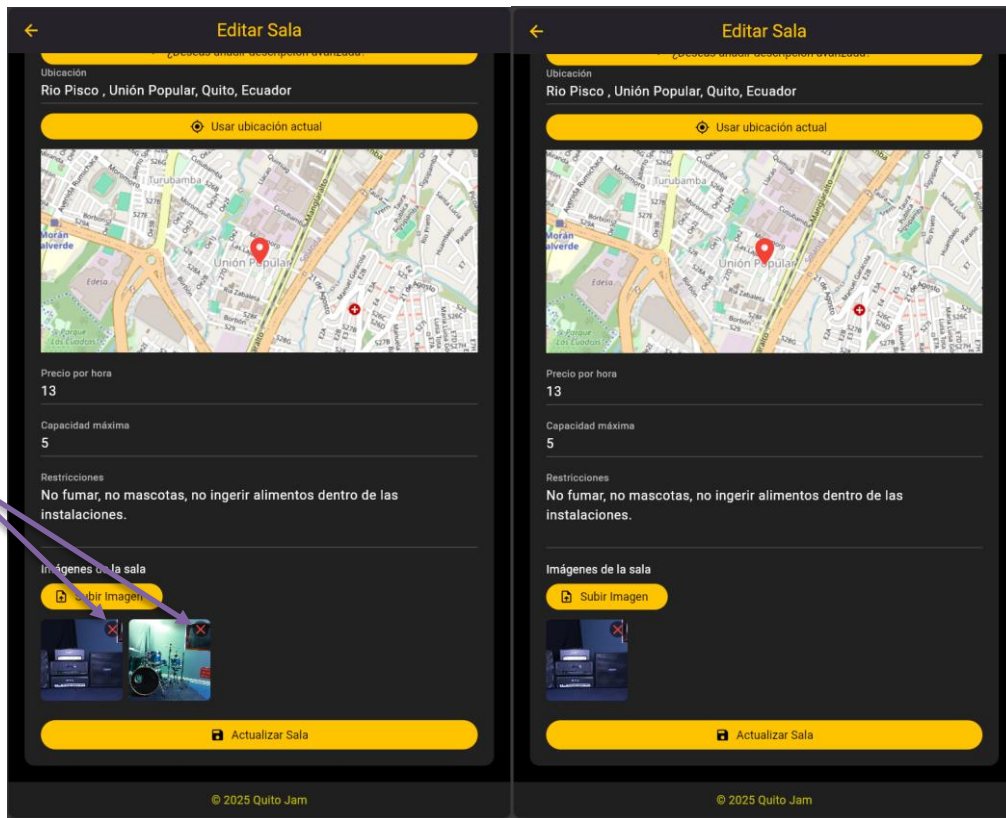


Lista de detalles sobre equipos específicos:

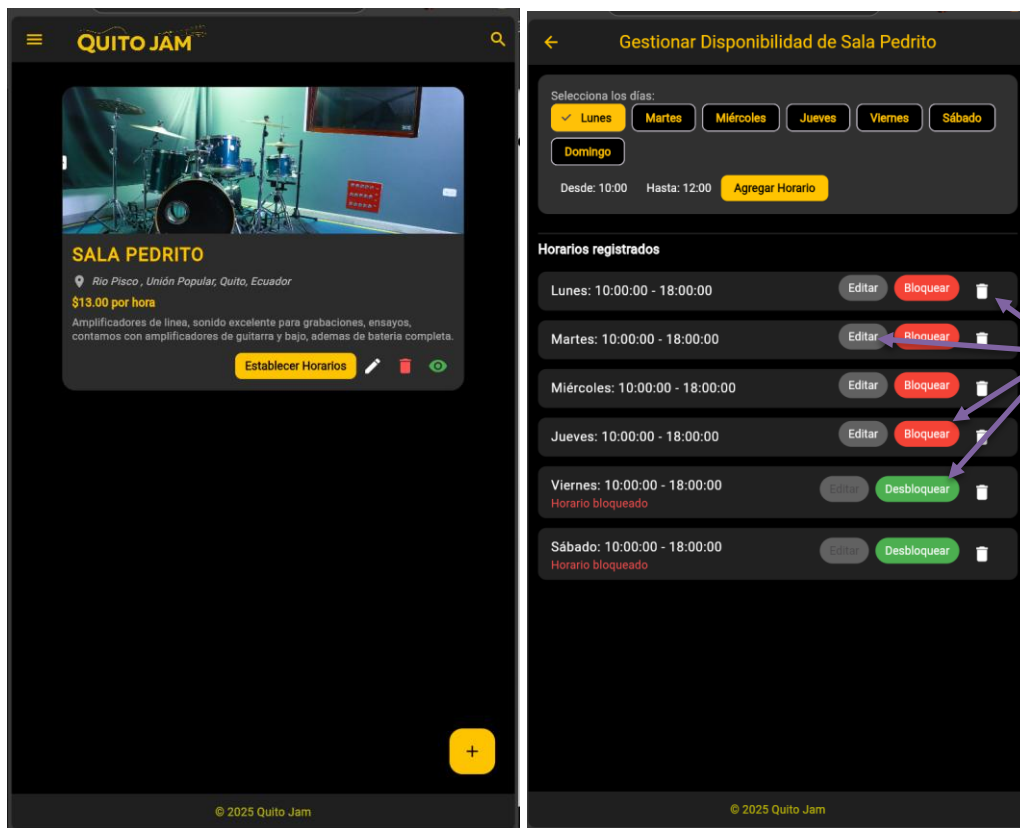


Botón para eliminar imágenes:

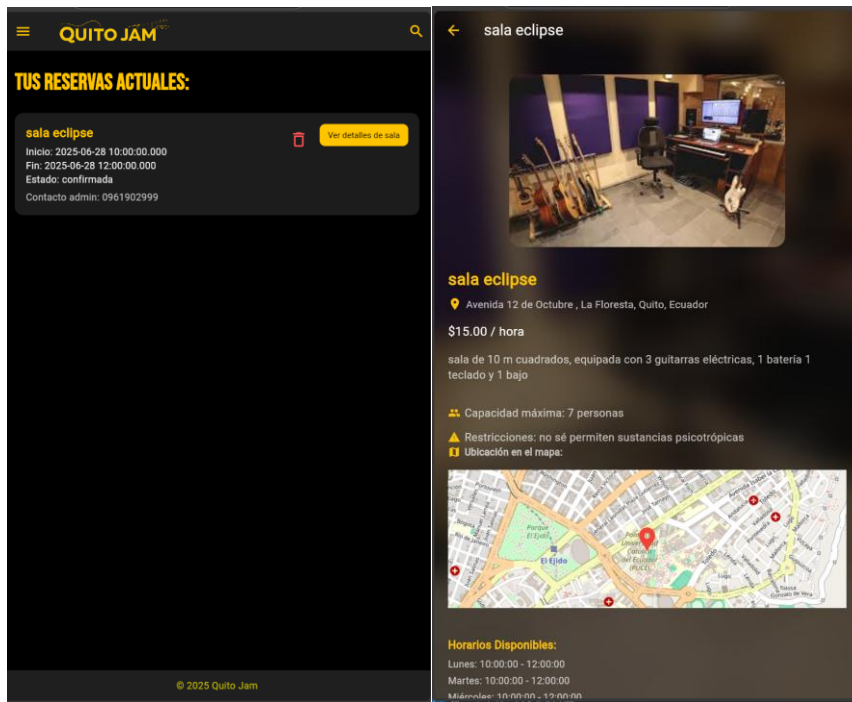
Botón para eliminar imagen.



UI Mejorada en la pantalla de Gestión de Horarios:



Botón para acceder a los detalles de la sala desde ‘Mis Reservas’:



The image shows a mobile application interface for 'QUITO JAM'. On the left, under 'TUS RESERVAS ACTUALES:', there is a reservation card for 'sala eclipse'. The card displays the start and end times (2025-06-28 10:00:00.000 to 12:00:00.000), the status 'confirmada', and the contact number '0961902999'. A yellow button labeled 'Ver detalles de sala' is positioned to the right of the reservation details.

The right side of the image shows the details page for 'sala eclipse'. It features a photo of the room, the address 'Avenida 12 de Octubre, La Floresta, Quito, Ecuador', the price '\$15.00 / hora', and a description: 'sala de 10 m cuadrados, equipada con 3 guitarras eléctricas, 1 batería 1 teclado y 1 bajo'. It also lists 'Capacidad máxima: 7 personas' and 'Restricciones: no se permiten sustancias psicotrópicas'. A map shows the location in Quito. At the bottom, the 'Horarios Disponibles' are listed: 'Lunes: 10:00:00 - 12:00:00', 'Martes: 10:00:00 - 12:00:00', and 'Miércoles: 10:00:00 - 12:00:00'. A copyright notice '© 2025 Quito Jam' is visible at the bottom left.