

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE ELECTRÓNICA E INFORMÁTICA
CONCEPCIÓN - CHILE



**“BEATMAP: PLAN DE GESTIÓN DE PROYECTO DE UNA PLATAFORMA GEOLOCALIZADA DE
EVENTOS PARA ARTISTAS EMERGENTES”**

IGNACIO ESTEBAN HORMAZAVAL HORMAZÁBAL

**MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO EN INFORMÁTICA**

Profesor Guía: Cristian Antonio Lara Valenzuela

Enero - 2026



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): Memoria o trabajo de título; Tesis de Postgrado;

Título del trabajo: BEATMAP: PLAN GESTIÓN DE PROYECTO DE UNA PLATAFORMA GEOLOCALIZADA DE EVENTOS PARA ARTISTAS EMERGENTES

Nombre del candidato(a): Ignacio Esteban Hormazaval Hormazábal

Carrera / Grado: Ingeniería en informática

Campus: Concepcion ; Departamento: Electronica e Informatica

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Cristian Antonio Lara Valenzuela, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL

El trabajo **NO contiene información que amerite confidencialidad** y puede ser publicado de inmediato en repositorio con acceso abierto.

El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (embargo) por:

6 meses; 12 meses; 2 años; 3 años; 5 años; 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 16/01/2025 ; Firma: 

Estudiante o Candidato(a):

Fecha: 16/01/2025 ; Firma: 

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

DEDICATORIA

Esta memoria se dedica a la familia, por el apoyo constante y por sostener el esfuerzo cuando el proyecto exigió tiempo, disciplina y paciencia. El acompañamiento cercano y la comprensión frente a las exigencias académicas y personales fueron fundamentales para completar este proceso.

De manera especial, esta memoria se dedica a la memoria de **Hilda Nely Hormazábal Fernández**, quien en paz descanse. Su confianza, sus palabras oportunas y su insistencia en no abandonar la carrera marcaron profundamente este camino. Fue ella quien alentó a continuar cuando surgieron dudas y dificultades, reforzando la convicción de seguir adelante y perseguir los propios sueños. Su apoyo permanece como un pilar significativo de este logro.

Asimismo, se agradece profundamente mis padres, por el respaldo permanente, el esfuerzo sostenido y la confianza depositada a lo largo de toda la formación universitaria. Su apoyo fue clave para enfrentar las distintas etapas del proceso académico y personal que implicó el desarrollo de esta memoria.

De igual forma, se dedica este trabajo a los hermanos **Mauricio Hormazábal** y **Fabián Hormazábal**, por el acompañamiento constante, la preocupación genuina y el apoyo entregado durante los años de estudio. Su presencia y ánimo contribuyeron a mantener la motivación en los momentos más exigentes del proceso.

Finalmente, se dedica también a quienes acompañaron durante la etapa universitaria y ayudaron a mantener el rumbo cuando el trabajo académico y el desarrollo del proyecto se volvieron especialmente demandantes, aportando apoyo, comprensión y motivación para llegar a término.

AGRADECIMIENTOS

De mi parte, sería injusto no reconocer a todas las personas que hicieron posible no solo el desarrollo de esta memoria, sino también que mi paso por la universidad se transformara en una etapa de aprendizaje, crecimiento y experiencias que marcaron profundamente mi formación personal y profesional.

En primer lugar, quiero agradecer a mi familia por el apoyo constante, la paciencia y la comprensión a lo largo de todo este proceso. Su respaldo emocional fue fundamental para sostener el esfuerzo, la constancia y la motivación necesarios durante las distintas etapas del proyecto y de mi formación universitaria.

Quiero expresar también mi sincero agradecimiento al profesor guía Cristian Antonio Lara Valenzuela, por su acompañamiento académico, su disposición permanente al diálogo y por las observaciones metodológicas que permitieron mantener coherencia entre los objetivos planteados, la evidencia recopilada y los resultados obtenidos. Asimismo, agradezco al profesor José Luis Carrasco por sus comentarios y retroalimentación, los cuales contribuyeron a mejorar la claridad de la propuesta y a fortalecer su justificación desde la perspectiva de la Ingeniería en Informática.

Agradezco especialmente al equipo BeatMap por el trabajo colaborativo, la confianza y la disposición constante a revisarnos entre pares: Tomás Solar, Fernando Cruces, Dismar Quiroz, Felipe Badilla y Javier Choque. Más allá de los resultados técnicos, valoro profundamente la constancia en las reuniones, la responsabilidad al asumir tareas y, sobre todo, la apertura a cuestionar y reformular la idea inicial cuando el proceso así lo exigió. El trabajo en equipo fue clave para transformar una idea en una propuesta más sólida, realista y validada.

Finalmente, quiero agradecer a todas las personas que participaron en las entrevistas y en las instancias de validación en terreno, por compartir su tiempo y experiencias de manera honesta. Sus testimonios permitieron identificar problemáticas reales y convertirlas en requerimientos concretos, dando sentido práctico al desarrollo de la plataforma y reforzando el aprendizaje obtenido durante todo el proyecto.

RESUMEN

Esta memoria documenta el proceso de gestión de producto y coordinación del desarrollo de BeatMap, una plataforma orientada a centralizar y visibilizar eventos de artistas emergentes mediante geolocalización. El proyecto se desarrolló en un contexto donde la información sobre eventos suele estar fragmentada entre redes sociales, afiches y mensajes privados, generando fricción tanto para quienes buscan panoramas como para quienes producen y difunden eventos.

Desde la perspectiva de Product Management, la gestión del producto lideró la definición de alcance, la estructuración de objetivos, la planificación por etapas y la priorización del Producto Mínimo Viable (MVP). El propósito operativo fue alcanzar un MVP funcional durante el primer semestre de 2025 y cerrar el año con un incremento validado y optimizado para su presentación en instancias académicas y de difusión.

El proceso de validación se realizó en dos etapas. La primera se basó en entrevistas semiestructuradas y observación en terreno, incluyendo participación en eventos de artistas emergentes para levantar testimonios y patrones de comportamiento. La segunda se enfocó en test de usabilidad con pauta y registro de notas, buscando detectar fricciones en tareas críticas como descubrir eventos, entender la información y navegar por el mapa.

Los hallazgos se tradujeron en módulos del sistema y en un *backlog* gestionado en Jira. Se definieron casos de uso, requerimientos funcionales y no funcionales, historias de usuario, criterios de aceptación y reglas explícitas para decidir qué entraba y qué quedaba fuera del MVP en cada iteración.

La ejecución se apoyó en una cadencia de reuniones recurrentes, seguimiento administrativo con Google Calendar y un flujo de integración basado en *Pull Requests* y revisión entre pares en GitHub. El control de calidad se abordó como práctica transversal, evitando una etapa separada de QA: cuando una entrega no cumplía criterios acordados, se documentaban observaciones en el *Pull Request* y la integración se postergaba hasta resolverlas.

Como resultado, BeatMap evolucionó desde una primera idea amplia hacia una propuesta reformulada, más sólida y realizable por etapas. La memoria presenta el marco conceptual utilizado, la propuesta de solución, una explicación introductoria del modelo de negocio y la evidencia de validación que sustentó decisiones claves del producto.

Palabras Clave— *Product Management*; desarrollo ágil; MVP; geolocalización; validación en terreno; eventos; artistas emergentes.

ABSTRACT

Abstract—*This report documents the product management process and development coordination of BeatMap, a platform designed to centralize and promote emerging artists' events through geolocation. The project addresses a context where event information is fragmented across social networks and informal channels, generating friction for both attendees and event promoters.*

From a Product Management perspective, the author led scope definition, objective setting, staged planning and MVP prioritization. The operational goal was to reach a functional MVP during the first semester of 2025 and close the year with a validated and optimized increment for academic and dissemination instances.

Validation was executed in two stages. The first stage relied on semi-structured interviews and field observation, including visits to emerging-artist events to capture real testimonies and patterns. The second stage focused on usability tests with a structured checklist and notes, aiming to detect frictions in critical tasks such as discovering events, understanding information and navigating the map.

Findings were translated into system modules and a Jira-managed backlog. Use cases, functional and non-functional requirements, user stories, acceptance criteria and explicit rules to decide what enters or stays out of the MVP in each iteration were defined.

Execution relied on recurring meetings, administrative tracking through Google Calendar, and an integration workflow based on Pull Requests and peer code reviews on GitHub. Quality control was handled as a cross-functional practice, without a separate QA stage.

Keywords— *Product Management; agile development; MVP; geolocation; field validation; events; emerging artists.*

GLOSARIO

API: Application Programming Interface; interfaz para comunicación entre sistemas.

BMC: Business Model Canvas; herramienta para describir el modelo de negocio por bloques.

CI/CD: Integración continua y despliegue continuo.

CRUD: Create, Read, Update, Delete; operaciones básicas sobre datos.

CSAT: Customer Satisfaction Score; indicador de satisfacción del usuario.

DoD: Definition of Done; criterios mínimos para considerar una tarea terminada.

Git: Sistema de control de versiones.

HTTP/HTTPS: Protocolo web (seguro en HTTPS).

ICS: Formato estándar de calendarios (iCalendar).

Jira: Gestión de backlog, sprints y seguimiento.

JWT: JSON Web Token; autenticación basada en tokens.

KPI/ Key Performance Indicator: indicador clave de desempeño.

MVP/ Minimum Viable Product: versión mínima para validar hipótesis.

PR/ Pull Request: solicitud de integración de cambios en GitHub.

REST: Estilo de arquitectura para APIs basado en recursos y métodos HTTP.

RICE: Reach, Impact, Confidence, Effort; marco de priorización.

UI/UX: User Interface / User Experience.

URL: Dirección de un recurso en internet.

UTFSM: Universidad Técnica Federico Santa María.

WIP/ Work in Progress; trabajo en curso.

Backlog: Lista priorizada de priorizada de trabajo pendiente del producto y necesidades del producto (tareas, mejoras, bugs)

Criterios de aceptación: Condiciones verificables para aceptar una historia/tarea como cumplida.

Branch (rama): Línea de trabajo y desarrollo aislada en Git para desarrollar una funcionalidad sin afectar la rama principal.

Commit: Registro de cambios en el repositorio.

Revisión de código: Evaluación por pares para asegurar coherencia, mantenibilidad y cumplimiento de criterios.

Comentario de revisión: Observación registrada en el PR para solicitar correcciones antes de integrar.

Merge: Integración de una rama a otra (por ejemplo, hacia la rama principal).

Épica: Conjunto de historias/tareas agrupadas por un objetivo mayor.

Historia de usuario: Requisito descrito desde la perspectiva del usuario y su necesidad.

Iteración: Ciclo corto de trabajo con entregables revisables.

Sprint: Iteración acotada en tiempo con objetivo definido.

Hito: Punto del plan asociado a una entrega o fecha clave.

Cadencia de reuniones: Ritmo acordado de coordinación para sostener alineamiento y seguimiento.

Daily/Dailies: Reunión breve de coordinación para objetivos del día y bloqueos.

Seguimiento (día por medio): Reunión para revisar avance, ajustar prioridades y remover bloqueos.

Stakeholders: Personas o entidades afectadas por el producto o con influencia sobre su desarrollo.

Hipótesis: Suposición verificable que guía decisiones de producto.

Validación en terreno: Contraste de hipótesis con evidencia real (entrevistas/observación/pruebas).

Geolocalización: Uso de ubicación para filtrar o recomendar información según cercanía.

Tracker (mapa): Módulo que muestra eventos en mapa con filtros para descubrimiento.

Reformulación: Redefinición del enfoque del proyecto basada en evidencia.

Plan B/C/D: Alternativas definidas para reducir desviaciones de plazo o alcance ante riesgos.

Branch (rama): Línea de trabajo aislada en Git para desarrollar una funcionalidad sin afectar la rama principal.

Definition of Done (DoD): Umbral mínimo acordado para considerar un incremento como terminado y listo para revisión/entrega.

ICE/RICE: Técnicas de priorización basadas en impacto, confianza, esfuerzo (ICE) y alcance, impacto, confianza, esfuerzo (RICE).

Lean Startup: Enfoque de desarrollo basado en hipótesis, experimentos y aprendizaje validado.

Pull Request (PR): Solicitud de integración de cambios desde una rama hacia otra, sujeta a revisión.

Wireframe: Boceto de baja fidelidad que representa estructura y flujo de una interfaz.

Excalidraw: Herramienta utilizada para iterar wireframes y flujos de manera rápida.

INDICE DE CONTENIDOS

ABSTRACT	6
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA	12
CAPÍTULO 2: MARCO CONCEPTUAL.....	20
CAPÍTULO 3: PROPUESTA DE SOLUCION	27
CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN	39
CAPÍTULO 5: CONCLUSIONES	63
REFERENCIAS BIBLIOGRÁFICAS.....	65
ANEXOS	66

INDICE DE FIGURAS

Figura 1: Evolución de la idea y reformulaciones BeatMap de servicios múltiples a tres actores.....	14
Figura 2: Arquitectura General Del Proyecto	37
Figura 3: Extracto de calendario de reuniones y hitos del proyecto (reuniones día por medio y validaciones).....	43
Figura 4: Flujo de trabajo de integración continua (Branching Model).....	44
Figura 5: Ejemplos de publicaciones y material audiovisual de difusión (Instagram/LinkedIn) vinculados a hitos del proyecto.	49
Figura 6: Sprints/hitos (ubicación sugerida para incorporar la visualización temporal del proyecto).	58
Figura 7: Evidencias de trabajo colaborativo: ejemplo de Pull Request comentado y ajustes aplicados antes de integrar cambios.	62

INDICE DE TABLAS

Tabla 1: Evolución de la propuesta BeatMap (dos reformulaciones).	13
Tabla 2: Relación entre objetivos específicos, marco conceptual y propuesta.	17
Tabla 3: Alcance del MVP (definición inicial)	18
Tabla 4: Marco de gestión del proyecto	19
Tabla 5: Mapeo de prácticas del proyecto a áreas de marco de gestión del proyecto.....	21
Tabla 6: Conceptos Lean Startup utilizados.	22
Tabla 7: Comparación resumida de enfoques de trabajo utilizados como referencia. Fuente: Elaboración propia.	23
Tabla 8: Comparación de enfoques de gestión	24
Tabla 9: Ventajas y desventajas de aplicar las Metodologías Ágiles en Beatmap.	26
Tabla 10: Segmentos y propuesta de valor	27
Tabla 11: Business Model Canvas (BMC) resumido.....	28
Tabla 12: Definiciones económicas utilizadas en la estimación temprana Fuente: Elaboración propia.	29
Tabla 13: Comparación con alternativas existentes.....	30
Tabla 14: Síntesis de requerimientos RF/RNF priorizados para el MVP.	31
Tabla 15: Criterios utilizados para decidir si una historia entra o queda fuera del MVP. ...	34
Tabla 16: Herramientas de gestión y su propósito.....	41
Tabla 17: Cadencia de coordinación utilizada.	42
Tabla 18: Definition of Done (DoD) técnico-funcional.....	44
Tabla 19: Plan de difusión y señales buscadas.	48
Tabla 20: Tarjetas de hipótesis (resumen).	50
Tabla 21: Ejemplos de problemas levantados en entrevistas y su traducción a módulos del MVP.....	51
Tabla 22: Resumen de hallazgos del test y acción tomada.....	53
Tabla 23: Notas CSAT registradas (n=20)	54
Tabla 24: Síntesis cuantitativa (CSAT)	55
Tabla 25: Hallazgo → Decisión →	55
Tabla 26: Resumen de validaciones planificadas (según planificación)	57
Tabla 27: Resumen de sprints e incrementos (plantilla).	58
Tabla 28: Resumen de los sprints relevantes en Beatmap	59

INTRODUCCIÓN

La presente memoria aborda la gestión de producto y el proceso de validación en terreno de la plataforma digital BeatMap, orientada al ecosistema de eventos y artistas emergentes a nivel local. En dicho contexto, la difusión de tocatas y actividades culturales opera de manera fragmentada a través de redes sociales y canales informales, lo que dificulta el descubrimiento oportuno de eventos y limita la coordinación entre artistas, organizadores y público.

Esta fragmentación informativa no solo reduce la visibilidad de eventos existentes, sino que también impide registrar aprendizajes, medir impacto y sostener continuidad operativa. Frente a este escenario, BeatMap se plantea como una solución digital enfocada en centralizar el descubrimiento de eventos mediante geolocalización, estructurando la interacción entre los actores del ecosistema y reduciendo la fricción asociada a la búsqueda y difusión de actividades culturales.

El proyecto se aborda desde una perspectiva de **gestión de producto**, más que como la mera implementación de un sistema informático. En este sentido, el foco se sitúa en la definición de alcance, la priorización por valor y la toma de decisiones fundamentadas en evidencia. La propuesta inicial del proyecto, caracterizada por un alcance amplio, fue reformulada tras las primeras instancias de validación y demostración, dando paso a un enfoque más acotado, realizable y alineado con necesidades reales del dominio.

La metodología de trabajo integró principios de Lean Startup, prácticas ágiles y herramientas de gestión para sostener una ejecución iterativa y controlada. La validación se estructuró en dos etapas: una fase exploratoria basada en entrevistas y observación en terreno, y una fase evaluativa mediante pruebas de usabilidad. Ambas influyeron directamente en la evolución del producto y en la priorización del MVP.

En conjunto, la presente memoria documenta el proceso de gestión, reformulación y validación de BeatMap, destacando cómo la investigación aplicada y la conducción disciplinada del producto permiten reducir incertidumbre y construir una solución tecnológica coherente con el contexto cultural y organizacional en el que se inserta.

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA

1.1 Contexto y motivación

La escena musical emergente de Concepción presenta una actividad constante de tocatas, presentaciones y eventos de bajo y mediano presupuesto. Sin embargo, la comunicación hacia audiencias potenciales se dispersa entre plataformas y formatos de publicación de corta vida. Con ese contexto, la motivación principal del proyecto BeatMap es reducir la pérdida de información y facilitar el descubrimiento oportuno de eventos, a la vez que se habilita un camino gradual hacia trazabilidad y profesionalización de la gestión.

Un elemento organizacional relevante del proyecto fue la reformulación de la idea inicial. Si bien la primera iteración era atractiva a nivel conceptual, no resultó suficientemente apropiada al contrastarla con necesidades observadas en el entorno. A partir de la reformulación, BeatMap se consolidó como un proyecto ejecutable por etapas, con un alcance gradual que permitió pasar de hipótesis a entregables con evidencia de validación, priorizando un MVP enfocado en descubrimiento y claridad de la información.

1.1.1 Evolución de la idea BeatMap y reformulación del producto

BeatMap no se desarrolló como una idea estática. Durante el semestre, la propuesta atravesó dos cambios relevantes de enfoque, motivados por evidencia recogida en conversaciones con actores del ecosistema y por la necesidad de transformar una idea potente, pero demasiado amplia, en un proyecto ejecutable con recursos acotados. En términos de gestión de producto, estos cambios se entendieron como reformulaciones controladas: se ajustó el problema central, se redefinieron usuarios objetivos y se acotó el MVP para maximizar aprendizaje y viabilidad técnica.

La primera formulación priorizaba la agregación general de eventos, asumiendo que bastaba con reunir información dispersa. Sin embargo, la indagación inicial mostró que el mayor dolor no era solo falta de información, sino la ausencia de una forma simple de descubrir eventos relevantes a tiempo, con contexto suficiente y con una estructura que permitiera mantener los datos actualizados sin depender exclusivamente de publicaciones sueltas en redes sociales.

La segunda reformulación consolidó el producto en torno a dos actores principales: artistas y organizadores. Se observó que ambos comparten el problema de visibilidad y coordinación, pero con necesidades distintas. Los artistas requieren exposición y conexión con audiencias; los organizadores necesitan publicar eventos, comunicar cambios y administrar información de manera consistente. A partir de esto, BeatMap se diseñó para conectar usuarios, negocio y tecnología, articulando módulos que soportaran publicación, descubrimiento geolocalizado y validación de información.

Iteración	Enfoque / Idea central	Usuarios objetivo	Hallazgo gatillante	Decisión de producto
V0 (idea inicial)	Mapa general de eventos; centralización de información dispersa	Público general	Información existe pero se pierde; falta contexto y oportunidad	Definir validación temprana y acotar hipótesis
V1 (reformulación 1)	Descubrimiento geolocalizado con foco en claridad de evento	Oyentes + artistas emergentes	Dolor principal: descubrir a tiempo y entender el evento rápido	Definir módulos de descubrimiento, detalle de evento y perfiles
V2 (reformulación 2)	Plataforma conectando artistas y organizadores con oyentes	Artistas + organizadores + oyentes	Necesidad de publicación mantenible y coordinación; evitar desactualización	Incluir flujo de publicación/gestión de eventos + reglas mínimas de calidad y trazabilidad

Tabla 1: Evolución de la propuesta BeatMap (dos reformulaciones).

Fuente: elaboración propia.

En una primera aproximación, la propuesta se concibió como una landing page orientada a concentrar distintos servicios en un mismo lugar: un feed de publicaciones, un espacio de “jobs” y oportunidades, un marketplace y un listado de eventos. Esta visión inicial buscaba maximizar cobertura funcional, pero generó un alcance amplio y difícil de validar en un plazo acotado. La primera demostración interna evidenció que, aunque la idea era atractiva, la mezcla de servicios no respondía con precisión a un problema único y verificable, lo que aumentaba el riesgo de construir características que no serían utilizadas.

Tras esa primera demostración, se decidió iniciar un proceso de reformulación guiado por evidencia. El foco dejó de ser “agregar módulos” y pasó a ser “entender a quién se está ayudando y por qué”. Esto implicó conversar con actores reales del entorno musical local y observar dinámicas en terreno (eventos de artistas emergentes, organizadores y público), con el objetivo de identificar fricciones concretas: dificultad para enterarse de eventos en tiempo real, falta de canales simples para postular a escenarios, y ausencia de mecanismos para publicar convocatorias y trabajos asociados a la producción de eventos.

Como resultado, BeatMap evolucionó hacia una segunda formulación centrada en tres actores principales: artista, organizador y oyente. Desde esa estructura se redefinieron funcionalidades clave, priorizando un núcleo operativo más coherente: (i) un *tracker* de eventos geolocalizados en tiempo real, (ii) postulación de artistas a eventos publicados por organizadores y (iii) publicación de trabajos/convocatorias por parte de organizadores. En esta memoria, esta transición se entiende como un ajuste de producto por etapas, donde el valor se consolida al reducir ambigüedad y aumentar trazabilidad entre necesidad, módulo y evidencia de validación.

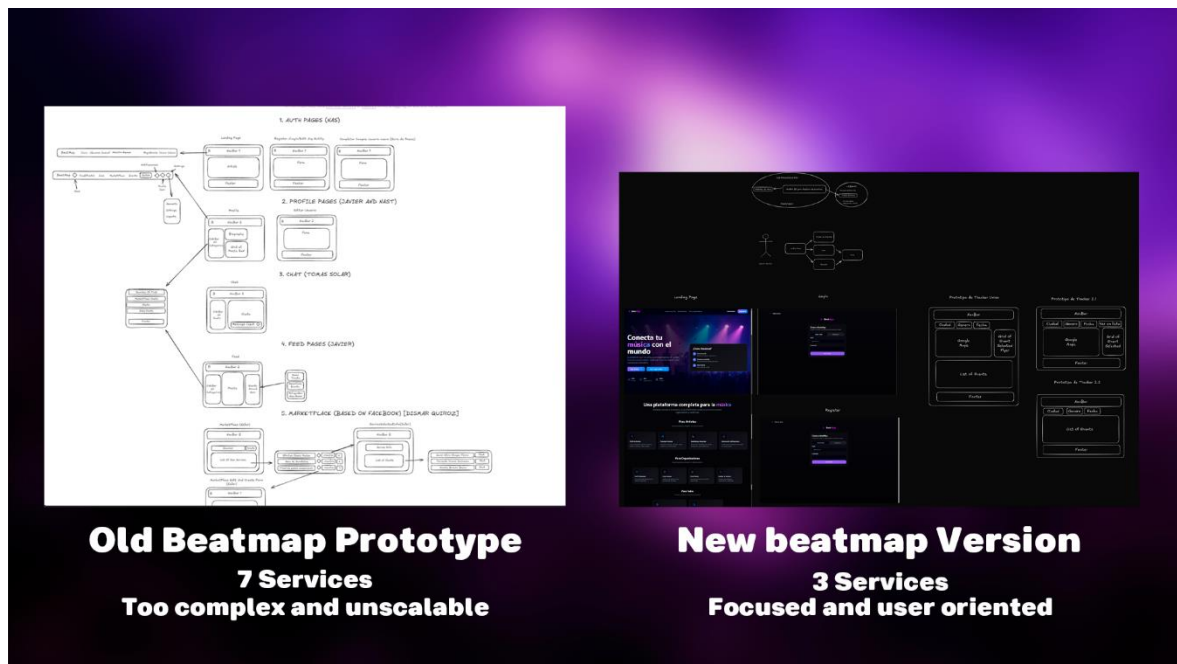


Figura 1: Evolución de la idea y reformulaciones BeatMap de servicios múltiples a tres actores.

Fuente: Documentación Beatmap.

Un elemento que condicionó fuertemente la toma de decisiones fue la ausencia de financiamiento dedicado. Las salidas a terreno, traslados a eventos, generación de material audiovisual, gráficas, branding y mantención de la página web debieron ser costeados por el propio equipo. Esta restricción reforzó la necesidad de priorizar: cada incremento debía justificar su valor en términos de aprendizaje y cercanía a un MVP demostrable para instancias académicas y de difusión (por ejemplo, presentaciones y ferias), evitando inversiones de tiempo en funcionalidades que no aportaran evidencia nueva.

1.2 Formulación del problema

El problema se expresa como una desconexión entre oferta y demanda: existe oferta de eventos y artistas, pero falta un mecanismo estructurado y centralizado para concentrar información, permitir exploración geolocalizada y sostener continuidad operacional.

En términos prácticos, la audiencia depende de publicaciones efímeras o recomendaciones personales, mientras que organizadores y artistas dependen de redes cercanas y coordinación informal.

- Difusión fragmentada: múltiples canales, información duplicada y baja permanencia.
- Baja trazabilidad: ausencia de registro formal de eventos previos y resultados.
- Coordinación informal: mensajes dispersos dificultan decisiones y seguimiento.
- Fricciones de descubrimiento: falta de filtros simples por fecha y lugar.

1.3 Preguntas de investigación y enfoque

Desde la perspectiva de gestión de producto, la pregunta central se reformula en términos de valor: ¿qué combinación mínima de funcionalidades permite que usuarios reales encuentren un evento y perciban utilidad suficiente para repetir el uso? A partir de esta pregunta, se establecen subpreguntas orientadas a hipótesis y decisión (perseverar, iterar o pivotar).

1.4 Objetivo general

Gestionar el diseño, planificación y evolución del producto BeatMap desde el rol de Product Manager (Product Owner), definiendo el MVP, el backlog y un proceso de validación empírica en terreno para una plataforma geolocalizada de eventos musicales emergentes.

1.5 Objetivos específicos

- Definir el rol y responsabilidades del Product Manager dentro del proyecto
- Analizar metodologías ágiles aplicables a la gestión de producto
- Definir mecanismos de levantamiento y gestión de requerimientos
- Establecer criterios de priorización del producto
- Analizar la gestión de stakeholders y la comunicación del producto
- Identificar herramientas de apoyo al Product Management

Objetivo Específico	Marco Conceptual	Propuesta de Solución
Definir el rol y responsabilidades del Product Manager dentro del proyecto	Concepto de Product Management; rol del Product Manager en el desarrollo de software; responsabilidades principales; relación con el equipo de desarrollo y stakeholders; enfoque centrado en el valor del producto.	Establecer formalmente el rol de Product Manager dentro del equipo, delimitando sus responsabilidades y su interacción con los distintos actores del proyecto.
Analizar metodologías ágiles aplicables a la gestión de producto	Metodologías ágiles: Scrum y Kanban; principios del Manifiesto Ágil; rol del Product Manager en entornos ágiles; adaptación de metodologías al contexto del proyecto.	Seleccionar y aplicar una metodología ágil que permita una gestión eficiente del producto y una correcta coordinación con el equipo de desarrollo.
Definir mecanismos de levantamiento y gestión de requerimientos	Requerimientos funcionales y no funcionales; historias de usuario; criterios de aceptación; backlog de producto; refinamiento de backlog.	Definir y documentar los requerimientos del sistema mediante historias de usuario, manteniendo un backlog priorizado y alineado con los objetivos del producto.
Establecer criterios de priorización del producto	Concepto de priorización en Product Management; técnicas de priorización (MoSCoW, valor vs esfuerzo); roadmap de producto; toma de decisiones basada en valor.	Aplicar una técnica de priorización para ordenar las funcionalidades del producto y definir un roadmap acorde a las necesidades del proyecto.
Analizar la gestión de stakeholders y la comunicación del producto	Concepto de stakeholders; gestión de expectativas; comunicación efectiva en proyectos de software; documentación funcional del producto.	Definir una estrategia de comunicación que permita coordinar al equipo, informar avances y gestionar expectativas de los stakeholders.
Identificar herramientas de	Herramientas de gestión de	Seleccionar herramientas que

apoyo al Product Management	producto; herramientas de documentación; herramientas de prototipado; herramientas de comunicación y seguimiento.	faciliten la planificación, documentación y seguimiento del producto durante el desarrollo del proyecto.
Definir mecanismos de validación del producto	Validación de funcionalidades; aceptación del producto; feedback del usuario; mejora continua basada en métricas y resultados.	Establecer criterios de validación que permitan verificar que las funcionalidades desarrolladas cumplen con los objetivos del producto y las necesidades del usuario.

Tabla 2: Relación entre objetivos específicos, marco conceptual y propuesta.

Fuente: elaboración propia.

Explicar el modelo de negocio de BeatMap para un lector no experto, conectando propuesta de valor y sostenibilidad.

Definir roles, segmentos, flujos críticos y requisitos funcionales del MVP.

Comparar metodologías de gestión (predictiva vs ágil) y fundamentar la selección de un enfoque híbrido (marco de gestión del proyecto + Lean Startup + Scrum).

Documentar herramientas y gobernanza de ejecución: Jira, Google Calendar, GitHub PR, revisiones del equipo y revisión de código por *branch*.

Ejecutar validación en terreno y test de usabilidad, analizando métricas (CSAT) y derivando decisiones de backlog.

1.6 Alcance, supuestos y limitaciones

El alcance se define de forma incremental. El núcleo del MVP corresponde a un *tracker* público geolocalizado con exploración básica y visualización de eventos. La reformulación del producto establece roles y un onboarding por etapas (Register y Complete Profile) como base para funciones futuras (organizador crea eventos; artista postula). Se asume disponibilidad de usuarios para validación inicial y la existencia de eventos reales en el período de observación. Entre las limitaciones se incluyen la muestra por conveniencia, variabilidad de dispositivos móviles y restricciones de tiempo académico.

Categoría	Incluido	Excluido (por fase)
Descubrimiento (Oyente)	Tracker público, mapa/pines, detalle evento, filtros básicos	Recomendación avanzada/IA, notificaciones complejas
Onboarding y roles	Register + Complete Profile (rol, ciudad, redes opc.)	Verificación avanzada, perfiles extendidos
Operación de eventos	Publicación por organizador (fase siguiente), postulación (fase siguiente)	Ticketing completo, pagos integrados
Medición	Métricas mínimas (planificadas)	BI avanzado, dashboards complejos

Tabla 3: Alcance del MVP (definición inicial)

Fuente: Elaboración Propia.

1.7 Aporte y relevancia

El aporte principal de esta memoria se concentra en la dimensión de gestión de producto, específicamente en la capacidad de transformar un problema observable del entorno real en una solución digital validable, gobernada por un proceso estructurado de planificación, ejecución y control. A diferencia de trabajos centrados exclusivamente en la implementación técnica, este proyecto aborda la problemática desde una perspectiva sistémica, donde las decisiones de alcance, priorización y validación se sustentan en evidencia empírica y en marcos metodológicos reconocidos, tales como el marco de gestión de proyectos y los principios de Lean Startup.

Desde esta perspectiva, la problemática no se limita a la inexistencia de herramientas tecnológicas, sino a la ausencia de procesos formales que permitan identificar necesidades reales, reducir incertidumbre y orientar el desarrollo hacia la generación de valor. BeatMap se utiliza como caso de estudio para demostrar que, en contextos de alta ambigüedad, la gestión del producto cumple un rol clave al articular negocio, usuarios y tecnología, evitando la construcción de soluciones desconectadas del problema original.

La aplicación de prácticas de Product Management permitió estructurar el proyecto en torno a ciclos de planificación claros, mecanismos de control y validación continua, y una gestión consciente de recursos limitados. Esto se tradujo en decisiones explícitas de recorte, reformulación y priorización, las cuales fueron documentadas y justificadas en función del aprendizaje obtenido durante el desarrollo. De este modo, el problema abordado no solo se define en términos técnicos, sino también como un desafío de gestión, coordinación y toma de decisiones informadas en el desarrollo de productos digitales.

Área	Cómo se aplicó	Evidencia a adjuntar
Planificación	Gantt + Jira (épicas/sprints)	Figura Gantt; Tablas de sprints; Backlog priorizado
Ejecución y liderazgo	Cadencia de reuniones	Tabla de reuniones; Google Calendar; minutas
Gestión de riesgos	Planes alternativos	Anexo de registro de riesgos y contingencias
Control y seguimiento	Seguimiento de avance	Jira (estado de issues); reportes de sprint
Gestión de recursos	Priorización por capacidad	Decisiones de recorte/ajuste de alcance
Comunicación	Presentaciones y coordinación	Anexo de presentaciones; correos/minutas
Cierre	Consolidación de incrementos	Checklist de entrega; evidencia de validación

Tabla 4: Marco de gestión del proyecto

Fuente: Elaboración Propia.

CAPÍTULO 2: MARCO CONCEPTUAL

2.1 Metodologías ágiles: contexto general

Las metodologías ágiles corresponden a un conjunto de enfoques para la gestión y desarrollo de proyectos, particularmente de software, que priorizan la entrega incremental de valor, la adaptación al cambio y la colaboración continua entre los actores involucrados. A diferencia de los modelos tradicionales secuenciales, las metodologías ágiles asumen que los requisitos pueden evolucionar durante el desarrollo y que el aprendizaje obtenido en cada iteración debe retroalimentar la planificación futura.

En el contexto de BeatMap, este paradigma resultó especialmente pertinente, dado que el problema a resolver no se encontraba completamente definido desde el inicio y requería validación constante con usuarios reales (artistas, organizadores y asistentes a eventos).

2.2 ¿Qué es Scrum?

Scrum es un marco de trabajo ágil orientado a la entrega incremental de productos complejos mediante ciclos cortos de trabajo denominados sprints. Define roles claros, eventos recurrentes y artefactos que permiten estructurar el trabajo, mantener transparencia y facilitar la inspección y adaptación.

Entre sus elementos principales se encuentran el Product Backlog, el Sprint Backlog y el incremento de producto; así como eventos como la planificación del sprint, reuniones diarias, revisión y retrospectiva. Si bien Scrum ofrece una estructura robusta para equipos de desarrollo, su adopción estricta puede resultar exigente en contextos donde los roles no se encuentran completamente separados o donde el equipo cumple funciones híbridas.

2.3 ¿Qué Es Kanban?

Kanban es un método ágil enfocado en la visualización del flujo de trabajo y en la limitación del trabajo en curso (WIP). A diferencia de Scrum, no prescribe iteraciones fijas ni roles formales, permitiendo una gestión más continua y flexible de las tareas.

Este enfoque resulta útil en contextos donde las prioridades cambian con frecuencia o donde se requiere una respuesta rápida a incidencias. Sin embargo, su menor énfasis en la planificación por hitos puede dificultar la coordinación cuando existen objetivos de mediano plazo claramente definidos, como fue el caso de la entrega de un MVP validado en un horizonte académico específico.

2.4 ¿Qué es Lean Startup?

Lean Startup es un enfoque orientado a la creación de productos en contextos de alta incertidumbre, centrado en el aprendizaje validado. Su principio fundamental es el ciclo Construir–Medir–Aprender, mediante el cual se busca minimizar el desperdicio y maximizar el aprendizaje a partir de experimentos controlados con usuarios reales.

En BeatMap, Lean Startup fue especialmente relevante para guiar la reformulación del proyecto en sus primeras etapas. La idea original evolucionó en dos ocasiones a partir de la evidencia obtenida en terreno, hasta consolidar una propuesta enfocada en artistas y organizadores. Este enfoque permitió justificar la

construcción de un MVP acotado, orientado a validar hipótesis críticas antes de escalar funcionalidades.

2.5 Marco de gestión del proyecto: planificación y control aplicados a un producto digital

marco de gestión del proyecto se utiliza como guía para sostener prácticas que suelen ser críticas incluso en contextos ágiles: gestión de alcance, cronograma, riesgos, comunicaciones, *stakeholders* y calidad. En esta memoria no se ejecuta un ciclo marco de gestión del proyecto completo de forma rígida; más bien, se adoptan sus principios para estructurar el trabajo y justificar decisiones ante *stakeholders* académicos y del dominio.

Área PMBOK	Aplicación en BeatMap	Evidencia/artefacto
Alcance	Definición de MVP y exclusiones por fase	Tabla 1.1; backlog priorizado en Jira
Cronograma	Roadmap y sprints; carta Gantt complementaria	Artefacto 9 (Gantt/roadmap)
Calidad	DoD, PR, revisión de código por branch, revisiones del equipo	Sección 4.4; PRs en GitHub
Riesgos	Iteración temprana sobre UX móvil; hipótesis falsables	Tabla 5.2; Artefacto 12
Stakeholders	Roles: oyente/organizador/artista; validación en terreno	Cap. 3 y 5
Comunicaciones	Dailys + reuniones día por medio (Calendar)	Tabla 4.1; Figura 4

Tabla 5: Mapeo de prácticas del proyecto a áreas de marco de gestión del proyecto

Fuente: Elaboración Propia.

2.6 Lean Startup aplicado: hipótesis, experimentos y aprendizaje validado

Lean Startup plantea que el aprendizaje validado debe guiar el desarrollo cuando el riesgo principal no es técnico sino de mercado/usuario. En BeatMap, las hipótesis se definieron de forma falsable (por ejemplo, satisfacción del usuario al completar una tarea en el *tracker*) y se validaron mediante experimentos en terreno con métricas como CSAT. La regla de decisión (GO/ITERATE/PIVOT) se estableció antes de ejecutar el experimento, evitando reinterpretaciones posteriores.

En particular, la validación de H1 (valor del *tracker*) se implementó como test de usabilidad con tarea única y evaluación de satisfacción. Esta práctica permite reducir el riesgo más crítico del MVP: que el usuario no logre descubrir eventos o perciba alta fricción en móvil.

Concepto	Aplicación en BeatMap	Resultado/decisión
Hipótesis falsable	H1: satisfacción al usar tracker	CSAT promedio 3.545; decisión ITERATE
Experimento	Test en terreno con tarea guiada	Hallazgos UX móvil; filtros y lista bajo mapa
Métrica accionable	CSAT 1–5 y distribución (4–5 / 3 / 1–2)	Priorización de backlog para mejorar UX
Build–Measure–Learn	Iteraciones sobre tracker y estabilidad	Evolución del MVP antes de escalar canal

Tabla 6: Conceptos Lean Startup utilizados.

Fuente: Elaboración Propia.

2.7 Scrum aplicado: entrega incremental y coordinación del equipo

En BeatMap, Scrum se aplicó como estructura mínima para ordenar la entrega y sostener una cadencia de trabajo. Sin embargo, el proyecto enfrentó incertidumbre alta por la reformulación: el backlog cambiaba al incorporar evidencia de terreno. Por ello, se adoptó un enfoque híbrido, combinando prácticas de Scrum con visualización de flujo tipo Kanban en Jira para gestionar urgencias, bloqueos y tareas administrativas sin perder el objetivo del sprint.

criterio	Scrum (base)	Kanban (flujo)	Cascada (referencia)
Unidad de planificación	Sprint	Flujo continuo	Fase
Cambio de requerimientos	Se gestiona por sprint con refinamiento	Se gestiona continuamente	Costoso; se evita
Ventaja principal	Foco y revisión periódica del incremento	Visibilidad de bloqueos y WIP	Planificación detallada inicial
Riesgo principal	Rigidez si el sprint se interpreta como contrato	Pérdida de foco si no existe objetivo	Validación tardía
Ajuste al caso BeatMap	Estructuró reuniones y entregables	Complementó coordinación diaria	No apto por reformulación e incertidumbre

Tabla 7: Comparación resumida de enfoques de trabajo utilizados como referencia.

Fuente: Elaboración propia.

Esta comparación se utilizó como soporte para justificar el enfoque de trabajo. En la práctica, el equipo sostuvo reuniones frecuentes (dailys con objetivos, reuniones día por medio para seguimiento del sprint), y mantuvo la trazabilidad de decisiones a través de historias de usuario y criterios de aceptación.

Scrum se adoptó como marco de trabajo para estructurar ciclos cortos, inspección y adaptación. La memoria no aborda roles por identificación personal, sino prácticas observables: objetivos de sprint, seguimiento diario, coordinación frecuente y revisión de avance. El foco desde producto se centra en la priorización del backlog, definición de criterios de aceptación y validación funcional de entregas.

En el proyecto se sostuvieron dailys calendarizadas y reuniones día por medio asociadas al avance del sprint. Esta cadencia permitió detectar bloqueos, reasignar tareas, y ajustar alcance sin perder el objetivo del incremento.

2.8 Comparación metodológica y justificación del enfoque híbrido

Enfoque	Ventaja	Riesgo principal	Adecuación a BeatMap
Predictivo (cascada)	Planificación completa antes de construir	Validación tardía; cambios costosos	Baja: dominio requiere aprendizaje temprano
Ágil puro (sin gobernanza)	Velocidad e iteración	Scope creep; decisiones no trazables	Media: se requería control por evidencia
Híbrido (PMBOK + Lean + Scrum)	Control + aprendizaje + entrega incremental	Requiere disciplina y documentación mínima	Alta: permitió planificar y ajustar con evidencia

Tabla 8: Comparación de enfoques de gestión

Fuente: Elaboración Propia.

Se concluye que el enfoque híbrido fue el más apropiado: marco de gestión del proyecto entrega estructura mínima para justificar alcance, riesgos y calidad; Lean Startup asegura que el producto se mida contra comportamiento real; y Scrum permite iterar con incrementos cortos y controlables.

2.8.1 Comparación de metodologías ágiles relevantes

Lean Startup no reemplazó Scrum: se utilizó para decidir qué construir. Dado que BeatMap nació con alta incertidumbre sobre el dolor real y sobre quién debía generar el contenido, se necesitó un mecanismo explícito para formular hipótesis, medir y aprender rápido. Por eso se incorporaron entrevistas exploratorias (etapa 1) y tests de usabilidad (etapa 2) como experimentos; los resultados se tradujeron a cambios concretos de backlog, evitando construir funcionalidades solo por intuición.

Se compararon enfoques ágiles comúnmente utilizados en equipos de software: Scrum, Kanban, Extreme Programming (XP) y Scrumban. La comparación se realizó bajo criterios prácticos para BeatMap: incertidumbre del producto, necesidad de coordinar entregables con validación en terreno, tamaño del equipo y necesidad de trazabilidad. El resultado respalda la decisión de usar Scrum como marco de entrega, complementado con prácticas de Kanban para visualizar flujo y con principios de Lean Startup para aprendizaje validado.

2.9 Integración de Scrum, Kanban y Lean Startup en contextos de incertidumbre

Para que un lector sin experiencia previa pueda comprender la elección metodológica, resulta útil distinguir entre **marcos de gestión del trabajo** y **marcos de aprendizaje del producto**. Scrum y Kanban se ubican principalmente en el primer grupo: su propósito es ordenar la ejecución, hacer visible el avance, coordinar al equipo y reducir la incertidumbre operativa asociada a plazos, dependencias y carga de trabajo. Lean Startup, en cambio, se posiciona en el segundo grupo, ya que su foco está en reducir la incertidumbre respecto del problema y del mercado mediante hipótesis explícitas, experimentos controlados y aprendizaje validado.

En el caso de BeatMap, el desafío no se limitaba a construir un sistema funcional, sino a **determinar qué sistema tenía sentido construir**. La idea inicial del producto fue reformulada en dos oportunidades, no por fallas técnicas, sino porque la evidencia levantada en terreno mostró que la propuesta original no resolvía de forma directa las necesidades reales de los actores involucrados. En este contexto, Lean Startup operó como un marco de decisión estratégica, orientando qué hipótesis validar, qué funcionalidades descartar y cuáles postergar, mientras que Scrum y prácticas inspiradas en Kanban permitieron transformar esas decisiones en entregas incrementales y controladas.

Esta integración metodológica tiene un fundamento ingenieril claro: el proyecto se abordó como un **sistema socio-técnico**, donde el desempeño del producto depende tanto de decisiones técnicas (arquitectura, módulos, requerimientos) como de decisiones humanas y organizacionales (coordinación del equipo, comunicación, adopción por parte de usuarios). Por esta razón, el Marco Conceptual se estructura de manera articulada, conectando conceptos de gestión de proyectos, desarrollo de software y validación de producto, evitando presentar cada metodología como un conjunto aislado de reglas.

Finalmente, se reconoce explícitamente que ninguna metodología, aplicada de forma rígida, resuelve por sí sola los desafíos de un proyecto académico con componentes de innovación. En este tipo de contextos, el objetivo no es “cumplir” una metodología de manera estricta, sino **sostener un proceso controlado, trazable y adaptable**, que permita tomar decisiones informadas frente a la incertidumbre. BeatMap se desarrolló bajo esta premisa, priorizando un enfoque pragmático, centrado en objetivos, evidencia y decisiones de producto, por sobre la adhesión formal a rituales o prácticas prescriptivas.

Metodología	Cuándo calza mejor	Ventajas principales	Limitaciones	Prácticas compatibles	Justificación en BeatMap
Scrum	Trabajo por incrementos con coordinación frecuente	Ritmo estable; facilita compromiso; revisión por sprint	Riesgo de rigidez si el backlog cambia a diario	Historias, criterios, review/retro	Base del trabajo: sprints y metas claras

REFORMULACIÓN Y GESTIÓN DE PRODUCTO DE UNA PLATAFORMA GEOLOCALIZADA DE EVENTOS PARA ARTISTAS EMERGENTES: CASO BEATMAP

Kanban	Flujo continuo; cambios frecuentes de prioridad	Visibilidad; limita WIP; reduce cuellos de botella	Menos útil para hitos por sprint	Tablero, WIP, lead time	Apoyo para bloqueos y tareas administrativas
XP	Alta exigencia técnica y calidad	Reduce defectos; mejora mantenibilidad	Costoso si el equipo no está habituado	TDD, pair, refactor	Adopción parcial: code review y estándares
Scrumban	Scrum con necesidad de más flexibilidad	Mezcla planificación por sprint y flujo	Puede perder cadencia si no se gobierna	Backlog + tablero continuo	Alternativa, pero se priorizó claridad Scrum + Lean

Tabla 9: Ventajas y desventajas de aplicar las Metodologías Ágiles en Beatmap.

Fuente: elaboración propia.

CAPÍTULO 3: PROPUESTA DE SOLUCION

3.1 ¿Qué es un modelo de negocio? (explicación desde cero)

Un modelo de negocio describe cómo una solución crea valor, lo entrega y lo captura. En productos digitales, esta definición es relevante incluso si el proyecto inicia como prototipo académico, ya que condiciona decisiones como: qué funcionalidades son públicas, qué actores deben autenticarse, qué costos se generan y qué estrategias permitirían sostenibilidad. Por ello, BeatMap se fundamenta mediante un Modelo de Negocio tipo Canvas como herramienta de comunicación para personas no expertas.

3.2 Segmentos, roles y propuesta de valor

La reformulación final del producto se construyó después de dos ajustes de enfoque (ver Tabla 1.1). En la versión consolidada, BeatMap define tres roles, pero prioriza dos actores de negocio: artistas y organizadores. El oyente se mantiene como usuario final de descubrimiento (mapa y detalle de eventos), porque su participación valida la utilidad del sistema; sin embargo, la sostenibilidad del contenido depende de que artistas y organizadores puedan publicar y mantener información de forma simple. En términos de módulos, esta decisión se reflejó en: perfiles de artista y organizador, publicación y edición de eventos con reglas mínimas de completitud, mapa geolocalizado con filtros básicos, y un detalle de evento orientado a comprensión rápida (qué, dónde, cuándo, quién y cómo asistir).

Segmento	Dolor actual	Propuesta de valor BeatMap
Oyente	No encuentra tocatas a tiempo; info dispersa	Tracker geolocalizado y filtros para descubrir eventos
Organizador	Difusión fragmentada; coordinación informal; baja trazabilidad	Publicación centralizada; orden; base para métricas
Artista/Banda	Dependencia de contactos; baja visibilidad	Acceso a eventos y postulación con mayor transparencia

Tabla 10: Segmentos y propuesta de valor

Fuente: Elaboración Propia.

3.3 Business Model Canvas (resumen)

Bloque	Contenido BeatMap
Segmentos de clientes	Oyentes de música local; organizadores; artistas/bandas
Propuesta de valor	Centralización de eventos emergentes con geolocalización; reducción de pérdida de información
Canales	Plataforma web/móvil; redes sociales; alianzas con eventos/locales/organizaciones
Relación con clientes	Autoservicio para oyentes; registro por rol para organizadores/artistas
Fuentes de ingreso (futuro)	Promoción destacada; patrocinios; planes para organizadores; servicios B2B locales
Recursos clave	Plataforma; base de eventos; comunidad; infraestructura técnica
Actividades clave	Curación/publicación; mejoras UX; soporte; validación; gestión de calidad
Socios clave	Locales, productoras, universidades, comunidades culturales
Estructura de costos	Infraestructura; desarrollo; mantenimiento; marketing mínimo

Tabla 11: Business Model Canvas (BMC) resumido
Fuente: Elaboración Propia.

3.4 Estimación preliminar de unit economics

Para dimensionar la viabilidad económica en etapas tempranas, se elaboró una estimación preliminar de *unit economics* basada en supuestos. Este análisis se utilizó como instrumento de decisión para estimar órdenes de magnitud (costos, adquisición y posibles ingresos), y **no** como un resultado final. En un producto digital como BeatMap, este ejercicio cumple un rol práctico: permite anticipar restricciones (por ejemplo, costo de adquisición y costos operativos) y evita diseñar una propuesta que sea inviable incluso si la solución funciona técnicamente.

En la estimación interna se consideró un esquema mixto de ingresos a futuro (por ejemplo, promoción destacada y publicidad en el *tracker*), contrastando escenarios de adopción gradual. Dado que el proyecto se desarrolló sin financiamiento externo y con validación presencial costada por el equipo, la estimación también ayuda a justificar por

qué el MVP prioriza primero el flujo de valor (publicación → descubrimiento → decisión) antes que módulos “atractivos” pero de baja contribución inmediata.

Conceptos utilizados (definición desde cero):

- **CPC (Costo por Clic):** costo promedio asociado a conseguir un clic en una campaña o acción de difusión. Se utiliza como aproximación cuando la adquisición depende de anuncios o difusión pagada.
 - Fórmula típica: $CPC = \text{gasto en campaña} / \text{número de clics}$.
- **CAC (Costo de Adquisición de Cliente):** costo promedio de adquirir un usuario/cliente que efectivamente se registra o realiza la acción objetivo (por ejemplo, organizador que publica o artista que postula).
 - Fórmula típica: $CAC = \text{gasto total de adquisición} / \text{número de clientes adquiridos}$.
- En etapas tempranas, CPC y CAC **no son valores “verdaderos”**, sino estimaciones que se ajustan cuando existen datos reales (métricas observadas de conversión, retención y recurrencia).

A modo referencial, el documento interno consideró supuestos como CPC aproximado de \$200 CLP y un CAC estimado de \$6.000 CLP, además de un margen neto por usuario estimado de \$8.426 CLP y escenarios de OPEX mensual del orden de \$4,8 millones CLP. Estos valores deben entenderse como hipótesis económicas sujetas a ajuste tras obtener datos reales de uso y adquisición.

Concepto	Qué representa	Para qué se usa en BeatMap	Riesgo si no se considera
CPC	costo por interacción (clic)	aproximar costo de atraer tráfico	subestimar gasto real de difusión
CAC	costo por adquisición efectiva	estimar costo de lograr organizadores/artistas activos	diseñar un modelo inviable
OPEX	gasto operativo mensual	dimensionar sostenibilidad mínima	no anticipar costos fijos
Margen neto estimado	diferencia ingresos–costos por usuario	comparar escenarios de monetización	sobreestimar ingresos tempranos

Tabla 12: Definiciones económicas utilizadas en la estimación temprana

Fuente: Elaboración propia.

3.5 Competidores y alternativas actuales

El análisis competitivo se aborda desde la perspectiva de alternativas de solución, no solo plataformas directas. En el contexto local, las alternativas típicas incluyen redes sociales (Instagram/Facebook), mensajería (WhatsApp/Telegram), y difusión por cuentas de terceros. Estas alternativas son eficaces para difusión puntual, pero fallan en permanencia, búsqueda estructurada y trazabilidad.

Alternativa	Fortaleza	Debilidad	Diferenciación BeatMap
Instagram/Stories	Alcance orgánico rápido	Información efímera; sin filtros; difícil de buscar	Tracker persistente y filtrable
Facebook Events	Estructura de evento	Menor uso en público joven; fragmentación local	Foco local + geolocalización + comunidad
WhatsApp/Grupos	Coordinación inmediata	No escalable; sin registro; difícil de difundir	Visibilidad pública + trazabilidad futura
Google Maps/Reseñas	Ubicación	No orientado a eventos; sin curación temporal	Eventos con fecha/hora y detalle

Tabla 13: Comparación con alternativas existentes

Fuente: Elaboración Propia

3.6 Requisitos y flujos críticos del MVP

Para aterrizar los flujos del MVP y reducir ambigüedad, se trabajó con wireframes de baja fidelidad en Excalidraw. Estos bocetos se iteraron junto al equipo de frontend y el resto del equipo, ajustando pantallas por ciclos cortos según retroalimentación interna, coherencia con la lógica de negocio y hallazgos levantados en terreno. El objetivo principal del prototipado no fue estético, sino **decisional**: definir qué información debía estar presente, qué acciones debían estar disponibles por rol y qué respuesta debía entregar el sistema en cada flujo crítico.

Desde la gestión de producto, la estructuración de requerimientos se abordó como un proceso progresivo (no como una lista estática), con el propósito de mejorar el producto sin perder trazabilidad:

1. **Levantamiento del problema y contexto** (validación exploratoria): identificación de fricciones reales y lenguaje del usuario.

2. **Traducción a comportamientos y necesidades:** qué intenta lograr cada actor (oyente, artista, organizador) y qué bloqueos enfrenta.
3. **Formalización en casos de uso:** actor, precondiciones, flujo principal, alternativos, resultado esperado.
4. **Derivación de requerimientos funcionales y no funcionales:** lo mínimo para que el MVP sea demostrable y validable.
5. **Conversión a historias de usuario** con criterios de aceptación verificables: qué se considera “cumplido”.
6. **Priorización y recorte por iteración:** qué entra al MVP y qué se posterga, con reglas explícitas.

Este enfoque permite sostener coherencia entre: evidencia levantada → decisión de producto → requerimiento → historia → entrega.

3.6.1 Casos de uso y comportamiento esperado del sistema

Bajo el rol de Product Manager, la gestión de producto definió casos de uso como traducción directa de problemáticas levantadas en terreno. Cada caso de uso describió actor, precondiciones, flujo principal, flujos alternativos y resultado esperado. Esto permitió que el backlog no se transformara en tareas sueltas, sino en una representación coherente de valor para el usuario.

Los casos de uso priorizados se concentraron en: **(i)** descubrir eventos cercanos (mapa/lista), **(ii)** revisar detalle estructurado de evento, **(iii)** publicación y administración de eventos/convocatorias para organizadores, **(iv)** postulación de artistas y seguimiento del estado, **(v)** perfil por rol con enlaces y antecedentes mínimos.

3.6.2 Requerimientos funcionales y no funcionales (síntesis)

Los requerimientos funcionales se levantaron desde historias y casos de uso, priorizando aquellos que habilitan el flujo mínimo de valor: publicación → descubrimiento → decisión. En paralelo, se definieron requerimientos no funcionales para asegurar que el MVP fuese demostrable y usable, considerando seguridad, rendimiento de mapa y consistencia de datos.

Tipo	Ejemplos priorizados (resumen)	Justificación PM
RF	Mapa con pines; filtros básicos; detalle de evento; CRUD de evento; autenticación/perfil	Sin estos RF no hay propuesta de valor verificable
RNF	Seguridad (roles/JWT); rendimiento de carga; trazabilidad básica; usabilidad mínima	Reduce riesgo de demo fallida y permite validación real

Tabla 14: Síntesis de requerimientos RF/RNF priorizados para el MVP.

Fuente: elaboración propia.

3.6.3 Historias de usuario, criterios de aceptación y Definition of Done (DoD)

Para mejorar la trazabilidad entre objetivos, requerimientos y construcción del MVP, las funcionalidades principales se expresaron como historias de usuario. Este formato es útil en gestión de producto porque evita describir “pantallas” o “tareas técnicas” sin valor, y obliga a mantener el foco en: actor + necesidad + propósito.

A continuación, se incorporan historias representativas (usuarios, organizadores y artistas), con criterios de aceptación verificables. Estas historias sirven como base para el refinamiento del backlog y para revisar si cada incremento realmente cumple el objetivo del sprint.

ID	Como	Necesito	Para	Prioridad	Criterios de aceptación (síntesis)	Caso de uso
H U-01	Usuario visitante	ver eventos disponibles en su ciudad	asistir a eventos de interés	Alta	se detecta/selecciona ciudad; lista por fecha; filtros (género/tipo/gratis-pago); detalle visible sin registro	I y II
H U-02	Organizador	publicar convocatoria/evento	recibir postulaciones	Alta	formulario solicita requisitos (género/horario/tipo/pago); artistas postulan; organizador gestiona postulaciones	III
H U-03	Organizador	ver estadísticas básicas	evaluar éxito/mejorar decisiones	Media	métricas mínimas (postulaciones/asistencia est.); actualización tras finalizar evento (si aplica al MVP)	III
H U-04	Organizador	perfil profesional con historial	generar confianza	Alta	perfil con datos base + links; muestra eventos pasados publicados; comentarios/valoraciones si aplica	V
H U-05	Organizador	evaluar artista tras evento	ayudar a futuras decisiones	Media	solo si evento finalizado; puntaje 1-5 + comentario opcional; 1 evaluación por evento/artista	III

H U- 06	Artista	crear y personalizar perfil	mostrar portafolio	Alta	datos (nombre/ciudad/bio/géneros); links (Spotify/YouTube/IG); perfil accesible desde postulaciones	V
H U- 07	Artista	postular a convocatoria	ser considerado	Alta	lista convocatorias abiertas; postulación con comentario opcional; organizador notificado; estado (pendiente/aceptada/recusada)	IV
H U- 08	Artista	revisar estado y trazabilidad	tomar decisiones	Media	historial de postulaciones; estado visible; fecha y organizador asociado	IV
H U- 09	Artista	evaluar organizador	informar a otros artistas	Media	solo si participación confirmada; puntaje 1-5; comentario opcional; 1 evaluación por evento/organizador	IV

Tabla 17: Historias de usuario priorizadas del MVP (resumen)

Fuente: Elaboración propia

- historia cumple criterios de aceptación del ticket.
- revisión por *Pull Request* con al menos 1 revisor distinto a quien desarrolló.
- flujo principal (*happy path*) validado en ambiente de prueba.
- no se detectan fallas críticas conocidas en rutas relacionadas.
- descripción mínima del PR y vínculo al ticket cuando aplica.

Criterio	Descripción operativa
Revisión de PR	PR revisado y aprobado por al menos un integrante distinto
Criterios de aceptación	comportamiento cumple lo definido en la historia
Happy path	flujo principal ejecutado sin fallas visibles
No regresión	rutas críticas previas mantienen funcionamiento
Documentación mínima	PR describe cambio y referencia ticket cuando aplica

Tabla 18: *Definition of Done* (DoD) técnico-funcional (resumen)
Fuente: Elaboración propia.

3.6.4 Priorización del MVP y reglas de recorte por iteración

La priorización del MVP se realizó con criterio mixto: valor para el usuario (impacto directo en el flujo publicación–descubrimiento–decisión), esfuerzo estimado según capacidad del equipo y riesgo técnico/organizacional. La lógica de priorización se justifica en un contexto de proyecto académico con restricciones reales: tiempo limitado, validación en terreno costeadada por el equipo y necesidad de entregar un incremento demostrable.

Para evitar sobrecarga, la gestión de producto aplicó reglas explícitas de recorte: si una funcionalidad no fortalecía el flujo principal o no reducía incertidumbre relevante, se postergaba. Al evaluar nuevas ideas, se utilizó un filtro consistente:

- ¿qué problema real resuelve y para qué rol?
- ¿cómo se valida (evidencia concreta) y en qué plazo?
- ¿qué se sacrifica si entra ahora?
- ¿cuál es la versión mínima que cumple el objetivo?

Criterio	Pregunta guía	Decisión típica
Valor directo	¿Aporta al flujo principal del usuario?	Si no aporta, se mueve a release posterior
Verificabilidad	¿Se puede validar con usuarios/terreno?	Si no se puede validar, se evita en MVP
Esfuerzo	¿Cabe en el sprint sin romper entregable?	Si excede, se recorta o se divide
Riesgo	¿Riesgo técnico alto sin plan B?	Se busca alternativa o mock/alcance mínimo

Tabla 15: Criterios utilizados para decidir si una historia entra o queda fuera del MVP.
Fuente: elaboración propia.

Como resultado, el backlog dejó de ser una “lista infinita” y pasó a ser un instrumento de decisión. Esto fue especialmente útil durante la reformulación, donde la tentación de sumar funcionalidades era alta, permitiendo filtrar tareas según su valor real para el MVP y postergar todo lo que no aportara evidencia inmediata para la implementación.

3.7 Roadmap y estrategia de releases

El Roadmap se definió por incrementos orientados a valor y reducción de riesgo. En vez de “sumar módulos”, se priorizó construir un núcleo utilizable y demostrable, porque en productos basados en comunidad el riesgo principal no es técnico: es construir una solución que no se adopta por falta de claridad o por fricción de uso.

- **Release 0 (credibilidad / punto de entrada):** landing + explicación del problema + registro de interés (cuando aplica).
- **Release 1 (valor visible):** *tracker* público + lista/detalle con filtros mínimos.
- **Release 2 (oferta y contenido):** organizador publica evento/convocatoria + artista postula.
- **Release 3 (reducción de fricción):** mejoras UX, estabilidad móvil y jerarquía de información.
- **Release 4 (crecimiento y sostenibilidad):** métricas, alianzas, difusión estructurada, monetización gradual.

Esta estrategia se justifica porque el éxito del MVP depende de dos condiciones simultáneas: (i) que el usuario descubra eventos con facilidad y (ii) que exista contenido actualizado. Por ello, el roadmap integra funcionalidades para el oyente (descubrimiento) y para el organizador/artista (contenido), evitando depender exclusivamente de difusión manual.

3.8 Arquitectura del producto y decisiones tecnológicas (síntesis)

Aunque el foco de esta memoria es la gestión de producto, se presenta una síntesis de decisiones técnicas porque afectaron alcance, riesgos y plazos. BeatMap se desarrolló con una arquitectura modular, separando frontend, backend y base de datos. Esta separación permitió trabajar en paralelo, reducir acoplamiento y facilitar despliegue incremental.

Gestión del tiempo y planificación orientada a hitos

Desde la gestión de producto, la planificación se trató como un puente entre el objetivo académico (entregar una memoria con un producto demostrable) y el objetivo de producto (probar que la propuesta era valiosa para usuarios reales). Para ello, la gestión del producto definió hitos claros: reformulación validada, definición del MVP, implementación de módulos críticos y una entrega demostrable con retroalimentación externa.

La planificación no se limitó a fechas; incluyó supuestos y dependencias. Por ejemplo, la priorización del mapa y el flujo de descubrimiento se vinculó directamente con la hipótesis principal de BeatMap, si las personas pueden encontrar eventos cercanos de forma simple, entonces aumenta la asistencia y mejora la difusión de artistas emergentes. Bajo esa hipótesis, funcionalidades accesorias se postergaron, aunque fueran atractivas a nivel técnico.

El control del tiempo se realizó combinando un Gantt de alto nivel (para visualizar hitos) con planificación iterativa por etapas (sprints) para organizar el trabajo semanal. Esta

combinación permitió mantener dirección, sin perder capacidad de ajuste. Cuando aparecieron desviaciones —por carga académica, bloqueos técnicos o ajustes de alcance— la gestión del producto propuso alternativas (plan B o C) que protegieran el objetivo del MVP sin comprometer la coherencia del producto.

En términos prácticos, esto se tradujo en decisiones de recorte deliberado: reducir alcance, simplificar reglas y priorizar la experiencia mínima completa antes que añadir capas de complejidad. Esta lógica es consistente con Lean Startup: asegurar aprendizaje y valor temprano antes de optimizar.

Organización del trabajo: cadencia de reuniones y seguimiento administrativo

La coordinación del equipo se apoyó en una cadencia de trabajo explícita. Se realizaron reuniones mayoritariamente diarias y breves (dailys) con objetivos diarios claros, y reuniones de seguimiento día por medio para alinear prioridades, despejar bloqueos y revisar el avance respecto del sprint. Este esquema permitió mantener foco sin sobrecargar al equipo con ceremonias extensas.

Google Calendar se utilizó como herramienta administrativa para sostener disciplina y trazabilidad ya que cuenta con la agenda de reuniones, objetivos por sesión y registros de hitos los que facilitaron el seguimiento. Esta práctica, aunque simple, fue relevante para un equipo con múltiples compromisos, ya que hizo visible la carga real de coordinación y redujo pérdidas por desalineación.

Desde la perspectiva del rol de PM, cada reunión tenía un propósito. La inspección rápida del progreso, decisiones de priorización y coordinación de dependencias. La clave fue evitar reuniones “por costumbre” y, en su lugar, tratarlas como un mecanismo de control del sistema.

Gestión de *stakeholders* y comunicación externa

BeatMap operó como un producto con múltiples actores los cuales fueron asistentes, artistas emergentes y organizadores. Además, existieron *stakeholders* académicos y comunitarios. La gestión del producto asumió la responsabilidad de mantener comunicación consistente y de traducir expectativas externas a objetivos internos alcanzables.

En paralelo, la difusión del proyecto en redes cumplió una función doble: construir comunidad y reclutar participantes para validaciones. Esta actividad se consideró un instrumento de investigación aplicada: permitió observar reacciones, comprender lenguaje y detectar fricciones en la propuesta de valor.

Cuando una necesidad de un actor entraba en conflicto con otra, el criterio principal fue el aprendizaje y el valor para el objetivo del MVP. Por ejemplo, se priorizó la claridad del descubrimiento y la información del evento antes que funciones avanzadas de personalización.

Calidad como práctica distribuida: revisión entre pares y control de cambios

No existió una fase separada de QA. En su lugar, la calidad se abordó como práctica transversal e integrada al flujo de trabajo. Cada funcionalidad se desarrolló en ramas por *feature* y se integró mediante *Pull Requests*, donde el equipo realizó revisiones cruzadas. Cuando un cambio no cumplía criterios acordados, se registraban observaciones y la integración se postergaba hasta corregir.

Desde la gestión de producto, este mecanismo se fortaleció definiendo expectativas mínimas de entrega: criterios de aceptación y una definición operativa de “listo” para cada incremento. Esto permitió que las revisiones verificaran comportamiento esperado y consistencia con el objetivo del sprint, además de aspectos de mantenibilidad.

Este enfoque aportó trazabilidad entre requerimientos y cambios, redujo retrabajo y promovió aprendizaje colectivo. También fue coherente con el contexto académico, al equilibrar velocidad y control sin depender de un rol dedicado.

El stack se eligió por disponibilidad de conocimiento en el equipo, soporte comunitario y rapidez de iteración: frontend en React + TypeScript, backend en Java Spring Boot con seguridad basada en JWT, y persistencia en PostgreSQL. Para el módulo geolocalizado, se priorizó performance de carga inicial y un modelo de datos que permitiera filtrar por fecha/categoría sin complejidad excesiva.

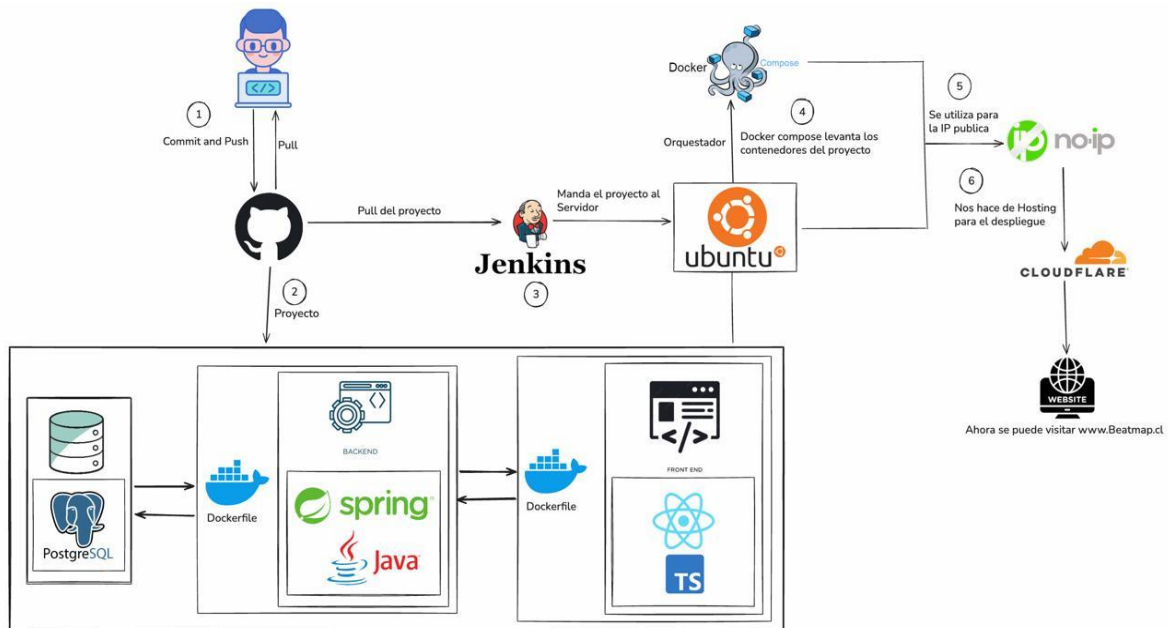


Figura 2: Arquitectura General Del Proyecto

Fuente: Dismar Quiroz

3.9 Documentación del producto y registro de decisiones

Capa de producto (qué y por qué):

Se Reunió el contexto y la lógica del producto: el problema observado en el ecosistema musical local, los roles priorizados (artista, organizador y oyente), la propuesta de valor y las hipótesis principales. También incluyó las decisiones de reformulación (por qué la

primera versión era demasiado amplia y por qué se acotó a flujos centrales), de manera que la visión no dependiera solo de conversaciones. Esta capa permitió responder, en cualquier momento, preguntas como: ¿qué dolor se está resolviendo?, ¿para quién?, ¿por qué este MVP y no otro?

Capa de requisitos (qué debe hacer):

Se Trabajo el problema en comportamiento verificable. Se trabajó con casos de uso, requerimientos funcionales y no funcionales (RF/RNF), historias de usuario y criterios de aceptación. La intención fue evitar interpretaciones distintas entre integrantes: cada historia debía describir actor, acción, resultado esperado y condiciones mínimas de cumplimiento. Esta capa también permitió sostener consistencia cuando el proyecto cambió de enfoque: al reformular, se ajustaron historias y criterios, evitando “arrastrar” funcionalidades que ya no aportaban al flujo principal.

Capa de ejecución (cómo se construye y controla):

Registró cómo se transformaban requisitos en trabajo. Incluyó el backlog organizado por épicas, estimaciones referenciales, acuerdos de integración (trabajo por ramas y revisiones), y una definición operativa de “hecho” (*Definition of Done*) que evitó entregas incompletas. Además, se documentaron reglas de recorte: qué condiciones justificaban postergar una historia (p. ej., no aporta al flujo principal, no es validable en el período, excede capacidad del sprint o presenta riesgo alto sin plan alternativo). Esta capa fue útil para sostener velocidad sin perder control, especialmente cuando el equipo debía compatibilizar el proyecto con carga académica y actividades en terreno.

Capa de evidencia (por qué se decidió así):

Concentró la base que justificó decisiones: entrevistas, notas de terreno, síntesis de hallazgos, resultados de pruebas de usabilidad (pauta, observaciones y notas), material audiovisual y registros de difusión en redes sociales. También incorporó supuestos económicos preliminares (unit economics) como herramienta de realidad para dimensionar órdenes de magnitud, sin tratarlos como cifras finales. Esta capa permitió conectar el proyecto con el método: no se trató de “opiniones del equipo”, sino de evidencia recopilada y sintetizada que sustentó la reformulación y la priorización del MVP.

Adicionalmente, se mantuvo un registro simple de decisiones relevantes, con una estructura consistente: qué decisión se tomó, qué evidencia la respaldó y qué impacto tuvo en el backlog (por ejemplo: historias eliminadas, recortadas o re-priorizadas). Este registro cumplió dos funciones: (i) reducir discusiones repetidas, porque los acuerdos quedaban visibles, y (ii) fortalecer la justificación ante evaluación académica, ya que BeatMap experimentó reformulaciones importantes antes de consolidar su versión por roles. En la práctica, este mecanismo operó como un “puente” entre aprendizaje en terreno y ejecución, asegurando que cada cambio de dirección tuviera fundamento y consecuencias explícitas en el trabajo planificado.

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN

4.1 Gestión Del Proyecto y Ejecución

4.1.1 Organización del trabajo y colaboración del equipo

El trabajo se ejecutó bajo un esquema colaborativo. Las revisiones de código no fueron delegadas a una sola persona: todo el equipo participó en *code reviews* y apoyo cruzado, lo que aumentó la transferencia de conocimiento y redujo el riesgo de puntos únicos de falla. Desde gestión de producto, esta práctica fortalece la capacidad de entregar valor de forma constante, especialmente cuando se requiere iterar rápido.

Desde la gestión del proyecto, la gestión del producto se enfocó principalmente en coordinación administrativa y conducción del trabajo del equipo. El diseño se desarrolló por el equipo de desarrollo y la gestión del producto aportó retroalimentación transversal, cuidando consistencia con el alcance y criterios de aceptación. Las responsabilidades de gestión se materializaron en:

- Planificación: definió el alcance, estableció objetivos, estructuró tareas, definió plazos y coordinó asignación de recursos según capacidad.
- Ejecución y liderazgo: condujo coordinación diaria, delegó tareas, sostuvo motivación y supervisó avance con foco en objetivos y bloqueos.
- Gestión de riesgos: identificó problemas potenciales y preparó alternativas (planes de contingencia) para evitar desviaciones relevantes.
- Control y seguimiento: monitoreó avance, comparó con lo planificado (Gantt + Jira) y coordinó correcciones cuando fue necesario.
- Gestión de recursos: administró uso del tiempo y del talento humano, priorizando lo esencial para cumplir objetivos por sprint e hitos.
- Comunicación: coordinó reuniones, preparó presentaciones de avance y mantuvo claridad sobre decisiones de alcance.
- Cierre: coordinó consolidación de incrementos, verificó criterios de aceptación y preparó evidencia para entrega.

Adicionalmente, la difusión de BeatMap en Instagram y LinkedIn se utilizó para visibilizar avances, atraer interés y construir comunidad. Estas acciones se consideraron complementarias a la validación en terreno, no un reemplazo de ella.

4.2 Gestión del backlog y sprints en Jira

Jira se utilizó como repositorio del *Product Backlog* y tablero operativo. En Jira se estructuraron épicas, historias de usuario y tareas técnicas, con estados de flujo (por ejemplo, *To Do*, *In Progress*, *Review*/revisión de código, *Done*). La priorización se alineó al objetivo del sprint y a decisiones derivadas de validación en terreno.

La gestión del backlog se mantuvo como una responsabilidad central de la gestión del producto en su rol de Product Manager. En términos operativos, esto implicó mantener claridad de alcance (qué se construye y qué se posterga), traducir hallazgos de validación a trabajo ejecutable y sostener la trazabilidad entre objetivos del sprint y tareas concretas.

En Jira se estructuró el trabajo mediante épicas asociadas a módulos del producto (por ejemplo: autenticación, perfil, CRUD de eventos, *Tracker* geolocalizado, panel administrativo y mejoras de experiencia). Cada épica se descompuso en historias y tareas, documentando criterios de aceptación y dependencias mínimas.

Durante el refinamiento, la gestión del producto coordinó la priorización considerando tres variables prácticas: (i) valor para el usuario en el MVP, (ii) esfuerzo estimado según capacidad del equipo, y (iii) riesgos técnicos u organizacionales (dependencias, incertidumbre, necesidad de validación previa).

Como resultado, cada sprint quedó asociado a un objetivo claro (sprint goal) y a un conjunto acotado de entregables. Cuando surgían hallazgos nuevos o impedimentos, el ajuste se realizaba sin perder el foco del *sprint*, reubicando tareas al *backlog* y registrando la decisión para evitar retrabajo y discusiones repetidas.

La coordinación del equipo se sostuvo mediante una cadencia definida desde el inicio y ajustada según el ritmo real de avance. Las *dailies* realizaron con objetivos explícitos: revisar el estado del día anterior, confirmar el objetivo del día, identificar bloqueos y acordar prioridades inmediatas. Este punto fue relevante, ya que permitió a la gestión del producto ejercer seguimiento sin microgestión, observando señales tempranas de desvío (tareas estancadas, dependencias no resueltas o exceso de trabajo en curso).

Adicionalmente, se realizaron reuniones de seguimiento día por medio, enfocadas en revisar avance de sprint, resolver decisiones de alcance y alinear próximos pasos. Estas reuniones también se utilizaron para preparar presentaciones internas y mantener un contexto compartido, especialmente durante la reformulación del proyecto, donde era fácil perder coherencia si no se registraban acuerdos.

Google Calendar se utilizó como soporte administrativo para formalizar la rutina, reforzar asistencia y dejar evidencia temporal. En la práctica, esto redujo fricción de coordinación y facilitó que el equipo trabajara con un ritmo predecible.

Herramienta	Uso principal	Aporte a trazabilidad/valor
Jira	Backlog, sprints, tickets, estados	Control de alcance y visibilidad diaria del progreso
Google Calendar	Agenda de dailies y reuniones día por medio	Disciplina de cadencia y coordinación
GitHub	Repositorio, branches, PR	Gobernanza de integración y control de cambios
revisión de código por branch	Verificación previa a merge	Reducción de defectos y regresiones

Tabla 16: Herramientas de gestión y su propósito
Fuente: Elaboracion Propia.

El control de calidad no se trató como una fase separada, sino como una práctica integrada al flujo normal de trabajo. En ese contexto, el rol de la gestión del producto se concentró en establecer expectativas mínimas de entrega (criterios de aceptación y un umbral operativo de “done”) y asegurar que estuvieran visibles para el equipo en Jira y en la conversación del *Pull Request*.

Cada funcionalidad se implementó en una rama por *feature*. La integración se realizó mediante *Pull Requests*, revisados por pares antes del *merge*. La revisión no se limitó al estilo: también consideró coherencia con la lógica de negocio acordada, claridad del flujo de usuario y mantenibilidad del código. Cuando un cambio no cumplía lo acordado, se dejaban comentarios en el PR y la integración se postergaba hasta corregirlo.

Este mecanismo permitió dos efectos relevantes para el proyecto: (i) detectar defectos temprano y reducir retrabajo, y (ii) distribuir conocimiento técnico, ya que el equipo revisaba cambios cruzados y compartía criterios. Desde la gestión, esto ayudó a sostener consistencia entre incrementos sin depender de una sola persona.

4.3 Cadencia: Dailies día por medio (Calendar)

Para sostener coordinación estable, se calendarizaron dailies y reuniones asociadas al sprint día por medio. Las dailies (Reuniones) se orientaron a objetivos concretos (qué mover hoy para acercarse al objetivo de sprint), mientras que las reuniones intermedias permitieron revisión de avance, detección de bloqueos y ajustes de prioridad.

La planificación de sprints se apoyó en la carta Gantt y en la planificación detallada del proyecto (Artefacto 9), donde se definieron hitos, objetivos por etapa y validaciones asociadas. El rol de la gestión del producto fue mantener coherencia entre esos hitos y el avance real, ajustando alcance cuando la evidencia indicaba que una funcionalidad debía simplificarse o postergarse.

Para aterrizar el plan, se trabajó con incrementos pequeños: funcionalidades mínimas que pudieran demostrarse y evaluarse. Esta forma de trabajo fue particularmente relevante tras la reformulación: el proyecto dejó de ser una idea amplia y pasó a construirse como una secuencia de entregas verificables, con prioridad en descubrimiento de eventos y claridad del flujo.

Las actividades de validación se calendarizaron como trabajo del proyecto, no como “algo extra”. En la etapa 1 se realizaron entrevistas y observación en terreno, y en la etapa 2 se aplicó un test de usabilidad con pauta y notas. En ambos casos, el resultado se tradujo en ajustes de backlog.

Instancia	Frecuencia	Objetivo	Salida esperada
Daily	Diaria	Visibilidad de avance y bloqueos; objetivo del día	Tareas priorizadas y bloqueos asignados
Reunión sprint-relacionada	Día por medio	Revisar avance vs sprint goal; reordenar	Ajuste de backlog y compromisos
Revisión de entregas	Según necesidad	Evaluar historias cercanas a Done	Aceptación o correcciones
Cierre de sprint (si aplica)	Fin de sprint	Evaluación de objetivo y aprendizajes	Acciones de mejora e iteraciones

Tabla 17: Cadencia de coordinación utilizada.

Fuente: Elaboración Propia.

La cadencia de reuniones se estructuró para sostener claridad operativa y evitar pérdida de sincronización. Se realizaron *dailies* con objetivos concretos (qué se avanzó, qué se haría y qué bloqueos existían), y además reuniones día por medio orientadas a coordinación del *sprint*, revisión de dependencias y definición de próximos pasos. Este esquema respondió a la realidad académica del equipo: una *daily* breve permitía mantener visibilidad, mientras que la reunión extendida permitía resolver decisiones que requerían más análisis y acuerdos.

Google Calendar se utilizó como herramienta de coordinación central, permitiendo establecer horarios estables y recordatorios. La utilidad práctica no fue “tener un calendario”, sino reducir fricción de coordinación: al existir múltiples compromisos externos, la agenda compartida evitó que la gestión del proyecto dependiera de mensajes dispersos. Adicionalmente, la evidencia de reuniones (convocatorias, fechas y objetivos) facilitó reconstruir la trazabilidad temporal de decisiones, aspecto relevante en una memoria académica.

En términos de control, cada reunión se vinculó a un objetivo de *sprint* o a un hito de validación. Por ejemplo, se definieron reuniones previas a salidas a terreno para preparar guiones de entrevista, y reuniones posteriores para sintetizar hallazgos y traducirlos a historias de usuario. Esta disciplina conectó el trabajo organizacional con el producto: las reuniones se entendieron como un mecanismo para reducir incertidumbre y mantener consistencia en el *backlog*.

REFORMULACIÓN Y GESTIÓN DE PRODUCTO DE UNA PLATAFORMA GEOLOCALIZADA DE EVENTOS PARA ARTISTAS EMERGENTES: CASO BEATMAP

Tipo	Actividad / Reunión	Fecha inicio	Fecha término	Descripción
Reunión	Beatmap reunión	03-09-2025	03-09-2025	Reunión inicial de coordinación
Reunión	Beatmap reunión	05-09-2025	05-09-2025	Planificación inicial
Sprint	Sprint 3 – Prototipo 3	29-08-2025	13-09-2025	Auth, Perfil, CRUD, Modelo de negocios v1
Reunión	Beatmap reunión	08-09-2025	08-09-2025	Seguimiento sprint 3
Hito académico	Exposición BMC + Prototipo	08-09-2025	08-09-2025	Presentación y entrega
Reunión	Beatmap reunión	10-09-2025	10-09-2025	Ajustes post presentación
Reunión	Beatmap reunión	12-09-2025	12-09-2025	Seguimiento técnico
Sprint	Sprint 4 – Validación clientes	13-09-2025	24-10-2025	CRUD, Tracker, Chat, Feedback
Reunión	Beatmap reunión	15-09-2025	15-09-2025	Plan validación
Reunión	Beatmap reunión	17-09-2025	17-09-2025	Seguimiento
Reunión	Beatmap reunión	22-09-2025	22-09-2025	Revisión avances
Reunión	Beatmap reunión	24-09-2025	24-09-2025	Iteración
Reunión	Beatmap reunión	26-09-2025	26-09-2025	Validación funcionalidades
Hito técnico	Envío informe AWS	30-09-2025	30-09-2025	Entrega backend
Reunión	Beatmap reunión	01-10-2025	01-10-2025	Ajustes
Reunión	Beatmap reunión	03-10-2025	03-10-2025	Seguimiento
Reunión	Beatmap reunión	06-10-2025	06-10-2025	Iteración
Reunión	Beatmap reunión	08-10-2025	08-10-2025	Validación
Reunión	Beatmap reunión	10-10-2025	10-10-2025	Revisión
Reunión	Beatmap reunión	13-10-2025	13-10-2025	Ajustes
Reunión	Beatmap reunión	15-10-2025	15-10-2025	Seguimiento
Reunión	Beatmap reunión	17-10-2025	17-10-2025	Control
Reunión	Beatmap reunión	20-10-2025	20-10-2025	Pre cierre
Sprint	Sprint 5 – Prototipo 4	24-10-2025	04-11-2025	Modelo negocios v2 y métricas
Reunión	Beatmap reunión	24-10-2025	24-10-2025	Inicio sprint 5
Reunión	Beatmap reunión	27-10-2025	27-10-2025	Seguimiento
Reunión	Beatmap reunión	29-10-2025	29-10-2025	Iteración
Reunión	Beatmap reunión	31-10-2025	31-10-2025	Validación
Sprint	Sprint 6 – Cierre	04-11-2025	11-11-2025	Pruebas finales y bugs
Reunión	Beatmap reunión	05-11-2025	05-11-2025	Revisión final
Reunión	Beatmap reunión	07-11-2025	07-11-2025	Ajustes finales
Reunión	Beatmap reunión final	10-11-2025	10-11-2025	Cierre del proyecto

Figura 3: Extracto de calendario de reuniones y hitos del proyecto (reuniones día por medio y validaciones).

Fuente: Documentación Beatmap

4.4 Control de calidad mediante revisión de código

El control de calidad se abordó como una práctica transversal integrada al flujo del equipo. En lugar de una etapa separada, se priorizó la detección temprana mediante revisiones entre pares, criterios de aceptación y consistencia con el incremento del *sprint*.



Figura 4: Flujo de trabajo de integración continua (Branching Model).
Fuente: Medium

Criterio	Descripción operativa
Revisión de PR	PR revisado y aprobado por al menos un integrante distinto al autor
Happy path	Flujo principal afectado ejecutado sin fallas visibles
No regresión	Rutas críticas previas mantienen funcionamiento
Coherencia con criterios	Cumple criterios de aceptación del ticket
revisión de código por branch	Verificación mínima realizada antes de merge
Documentación mínima	Descripción de PR y vínculo al ticket cuando aplica

Tabla 18: *Definition of Done (DoD)* técnico-funcional
Fuente: Elaboración propia.

4.5 Planificación de sprints y validaciones planificadas (Artefacto 9)

La planificación de sprints consideró tanto desarrollo como validaciones. Se programaron hitos de validación con usuarios y *stakeholders*, incluyendo una fase de validación con clientes (organizadores y artistas) y una fase de consolidación para demostración pública.

Esta planificación refuerza la integración entre entrega y validación, coherente con Lean Startup.

4.6 Acta de constitución (Project Charter) y alcance operativo

Desde la gestión de producto, la gestión del producto consolidó un acta de constitución operativa para alinear al equipo: propósito del producto, problema a resolver, usuarios objetivo, restricciones de tiempo (MVP primer semestre 2025) y criterios mínimos de éxito. Este documento se utilizó como referencia para decidir cuándo aparecían nuevas ideas o solicitudes que amenazaban el foco.

El alcance se definió por etapas: primero, habilitar descubrimiento y publicación de eventos con geolocalización; luego, mejorar filtros, consistencia y experiencia; y finalmente, dejar preparado el terreno para módulos avanzados. Esta separación permitió hacer el proyecto realizable y evaluable dentro del contexto académico.

4.7 Plan de comunicaciones, reuniones y coordinación

La coordinación se gestionó como un sistema. La gestión del producto actuó como enlace principal entre el equipo y los frentes del proyecto (validación, avances técnicos, difusión, coordinación académica). Se definió una cadencia: *dailies* con objetivos concretos, reuniones día por medio para seguimiento del sprint y resolución de bloqueos, y sesiones de revisión para cerrar incrementos.

Google Calendar se utilizó para calendarizar reuniones, hitos y validaciones. Con esto se evitó depender de coordinación informal y se hizo visible el costo de cada actividad. Cuando una reunión no aportaba al objetivo del sprint, se ajustaba o se eliminaba.

4.8 Gestión de riesgos y planes de contingencia (B/C/D)

La gestión de riesgos se basó en identificar amenazas frecuentes en proyectos con incertidumbre: dispersión del alcance, bloqueos técnicos, baja disponibilidad de participantes para validación y retrasos por carga académica. La gestión del producto mantuvo un registro de riesgos y definió planes alternativos (Plan B/C/D) para evitar desviaciones que comprometieran la entrega.

Por ejemplo, si una funcionalidad crítica se atrasaba, se definía un recorte funcional para mantener el flujo principal; si la validación presencial no lograba participantes, se activaba reclutamiento por redes y contactos; si el equipo se saturaba, se reordenaba el *backlog* priorizando lo demostrable por sobre lo deseable.

4.9 Control de calidad por revisión entre pares y trazabilidad de decisiones

El control de calidad se abordó mediante revisión de código entre pares, sin una etapa separada de QA. Como *Product Manager*, la gestión del producto definió expectativas mínimas por incremento (criterios de aceptación y un umbral de 'done') y verificó que se reflejaran en Jira y en los *Pull Requests* como guía común de revisión. Esto permitió vincular requerimientos con cambios integrados.

En la práctica, cada *feature* se desarrolló en una rama dedicada y se integró mediante *Pull Requests* revisados por el equipo. Cuando un cambio no cumplía criterios (coherencia con lógica de negocio, claridad del flujo, mantenibilidad), se dejaban comentarios y la integración se postergaba hasta corregir. Este mecanismo redujo retrabajo, fortaleció aprendizaje colectivo y mantuvo consistencia del producto.

4.10 Gestión administrativa: recursos, responsabilidades y seguimiento

El proyecto no operó con un presupuesto monetario formal, pero sí con recursos críticos: tiempo, coordinación y talento. La gestión del producto gestionó estos recursos como restricciones reales, asignando responsables por módulo y promoviendo apoyo cruzado. La delegación se sostuvo con una matriz simple de responsabilidades: cada tarea debía tener dueño, criterio de término y dependencia identificada, evitando tareas sin responsable y mejorando la coordinación.

En términos operativos, la gestión administrativa se abordó como una función transversal que permitió sostener la continuidad del trabajo. En un proyecto académico sin financiamiento dedicado, la disponibilidad de tiempo, el costo de traslados a terreno y la necesidad de producir material audiovisual y gráfico obligaron a priorizar con disciplina. Por lo mismo, la coordinación no se limitó a “asignar tareas”, sino que se orientó a asegurar que cada esfuerzo estuviera alineado con el objetivo del sprint y con evidencias de validación.

Para describir el rol asumido desde la gestión del producto en este contexto, se pueden distinguir responsabilidades que, aunque suelen presentarse por separado en literatura de proyectos, se ejecutaron de manera integrada: planificación, ejecución y liderazgo, gestión de riesgos, control y seguimiento, gestión de recursos, comunicación y cierre. La planificación incluyó delimitar el alcance por iteraciones, establecer metas por sprint y traducir objetivos de producto en trabajo abordable por el equipo (historias de usuario, criterios de aceptación y definición de “hecho”).

En ejecución y liderazgo, la coordinación buscó mantener un ritmo sostenible: reuniones breves para seguimiento, sesiones día por medio para desbloques y acuerdos, y revisiones orientadas a demostrar incrementos funcionales. En la práctica, Google Calendar se utilizó para consolidar la agenda común y reducir pérdidas de sincronización; esto fue relevante porque el equipo debía compatibilizar estudios, responsabilidades personales y, además, la participación en actividades en terreno.

En gestión de riesgos, se trabajó con un enfoque preventivo: cuando se identificaba una incertidumbre alta (por ejemplo, dependencia de un tercero, cambios de alcance, o retrasos en un módulo crítico), se registraban alternativas de respuesta (Plan B, C y D). Este mecanismo permitió reducir el impacto de bloqueos, sostener el avance hacia hitos y evitar que el proyecto quedara detenido por una sola decisión técnica o de diseño.

En control y seguimiento, Jira permitió mantener trazabilidad entre objetivos del sprint, tareas y cambios integrados. Las decisiones de incluir o excluir funcionalidades se

justificaron en función de evidencia (validación en terreno, observaciones de pruebas, coherencia con el MVP) y en función del costo de oportunidad de desarrollar módulos que no fortalecieran el núcleo del producto. Este criterio fue clave durante la reformulación: el objetivo no era “hacer más”, sino “hacer lo correcto”.

Finalmente, el cierre de cada ciclo se materializó en incrementos demostrables: una versión que podía presentarse y explicar con claridad qué problema resolvía, para quién y cómo se verificó. Este cierre por incrementos permitió, además, comunicar avances a actores externos (personas entrevistadas, comunidad en redes y contextos académicos), manteniendo una narrativa consistente entre negocio, usuarios y tecnología.

4.11 Difusión en redes y señal de comunidad

Además del desarrollo, BeatMap se difundió en redes sociales (Instagram y LinkedIn) para atraer comunidad y visibilizar avances. La difusión se utilizó como parte de la validación: permitió reclutar participantes, contrastar lenguaje y entender qué parte de la propuesta generaba interés real. Estas señales tempranas fueron útiles durante la reformulación, al mostrar qué explicaciones eran comprensibles y qué dudas se repetían.

Además del desarrollo, se ejecutó una estrategia básica de difusión para atraer comunidad y obtener señales tempranas de interés. La difusión se entendió como parte de la validación: no bastaba con construir una aplicación, también era necesario observar si existía disposición de actores reales para seguir el proyecto, interactuar con contenido y reconocer el problema que BeatMap intentaba resolver.

Los canales priorizados fueron Instagram y LinkedIn, debido a su complementariedad. Instagram permitió conectar con público y artistas mediante piezas visuales, publicaciones de avances y registros en terreno; LinkedIn se utilizó para comunicar el proyecto desde una perspectiva de emprendimiento y tecnología, apoyando la narrativa de reformulación y aprendizaje. La selección de canales se justificó por costo: ambos permiten difusión orgánica y registro de interacción sin inversión significativa.

En la práctica, el contenido se concentró en (i) avances del producto (capturas de prototipos y módulos), (ii) registros de asistencia a eventos y entrevistas, y (iii) hitos del proyecto (presentaciones, demostraciones y aprendizajes). Este tipo de contenido buscó evitar el problema común de difundir “una idea” sin evidencia, mostrando progreso verificable y conectando el producto con el contexto musical local.

Se observó que la difusión también funcionó como mecanismo de coordinación interna: al preparar material para publicar, se hacía explícito qué se había logrado, qué faltaba y cuál era el mensaje del sprint. En ese sentido, la difusión operó como una forma de “demo asíncrona”, útil para consolidar la propuesta cuando el tiempo del equipo era limitado.

Canal	Objetivo	Tipo de contenido	Frecuencia (referencial)	Señal / métrica cualitativa
Instagram	Alcance hacia artistas/oyentes; visibilizar eventos.	Reels/fotos de terreno, avances de prototipo, anuncios de demo.	Semanal / por hitos.	Comentarios, mensajes directos, compartidos; menciones de eventos.
LinkedIn	Posicionar el proyecto y registrar aprendizaje.	Publicaciones de reformulación, avances de MVP, presentaciones.	Por hito.	Interacciones de red; contactos y retroalimentación de pares.
Página web (landing)	Punto de referencia y credibilidad.	Descripción del problema, propuesta y acceso a demo.	Cuando hay cambios relevantes.	Visitas y consultas; claridad del mensaje del producto.

Tabla 19: Plan de difusión y señales buscadas.

Fuente: Elaboración propia.

El plan anterior no pretende ser una estrategia de marketing exhaustiva, sino un instrumento de validación y comunicación. La restricción de recursos obligó a mantenerlo simple, priorizando contenido que aportara evidencia o permitiera explicar el producto con claridad. En un escenario con financiamiento, este componente podría expandirse hacia campañas segmentadas y medición cuantitativa, pero no fue un objetivo del Mínimo Producto Viable (MVP).

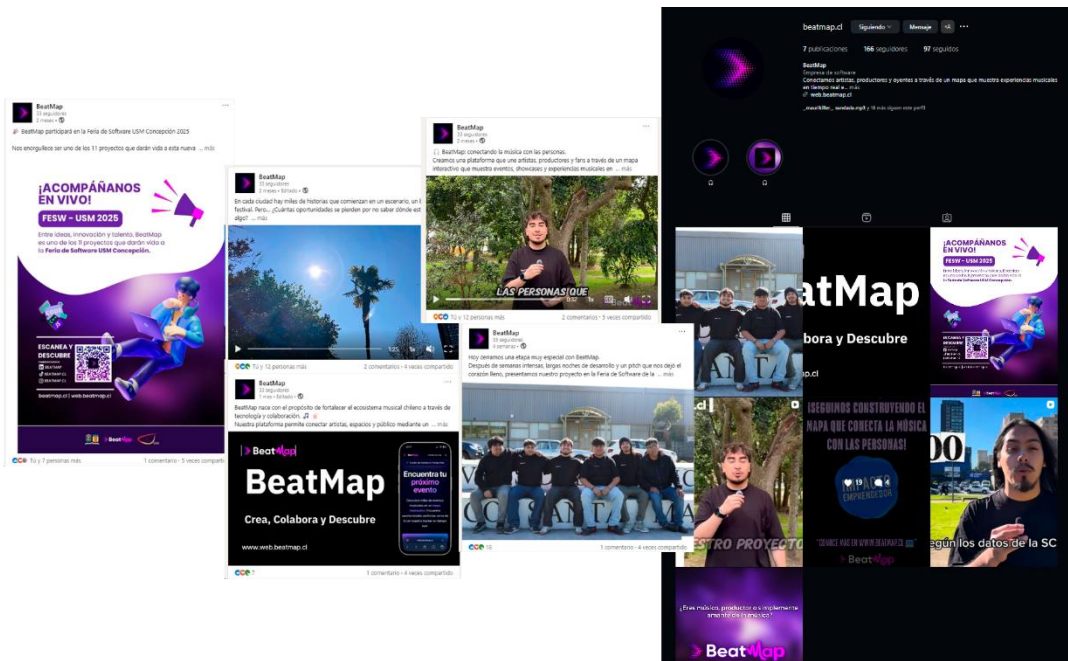


Figura 5: Ejemplos de publicaciones y material audiovisual de difusión (Instagram/LinkedIn) vinculados a hitos del proyecto.

Fuente: elaboración Propia.

4.12 Diseño general de validación (enfoque Lean Startup)

La validación se estructuró en dos etapas complementarias. La primera etapa fue cualitativa, basada en observación participativa y entrevistas en eventos de artistas emergentes, con el objetivo de caracterizar el problema y el comportamiento real del dominio. La segunda etapa fue un test de usabilidad en terreno con oyentes, orientado a medir satisfacción y fricciones de uso del *tracker*.

La lógica Lean Startup se implementó mediante tarjetas de hipótesis y reglas de decisión predefinidas. Esto permitió conectar evidencia con acciones de *backlog* sin depender de intuición o preferencia técnica.

4.13 Modelo de investigación aplicada avanzada (enfoque mixto)

El carácter avanzado se entiende como rigurosidad práctica: preguntas definidas, instrumentos replicables (guía y pauta), registro de notas, síntesis de hallazgos y trazabilidad hacia decisiones de producto (*backlog*). De esta forma, el proceso conectó evidencia - decisión - implementación, justificando cambios de alcance y priorización sin depender de opiniones aisladas.

Para efectos de esta memoria, la validación se trató como investigación aplicada avanzada: un enfoque mixto e iterativo que combina técnicas exploratorias y evaluativas. La etapa 1 (entrevistas y observación) se utilizó para comprender el problema y levantar hipótesis de valor; la etapa 2 (test de usabilidad con pauta) permitió evaluar la propuesta

con tareas y métricas simples. Este esquema se eligió porque el producto se construyó con incertidumbre alta y requería aprendizaje validado antes de consolidar módulos.

4.14 Hipótesis y experimentos

ID	Tipo	Segmento	Hipótesis	Métrica/Umbral	Decisión
H1	Valor	Universitarios y melómanos (iOS/Android)	Usuarios aceptarán y valorarán el tracker al buscar un evento	CSAT 1–5; Pivote si < 3.0	ITERATE (CSAT=3.545)
H2	Canal	Audiencia en redes sociales	Una pauta mínima en IG generará interés medible hacia el tracker	CTR/registro (definido en experimento)	Pendiente/iterable
H3	Técnica	Uso real de servicio	Latencia aceptable para uso sin fricción	p95 < 300 ms	Planificado

Tabla 20: Tarjetas de hipótesis (resumen).

Fuente: Elaboración Propia.

4.14 Etapa 1: entrevistas y observación en terreno

4.14.1 Objetivo, muestra y protocolo de entrevistas

La etapa 1 buscó descubrir el problema real detrás de “no encontrar eventos” y entender cómo se difunden en la práctica. Se realizaron entrevistas semiestructuradas a actores del ecosistema (asistentes, artistas y organizadores), complementadas con observación en terreno durante eventos. La pauta se diseñó para evitar respuestas guiadas y obtener relatos concretos (qué pasó, cuándo, por qué, qué hicieron después).

Como regla, se priorizaron preguntas sobre hechos recientes por sobre opiniones generales. Este enfoque permitió identificar patrones repetidos: uso de Instagram como fuente principal, dependencia de historias efímeras, dificultad para encontrar información de fecha/hora exacta, y falta de un lugar único donde comparar eventos por ubicación.

4.14.2 Principales hallazgos y traducción a módulos

Los hallazgos se tradujeron directamente a módulos del producto. Por ejemplo, la necesidad de “ver qué hay cerca” se convirtió en el *Tracker* (mapa con pines); la necesidad de información verificable se tradujo a un detalle de evento estructurado; y la necesidad de que organizadores pequeños publiquen sin fricción se abordó con un flujo de creación/edición simple.

Problema observado (testimonios)	Consecuencia	Módulo/decisión de producto
Eventos se difunden solo en historias	Asistentes pierden información en horas	Detalle estructurado + guardados
No existe forma simple de ver eventos por zona	Se elige “a ciegas” o por cercanía conocida	Tracker geolocalizado con pines
Información incompleta (hora/lugar)	Baja confianza; baja asistencia	Campos obligatorios + validación de formulario
Organizadores pequeños no usan ticketing	Difusión informal, sin plataforma central	Publicación liviana sin pago como requisito

Tabla 21: Ejemplos de problemas levantados en entrevistas y su traducción a módulos del MVP.

Fuente: elaboración propia.

La pauta de entrevista utilizada y un resumen de categorías se presentan en los anexos, como evidencia del instrumento aplicado.

La observación en terreno permitió identificar prácticas reales de difusión y coordinación. Se detectó que la información sobre eventos se disemina principalmente por redes sociales y mensajería, con baja permanencia y dificultad para búsqueda. En paralelo, la coordinación del evento se sostiene en canales informales, dificultando trazabilidad posterior. Estos hallazgos reforzaron el foco del MVP: el descubrimiento y la reducción de fricción de exploración.

Esta etapa correspondió a entrevistas semiestructuradas y observación en terreno, orientadas a entender cómo se difunden eventos actualmente, qué información resulta crítica para asistir (fecha, lugar, precio, artistas, links) y cuáles son las fricciones más repetidas al buscar panorama. El levantamiento permitió fundamentar la reformulación y priorizar el MVP.

Como salida de producto, los hallazgos se tradujeron en requisitos priorizados: filtros por fecha, presentación clara del detalle y alternativa de exploración mediante lista además del mapa. Esto permite atender la forma real en que los usuarios buscan eventos (rápido, por fecha, y desde móvil).

En esta primera etapa no se buscó “vender” la solución ni validar una interfaz específica, sino comprender rutinas, decisiones y restricciones de quienes participan del circuito musical emergente. Para ello se privilegió una lógica de entrevistas centradas en experiencias recientes, evitando preguntas hipotéticas. Esta aproximación se alineó con el espíritu del Test de Mom: en lugar de solicitar opiniones generales (“¿te gustaría una app así?”), se indagó en hechos (“¿cómo te enteraste del último evento al que fuiste?”; “¿qué pasó la última vez que intentaste postular a un show?”).

El levantamiento en terreno se complementó con observación directa en eventos. La asistencia a presentaciones permitió registrar fricciones operativas que rara vez aparecen en entrevistas abstractas: cambios de horario, dificultades para difundir en tiempo real, dependencia de contactos informales y pérdida de información cuando la difusión se concentra en publicaciones efímeras. Este componente de observación fue especialmente relevante para el diseño del *tracker* geolocalizado, ya que permitió entender qué información resulta útil durante la ejecución del evento y qué tan rápido caduca.

Posteriormente, los hallazgos se organizaron en categorías de necesidad y se transformaron en decisiones de producto mediante un proceso de síntesis. Primero se identificaron patrones repetidos (por ejemplo, problemas para conocer eventos con poca anticipación; dificultad para conectar organizadores con artistas; y ausencia de mecanismos claros para postular). Luego se tradujeron estos patrones a módulos implementables y a casos de uso verificables. Esta traducción permitió pasar de testimonios cualitativos a un *backlog* con elementos observables y medibles.

En términos de ingeniería en informática, este paso de “necesidad” a “módulo” se abordó como una actividad de análisis de requisitos. Se definieron requisitos funcionales y no funcionales por cada módulo priorizado, junto con criterios de aceptación que sirvieron como umbral mínimo de “hecho” para cada incremento. En este punto, el foco fue asegurar consistencia entre problema, hipótesis y funcionalidad: cada historia de usuario debía responder a una fricción real y, al mismo tiempo, ser técnicamente implementable con los recursos disponibles.

Finalmente, la etapa 1 dejó un resultado clave: la reformulación de BeatMap se volvió una decisión sustentada. La evidencia mostraba que el valor del producto no estaba en “acumular servicios”, sino en articular tres actores (artista, organizador y oyente) alrededor de un núcleo de acciones de alta frecuencia: descubrir eventos, publicar/gestionar convocatorias y postular de forma simple. Este resultado definió el alcance del MVP y redujo el riesgo de dispersión.

Es importante considerar que esta validación implicó costos de coordinación y ejecución. La generación de registros, material audiovisual y la asistencia a eventos se realizó sin financiamiento externo, lo que reforzó la necesidad de planificar salidas y entrevistas como experimentos con objetivos claros. Desde la gestión del producto, cada salida a terreno se trató como una inversión de aprendizaje: si no aportaba evidencia nueva o no reducía incertidumbre, no se justificaba repetirla.

4.15 Etapa 2: test de usabilidad con oyentes

4.15.1 Diseño del test y tareas evaluadas

La etapa 2 se enfocó en evaluar usabilidad del prototipo funcional. Se definieron tareas críticas alineadas al MVP: buscar un evento cercano, aplicar un filtro, abrir detalle, interpretar información (fecha/hora/ubicación) y volver al mapa sin perder contexto. Durante el test se registraron notas de observación (errores, dudas, preguntas recurrentes) y una escala de satisfacción general.

En total se reclutaron 22 participantes. De ellos, 20 entregaron una nota global al finalizar el test, obteniéndose un promedio de 6,5 en una escala de 1 a 7. Este resultado se interpretó como señal positiva, pero no como garantía de adopción: la utilidad real depende de contenido (oferta de eventos) y continuidad de difusión.

4.15.2 Resultados y ajustes priorizados

A partir de las notas, se priorizaron ajustes de flujo y claridad. Los cambios más recurrentes se relacionaron con: comprensión de íconos/leyendas del mapa, ubicación del filtro por fecha y dificultad para diferenciar tipos de eventos. Estas observaciones se ingresaron al *backlog* como historias o tareas de mejora, priorizadas por frecuencia e impacto en el flujo principal.

Hallazgo recurrente	Impacto en tarea	Acción en backlog
Confusión con simbología del mapa	Retraso para identificar eventos	Agregar leyenda + tooltip
Filtro por fecha poco visible	No se encuentra evento del día	Reubicar filtro + texto guía
Detalle de evento largo sin jerarquía	Se omite información crítica	Reordenar layout: fecha/hora arriba
Duda sobre confiabilidad de datos	Desconfianza para asistir	Campos obligatorios + moderación

Tabla 22: Resumen de hallazgos del test y acción tomada.

Fuente: elaboración propia.

El protocolo de test y el formato de registro de notas se incluyen en anexos para respaldar el procedimiento aplicado.

La segunda etapa se enfocó en usabilidad: se aplicó un test con pauta y notas para evaluar tareas concretas de descubrimiento. El objetivo fue medir fricción, comprensión del mapa y necesidades de filtrado. Los resultados se consolidaron para traducirlos a decisiones de producto y *backlog*.

Se reclutaron 22 participantes. De ellos, 20 entregaron nota numérica CSAT (1–5) y 2 no registraron calificación. La tarea se definió como: “Encuentra una tocata para este viernes usando el *tracker*”. Se registraron observaciones de uso (zoom, desplazamiento, interacción con pines) y se aplicó una encuesta breve de satisfacción.

N° (Entrevistado)	CSAT
1	4
2	4
3	4.5
4	4.5
5	5
6	5
7	3
8	2
9	4
10	4
11	3
12	3.9
13	2
14	3
15	3
16	4
17	3
18	2
19	4
20	3

Tabla 23: Notas CSAT registradas (n=20)

Fuente: Elaboración propia.

Indicador	Resultado
CSAT promedio	3.545
Satisfechos (4 o 5)	8/20 = 40%
Neutros (3)	45%
Insatisfechos (1-2)	15%
Decisión vs umbral	ITERATE (mejoras UX/UI y estabilidad)

Tabla 24: Síntesis cuantitativa (CSAT)
Fuente: Elaboración propia.

4.16 Hallazgos cualitativos recurrentes del test

Interfaz percibida como “tosca” en móvil; gestos con dos dedos generan fricción.

Solicitan filtros por fecha (día/semana/mes) y lista de eventos bajo el mapa.

Problemas intermitentes en iOS y Android 9 (reinicio/centrado).

La idea de pines en mapa es comprendida y valorada; se solicita mejor diferenciación visual (etiquetas/colores).

4.17 Decisiones de producto derivadas y actualización del backlog

Desde gestión de producto, la conclusión se interpreta como: existe valor percibido, pero la experiencia de uso reduce satisfacción. Por ello se decide ITERATE, priorizando cambios que aumentan control del usuario y reducen carga cognitiva: filtros visibles por fecha y una lista complementaria bajo el mapa. Adicionalmente, se prioriza estabilidad y compatibilidad móvil como requisito de calidad percibida.

Hallazgo	Decisión	Trabajo (épica/tipo)
Fricción de exploración en mapa móvil	Agregar lista bajo mapa	UX/Tracker
Búsqueda por fecha insuficiente	Filtros día/semana/mes	Filtros
Inconsistencias por dispositivo	revisión de código móvil y correcciones	Estabilidad/revisión de código
Necesidad de claridad visual	Etiquetas/colores y jerarquía	UI/Branding

Tabla 25: Hallazgo → Decisión → Trabajo Fuente: Elaboración propia.

4.18 Discusión de resultados y trazabilidad evidencia–decisión

La evidencia recopilada en las dos etapas de validación permitió convertir percepciones cualitativas en decisiones de producto. En la primera etapa, el foco estuvo en comprender la experiencia real de búsqueda y difusión de eventos, identificando patrones de comportamiento y puntos de dolor recurrentes. En la segunda etapa, la atención se trasladó a la usabilidad: si el usuario podía completar tareas clave sin fricción y si el sistema comunicaba su propuesta de valor con claridad.

Desde un punto de vista de ingeniería, la validación se interpretó como un proceso de verificación de hipótesis y de reducción de riesgo. Cada hallazgo relevante se tradujo en ajustes de alcance, cambios de flujo o redefinición de criterios de aceptación. Esta trazabilidad —del hallazgo a la decisión— es especialmente importante en una memoria de Ingeniería en Informática, porque demuestra control del proceso y justificación técnica de la evolución del sistema.

Un aprendizaje relevante fue que la reformulación no representó un fracaso, sino un mecanismo de diseño basado en evidencia. Al iterar la idea, el equipo redujo ambigüedad y delimitó mejor a quién servía el producto y qué problema resolvía. El resultado fue una propuesta más sólida: artistas y organizadores como actores centrales, con asistentes como beneficiarios del descubrimiento simplificado.

Se identificaron oportunidades futuras que no se abordaron por restricción de alcance: automatización de difusión, recomendaciones, analítica y mecanismos de reputación. Estas líneas se proponen como continuidad natural del producto, una vez que el MVP demuestre tracción y se cuente con datos suficientes para orientar decisiones.

Etapas	Objetivo	Método	Participantes / muestra	Evidencia producida	Decisiones derivadas
1 (Exploratoria)	Identificar problemas reales y validar necesidad.	Entrevistas semiestructuradas + observación en terreno (Test de Mom).	Artistas emergentes, organizadores y oyentes (muestra intencional)	Transcripciones, notas de campo y síntesis de hallazgos.	Reformulación hacia tres actores; definición de módulos núcleo y casos de uso.
2 (Evaluativa)	Evaluar comprensión y flujo del prototipo priorizado.	Prueba de usabilidad guiada + cuestionario de percepción (notas).	Oyentes y usuarios potenciales; evaluación por tareas.	Registro de tareas, dificultades, tiempos aproximados y notas.	Ajustes de flujo, priorización de backlog y definición de mejoras para el MVP.

Sprint/Etapa	Objetivo	Alcance esperado
Sprint 4	Validación con clientes (organizadores y artistas)	CRUD, Tracker, Chat y Feedback (según planificación)
Sprint 5	Validación interna de mejoras y métricas clave	Revisión con stakeholders; ajustes
Sprint 6	Consolidación y preparación para presentación	Calidad percibida, estabilidad, demo

Tabla 26: Resumen de validaciones planificadas (según planificación)

Fuente: Elaboración propia.

Sprint / período	Objetivo del sprint	Trabajo principal	Incremento demostrable	Evidencia / resultado
S0 (Inicio)	Alinear problema y definir backlog inicial.	Definición de roles, objetivos, casos de uso preliminares.	Backlog inicial en Jira + calendarización.	Ruta de trabajo y criterios de priorización.
S1	Construir base del tracker y registro de eventos.	Modelado inicial, endpoints base, UI mínima.	Mapa con eventos y estructura de datos.	Demo interna; detección de dispersión funcional.
S2 (Reformulación)	Redefinir propuesta a 3 actores y ajustar alcance.	Síntesis de entrevistas; redefinición de módulos.	Backlog reformulado; wireframes iterados (Excalidraw).	Decisión de pivot; foco en tracker + postulaciones + jobs.
S3	Implementar flujo organizador: publicar evento / job.	CRUD de eventos y convocatorias; permisos.	Organizador publica y gestiona.	Revisión por PR; validación funcional en equipo.
S4	Implementar flujo artista: postular y seguimiento.	Postulación a eventos; notificaciones básicas.	Artista postula a evento; seguimiento de estado.	Coherencia de flujo; ajustes por revisión en equipo.

S5	Mejoras de UX y consistencia de negocio.	Refinamiento de historias; mejoras de navegación.	Versión integrada con mejoras.	Prueba de usabilidad (Etapa 2) y priorización de ajustes.
S6	Preparación de demostración y difusión.	Correcciones; material audiovisual y redes.	Demo estable para presentación.	Difusión en Instagram/LinkedIn; feedback de comunidad.

Tabla 27: Resumen de sprints e incrementos (plantilla).

Fuente: Elaboración propia.

El resumen anterior se presenta como una síntesis, ya que en la práctica existieron ajustes dentro de cada período. A medida que avanzaron los sprints, la priorización se apoyó en dos criterios principales: (i) acercamiento al MVP demostrable dentro del primer semestre y (ii) reducción de incertidumbre mediante evidencia. Esto se tradujo en decisiones explícitas de dejar fuera funcionalidades atractivas pero de bajo impacto inmediato, evitando que el alcance se expandiera por inercia.

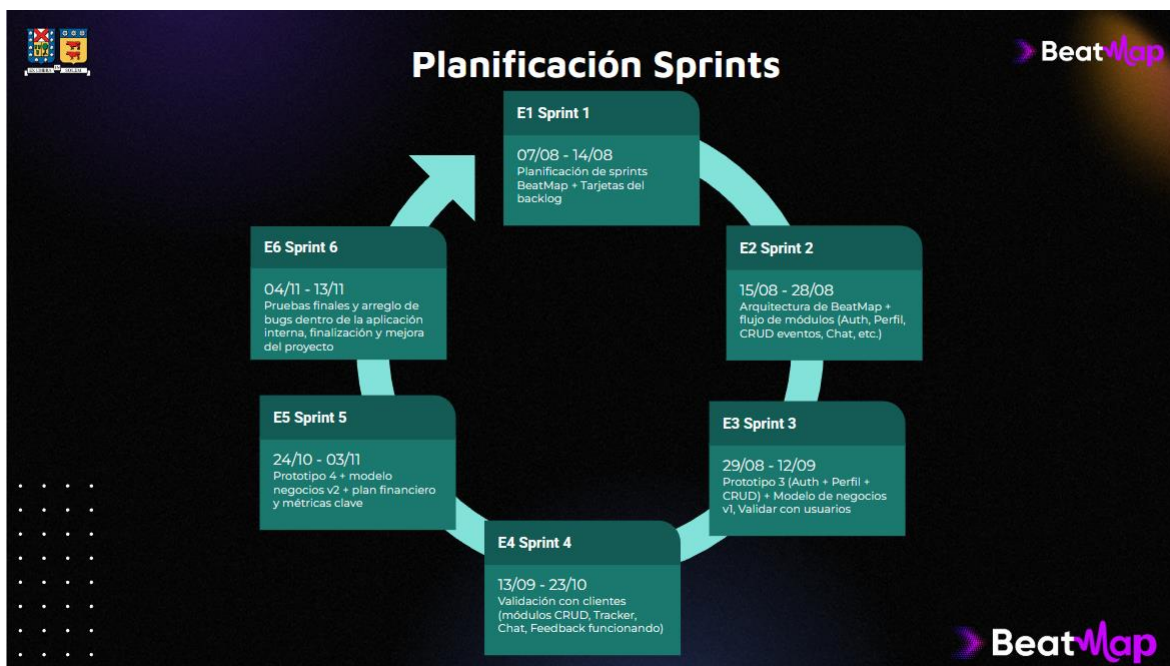


Figura 6: Sprints/hitos (ubicación sugerida para incorporar la visualización temporal del proyecto).

Fuente: Artefacto 9 Documentación Beatmap

En términos de planificación, la gestión del producto mantuvo un equilibrio entre horizonte y flexibilidad. Se definieron hitos relevantes (por ejemplo, la necesidad de

mostrar una versión validada y optimizada para instancias de exposición) y, al mismo tiempo, se mantuvo espacio para cambios cuando la evidencia lo justificaba. Esta combinación resulta coherente con proyectos de software en condiciones de incertidumbre: la planificación se utiliza para ordenar, no para impedir el aprendizaje.

Sprint	Objetivo	Incremento principal
Sprint 1	Aterrizar alcance y backlog inicial	Backlog priorizado, flujos críticos definidos
Sprint 2	Implementar núcleo de descubrimiento	Listado/detalle básico + estructura Tracker
Sprint 3	Ajustes según validación y consistencia	Filtros, mejoras de navegación, estabilización
Sprint 4	Preparación de validación y presentación	Evidencias, instrumentos, backlog actualizado

Tabla 28: Resumen de los sprints relevantes en Beatmap

Fuente: Beatmap

4.18.1 Integración del marco de gestión del proyecto, Lean Startup y Scrum en el ciclo completo

El ciclo completo del proyecto evidencia una integración coherente de marcos. marco de gestión del proyecto aporta estructura de planificación y control (alcance, riesgos, calidad); Lean Startup aporta método de validación y decisión por evidencia; y Scrum aporta cadencia e incrementos cortos. En particular, el uso de Jira y el flujo PR+revisión de código representan controles operativos que materializan gobernanza de calidad, mientras que Calendar sostiene la disciplina de coordinación. El resultado es una ejecución que evita tanto el exceso de documentación predictiva como el caos de un ágil sin control.

La principal lección del proyecto es que, en productos centrados en experiencia móvil, la adopción depende de fricción. Una funcionalidad con valor conceptual puede no alcanzar el umbral de satisfacción si el gesto de exploración falla. Por ello, priorizar UX y estabilidad antes de escalar canales se considera una decisión consistente con Lean Startup y con gestión de riesgos (marco de gestión del proyecto).

4.19 Desempeño del equipo de trabajo, observaciones y aprendizajes colectivos

El desempeño del equipo durante BeatMap estuvo marcado por un aprendizaje progresivo. En la primera iteración, todos compartíamos una visión ideal del producto: una plataforma que concentrara muchos servicios y resolviera, de una sola vez, gran parte de la difusión de eventos. Esa ambición ayudó a visualizar posibilidades, pero también evidenció una debilidad típica de proyectos tempranos: se estaba diseñando desde supuestos internos y no desde necesidades verificadas con usuarios reales.

La primera demostración del prototipo funcionó como un punto de inflexión. El alcance era demasiado amplio y, aunque el sistema parecía completo en términos de funcionalidades, no comunicaba una propuesta de valor clara. Además, se observó una diferencia de abstracción dentro del equipo respecto a qué debía ser BeatMap. Parte de esa brecha se explicaba porque no todos estaban igual de inmersos en la música y en la dinámica de eventos, por lo que no se percibían las mismas fricciones ni prioridades.

Con esa evidencia, se tomó la decisión de reformular la idea. BeatMap cambió de dirección para transformarse en un proyecto más sólido, realizable por etapas y consistente con el tiempo disponible. La propuesta se concentró en conectar actores y reducir fricción en el descubrimiento y la difusión, definiendo como ejes a artistas y organizadores, y como beneficiarios a quienes buscan panoramas. Este cambio permitió delimitar el MVP, recortar alcance sin perder coherencia y organizar el trabajo en incrementos alineados con objetivos de sprint y con criterios de aceptación verificables.

Una parte importante de las dificultades iniciales se relacionó con la inexperiencia y con no pensar suficientemente desde la perspectiva del usuario. Varias decisiones tempranas se tomaron desde una lógica técnica, sin considerar con detalle cómo una persona realmente busca un evento, qué información necesita para decidir asistir o qué le duele a un artista al intentar difundir. Ese desfase se tradujo en hipótesis optimistas y en flujos que requirieron simplificación e iteración.

El cambio más significativo ocurrió cuando la validación pasó de ser un objetivo para convertirse en práctica. En la primera etapa del prototipo 2, el equipo se insertó en eventos y realizó conversaciones con artistas, organizadores y oyentes. En estas instancias se utilizaron preguntas alineadas a la prueba Test de Mom, buscando evitar respuestas complacientes y comprender comportamientos reales: cómo se informan, cómo difunden, qué canales usan y qué frustraciones encuentran. Estas conversaciones permitieron alinear la propuesta de valor con necesidades concretas y convertir hallazgos cualitativos en decisiones de *backlog* y ajustes del MVP.

Durante la primera iteración, el trabajo del equipo estuvo marcado por una visión “ideal” del producto: se asumía que bastaba con reunir múltiples servicios en una misma plataforma para generar valor. Esa suposición fue razonable como punto de partida, pero se evidenció insuficiente cuando se intentó demostrar el producto. La primera demo mostró que el alcance era amplio y que la propuesta no explicaba con claridad qué dolor resolvía primero. En consecuencia, la organización del trabajo enfrentó un desafío no técnico: alcanzar una comprensión compartida del problema antes de profundizar en la solución.

Un factor que influyó en esa primera etapa fue la diferencia de inmersión de los integrantes en el mundo musical. No todos tenían la misma abstracción de las dinámicas de artistas, organizadores y público; por lo tanto, inicialmente se diseñaba desde la lógica de “lo que un equipo técnico cree útil” más que desde la lógica de quien usa la

plataforma. Este punto se convirtió en un aprendizaje transversal: pensar en términos de usuario no es un eslogan, sino un esfuerzo deliberado de observación escucha y síntesis.

Con el inicio de la reformulación, el equipo convergió gradualmente hacia una dirección común. La asistencia a eventos y el trabajo en terreno ayudaron a homogeneizar la comprensión del contexto: al ver cómo se difunden tocatas, cómo se coordinan line-ups y cómo el público se entera de actividades, se redujo la distancia entre la idea conceptual y la realidad operativa. En la práctica, esta inmersión facilitó acuerdos de priorización, porque permitió justificar qué quedaba fuera del MVP y por qué.

En términos de colaboración, la dinámica interna se fortaleció a través de revisiones entre pares y apoyo cruzado. Aunque no existió un rol de QA separado, la calidad se sostuvo mediante *Pull Requests* y comentarios de revisión, lo que promovió aprendizaje compartido. Este mecanismo no solo detectó defectos tempranos, sino que también ayudó a construir criterios comunes sobre coherencia de flujo, consistencia de lógica de negocio y mantenibilidad, aspectos críticos cuando se trabaja con iteraciones rápidas.

Las adversidades no fueron únicamente técnicas. La ausencia de presupuesto impuso restricciones reales: traslados, entradas, producción de piezas gráficas, edición de material audiovisual y actividades de difusión debieron financiarse con recursos propios. En este escenario, la gestión del tiempo y la coordinación fueron tan importantes como la programación. Se priorizaron actividades que generaran evidencia y visibilidad del avance, y se evitó sostener trabajo que no tuviera retorno en aprendizaje o en consolidación del núcleo del producto.

Finalmente, el principal indicador de desempeño colectivo fue la capacidad de cambiar sin fracturarse. La reformulación implicó reconocer errores iniciales y redefinir prioridades, pero también implicó sostener compromiso en reuniones, cumplir acuerdos y aceptar revisiones. En proyectos académicos con alta carga externa (estudios, evaluaciones, disponibilidad variable), este tipo de cohesión es un resultado por sí mismo: permitió convertir una idea potente pero difusa en un proyecto sólido, por etapas, con una propuesta entendible y validada en terreno.

REFORMULACIÓN Y GESTIÓN DE PRODUCTO DE UNA PLATAFORMA GEOLOCALIZADA DE EVENTOS PARA ARTISTAS EMERGENTES: CASO BEATMAP

Open 0 Closed 21 Author Labels Projects Milestones Assignees Types Newest

- Chore: implementar rabbit mq message broker service en el message controller Back-end Chore Feature BeatMap-music/BeatMap#71 · by tsolarc was closed last week · ✓ 2/2
- feat: cambios en el método de lectura de mensajes Back-end Feature BeatMap-music/BeatMap#69 · by tsolarc was closed on Nov 27, 2025 · ✓ 2/2
- feat: implementar beatbot para la sincronización entre branches Back-end Feature BeatMap-music/BeatMap#55 · by tsolarc was closed on Nov 7, 2025 · Approved · ✗ 1/2
- feat: se agregan nuevos casos de pruebas para validar el funcionamiento de RabbitMQ Back-end Feature BeatMap-music/BeatMap#54 · by tsolarc was closed on Nov 1, 2025 · Approved · ✓ 1/1
- feat: se agregan nuevos casos de prueba para el ChatServiceImpl Back-end Feature BeatMap-music/BeatMap#53 · by tsolarc was closed on Nov 1, 2025 · ✓ 1/1
- feat: se realizan modificaciones de los tests del servicio de mensajería Back-end Feature Refactor BeatMap-music/BeatMap#51 · by tsolarc was closed on Nov 1, 2025 · Approved · ✓ 1/1
- SECURITY: implementar seguridad en el gateway y el backend Back-end Feature Seguridad BeatMap-music/BeatMap#43 · by tsolarc was closed on Oct 14, 2025 · ✗ 0/1
- Feat: Crear scripts para generar release automatizado Back-end Feature BeatMap-music/BeatMap#41 · by tsolarc was closed on Sep 7, 2025 · Approved · ✓ 1/1
- feat: se implementa el MVP con 4 funcionalidades principales del sistema Feature Front-end BeatMap-music/BeatMap#39 · by Nastiano was closed on Jul 7, 2025 · Approved · ✗ 0/1
- feat: se agrega parcialmente funcionalidad user en frontend react Feature Front-end BeatMap-music/BeatMap#35 · by Keyori66 was closed on Nov 9, 2025 · ✓ 1/1
- feat: se integran los archivos de marketplace Feature Front-end Hotfix BeatMap-music/BeatMap#32 · by DismarOz was closed on Nov 9, 2025 · Changes requested · ✗ 0/1
- Feature/pantalla front jobs Feature BeatMap-music/BeatMap#29 · by Keyori66 was closed on Jul 3, 2025 · Approved · ✓ 1/1
- feat: implementar pantallas y funcionalidad desde el backend del modulo de chats Feature Front-end BeatMap-music/BeatMap#28 · by tsolarc was closed on Nov 9, 2025 · ✗ 0/1

Figura 7: Evidencias de trabajo colaborativo: ejemplo de *Pull Request* comentado y ajustes aplicados antes de integrar cambios.
Fuente: Repositorio Github Beatmap

CAPÍTULO 5: CONCLUSIONES

5.1 Conclusión general

En términos generales, BeatMap evidenció que en productos digitales con alta incertidumbre el avance depende menos de “construir mucho” y más de **reducir ambigüedad con evidencia**, sostener **foco de alcance** y mantener **trazabilidad** entre decisiones, *backlog* y entregables. La evolución del proyecto —incluyendo reformulaciones— permitió pasar de una idea amplia a una propuesta más concreta y viable por etapas, ajustada a restricciones reales de tiempo, coordinación y recursos.

5.2 Conclusiones por objetivo específico

Objetivo específico 1: Definir el rol y responsabilidades del Product Manager dentro del proyecto.

Se consolidó un rol de gestión con énfasis administrativo y de coordinación: definir alcance por iteraciones, traducir objetivos a trabajo ejecutable, asignar responsables, sostener seguimiento (daylis y reuniones día por medio) y resguardar coherencia entre lo planificado y lo entregado. El aporte principal fue evitar dispersión y proteger el objetivo del MVP frente a cambios y bloqueos.

Objetivo específico 2: Analizar metodologías ágiles aplicables a la gestión de producto. La experiencia mostró que una aplicación rígida de una única metodología no era suficiente para el contexto académico y de incertidumbre del producto. Se verificó como adecuado un enfoque híbrido: Scrum aportó cadencia (sprints, objetivos acotados e incrementos), Kanban permitió controlar flujo y visualizar trabajo en curso, y Lean Startup ordenó la validación mediante hipótesis, aprendizaje y decisiones de recorte o ajuste. La selección se justificó por necesidad de iterar con rapidez sin perder control: la cadencia sostuvo el ritmo, el flujo evitó saturación operativa y la validación evitó construir sobre supuestos. En consecuencia, la metodología se entendió como un sistema de trabajo adaptable, orientado a sostener foco y evidencia más que ceremonias.

Objetivo específico 3: Definir mecanismos de levantamiento y gestión de requerimientos. Se comprobó que el levantamiento de requerimientos fue más efectivo cuando partía desde hechos observables en terreno (entrevistas y observación) y se transformaba en artefactos accionables: casos de uso, historias de usuario, criterios de aceptación y *backlog* priorizado. Esta forma de trabajo permitió que los requerimientos no quedaran como declaraciones genéricas, sino como descripciones verificables del comportamiento esperado del sistema por actor (oyente, artista y organizador). La gestión de requerimientos se fortaleció al mantener trazabilidad entre hallazgos (problemas reales), traducción a módulos (funcionalidades) y verificación (criterios de aceptación). Como aprendizaje, quedó claro que el valor no está en “documentar mucho”, sino en documentar lo necesario para ejecutar, revisar y validar sin ambigüedad.

Objetivo específico 4: Establecer criterios de priorización del producto. La priorización resultó determinante para transformar una idea inicialmente amplia en un

MVP realizable. Se concluye que el criterio más efectivo combinó: (i) valor directo en el flujo principal (publicación → descubrimiento → decisión), (ii) esfuerzo según capacidad real del equipo, y (iii) riesgo técnico u organizacional. Esta lógica permitió aplicar recortes deliberados y sostener coherencia del MVP, especialmente durante la reformulación. Además, el análisis preliminar de viabilidad (unit economics) aportó un marco de realidad: aunque estimativo, ayudó a entender órdenes de magnitud y a incorporar conceptos como CPC y CAC para dimensionar adquisición y sostenibilidad. La principal conclusión es que la priorización no fue un ejercicio “de gustos”, sino una decisión basada en costo de oportunidad y evidencia disponible.

Objetivo específico 5: Analizar la gestión de *stakeholders* y la comunicación del producto. Se observó que BeatMap dependía de múltiples *stakeholders* con necesidades distintas: comunidad (oyentes), generadores de contenido (artistas), publicadores/gestores (organizadores) y entorno académico. La gestión de producto tuvo que mantener un relato consistente del propósito y, al mismo tiempo, traducir expectativas a entregables alcanzables. La comunicación se sostuvo con reuniones calendarizadas, presentaciones de avance y una estrategia de difusión en redes sociales que funcionó como canal de visibilidad y reclutamiento para validaciones. Se concluye que comunicar avances no fue solo “marketing”: ayudó a clarificar propuesta de valor, detectar dudas recurrentes y reforzar coherencia interna del equipo, especialmente cuando se necesitó alinear comprensión del problema tras la primera demo y la posterior reformulación.

Objetivo específico 6: Identificar herramientas de apoyo al Product Management. Se concluye que el uso disciplinado de herramientas fue un factor de estabilidad operativa. Jira permitió estructurar *backlog*, estados y responsabilidades; Google Calendar sostuvo la cadencia y redujo fricción de coordinación; GitHub (*branches* y *Pull Requests*) ordenó integración y control de cambios. Dado que el acceso a Jira se vio afectado por limitaciones de la cuenta gratuita, quedó como aprendizaje la necesidad de mantener evidencia complementaria (capturas, exportaciones o documentos espejo) para no perder trazabilidad en memoria académica. En conjunto, las herramientas no se utilizaron como fin, sino como soporte para ejecución: sostener ritmo, evitar desalineación y dejar evidencia verificable del proceso.

Objetivo específico 7: Definir mecanismos de validación del producto. La validación se consolidó como el eje que permitió justificar cambios de alcance y confirmar el núcleo del MVP. Se concluye que el esquema en dos etapas fue adecuado: una fase exploratoria (entrevistas/observación) permitió descubrir el problema real (fragmentación de información y fricción de descubrimiento), y una fase evaluativa (test de usabilidad con pauta/notas) permitió detectar fricciones concretas de navegación y priorizar mejoras. Esta validación se integró al trabajo y no se trató como actividad “extra”: se calendarizó, se planificó con instrumentos y se tradujo a *backlog*. En síntesis, el proyecto demostró que la validación, cuando se conecta a decisiones y a entregables, reduce riesgo de construir el producto equivocado y fortalece la coherencia entre evidencia, implementación y resultados.

REFERENCIAS BIBLIOGRÁFICAS

- Project Management Institute. (2021). *A Guide to the Project Management Body of Knowledge (marco de gestión del proyecto® Guide)* (7th ed.). Project Management Institute.
- Ries, E. (2011). *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business.
- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum.org.
- Blank, S. (2013). *The Four Steps to the Epiphany*. K&S Ranch.
- Kniberg, H., & Skarin, M. (2010). *Kanban and Scrum – Making the Most of Both*. C4Media.
- Cohn, M. (2004). *User Stories Applied: For Agile Software Development*. Addison-Wesley.
- Pressman, R. S., & Maxim, B. R. (2015). *Software Engineering: A Practitioner's Approach* (8th ed.). McGraw-Hill Education.
- Fitzpatrick, R. (2013). *The Mom Test: How to Talk to Customers & Learn If Your Business Is a Good Idea When Everyone Is Lying to You*. CreateSpace Independent Publishing Platform.
- ISO/IEC. (2011). *ISO/IEC 25010: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE)*.
- Equipo BeatMap. (2025). *Reformulación BeatMap* [Documento interno del proyecto].
- Equipo BeatMap. (2025). *Artefacto 9: Planificación del proyecto usando Scrum* [Documento interno del proyecto].
- Equipo BeatMap. (2025). *Artefacto 12: Validación con clientes (BeatMap)* [Documento interno del proyecto].
- Equipo BeatMap. (2025). *Guiones de entrevistas y validación en terreno (primera y segunda etapa)* [Documento interno del proyecto].
- Equipo BeatMap. (2025). *Unit Economics – estimaciones preliminares del modelo de negocio BeatMap* [Documento interno del proyecto].
- Equipo BeatMap. (2025). *Product Backlog, Sprint Backlog y registros de seguimiento* [Documentación interna del proyecto].
- Equipo BeatMap. (2025). *Prototipos funcionales y diagramas de sistema (Excalidraw)* [Material interno del proyecto].
- Equipo BeatMap. (2025). *Repositorio de código fuente y control de versiones* [Repositorio privado del proyecto].
- Equipo BeatMap. (2025). *Material gráfico y audiovisual de difusión del proyecto* [Archivo interno del proyecto].

ANEXOS

ANEXO A

- PLANIFICACIÓN DEL PROYECTO USANDO SCRUM

El presente anexo corresponde al **Artefacto 9** del proyecto BeatMap y documenta la planificación y organización del desarrollo utilizando la metodología *Scrum*. Este artefacto fue utilizado como base para la gestión del proyecto, la definición del alcance, la priorización de funcionalidades y el seguimiento del avance durante los distintos *sprints*. Su propósito fue estructurar el trabajo del equipo de forma incremental, permitiendo iterar sobre el producto y adaptarlo en función de los resultados obtenidos durante la validación.

- A.1 Objetivo del anexo

Definir la planificación general del proyecto BeatMap, estableciendo objetivos, roles, *backlog* inicial, criterios de aceptación y una visión general de la ejecución del proyecto, asegurando coherencia entre la gestión del producto y el desarrollo técnico.

- A.2 Roles definidos
- Product Manager*: responsable de la definición del alcance del producto, la priorización del *backlog*, la coordinación del equipo y el seguimiento del avance del proyecto.
- Equipo de desarrollo: encargado de la implementación técnica, revisión de código y apoyo en decisiones de diseño.
- Stakeholders*: artistas, organizadores de eventos y oyentes, considerados como actores clave durante el proceso de validación.
- A.3 Product *Backlog* inicial

ID	Historia de Usuario	Prioridad	Criterio de Aceptación
HU-01	Como artista, quiero crear un perfil para mostrar mi trabajo	Alta	El sistema permite registrar información básica y enlaces
HU-02	Como organizador, quiero publicar eventos	Alta	El evento se visualiza correctamente en el <i>tracker</i>
HU-03	Como artista, quiero postular a eventos	Alta	La postulación queda registrada y visible
HU-04	Como usuario, quiero visualizar eventos cercanos	Media	El sistema filtra eventos por ubicación
HU-05	Como organizador, quiero ofrecer trabajos	Media	Los trabajos pueden ser postulados por artistas

- A.4 Organización de sprints

El proyecto se estructuró en *sprints* de duración fija, con instancias de planificación, seguimiento (*daily*) y revisión. Cada *sprint* tuvo objetivos claros y alineados con el avance del MVP, priorizando funcionalidades críticas para la validación temprana del producto.

- A.5 Definición de terminado (*Definition of Done*)
- Funcionalidad implementada y operativa.
- Código revisado por al menos un integrante del equipo.
- Cumplimiento de los criterios de aceptación definidos.
- Integración correcta en la rama principal del repositorio.

ANEXO B

- VALIDACIÓN CON CLIENTES Y USUARIOS

El presente anexo corresponde al **Artefacto 12** del proyecto BeatMap y documenta el proceso de validación realizado con usuarios reales. Incluye las hipótesis planteadas, los guiones de entrevistas, los experimentos ejecutados y los resultados obtenidos durante las instancias de validación en terreno. Este proceso permitió contrastar las decisiones de producto con necesidades reales de los usuarios.

- B.1 Objetivo del anexo

Validar la propuesta de valor del proyecto BeatMap mediante entrevistas y observaciones en terreno, evitando decisiones basadas exclusivamente en supuestos internos.

- B.2 Etapas de validación

Primera etapa: entrevistas exploratorias

Se realizaron entrevistas semi-estructuradas a artistas emergentes, organizadores de eventos y asistentes, con el objetivo de comprender los métodos actuales de difusión y descubrimiento de eventos, así como las principales problemáticas asociadas.

Segunda etapa: validación en terreno

Se aplicaron entrevistas directamente en eventos reales, utilizando principios del *Test de Mom*, observando el comportamiento de los usuarios y contrastando hipótesis en el contexto real de uso.

- B.3 Hipótesis planteadas

Código	Hipótesis	Resultado
H1	Los artistas tienen dificultades para difundir eventos	Validada
H2	Los organizadores requieren un canal centralizado	Validada
H3	Los usuarios buscan eventos cercanos	Validada

H4	Un marketplace general aporta valor	Rechazada
----	-------------------------------------	-----------

- B.4 Guion base de entrevistas
 - ¿Cómo te enteras actualmente de eventos?
 - ¿Qué dificultades enfrentas al difundir o buscar eventos?
 - ¿Qué tipo de herramienta consideras más útil?
 - ¿Pagarías por mayor visibilidad u oportunidades?
- **B.5 Resultados y aprendizajes**

Los resultados evidenciaron una fragmentación significativa de la información, validando la necesidad de un *tracker* geolocalizado. Asimismo, se identificó que funcionalidades adicionales no aportaban valor en etapas tempranas, lo que condujo a una reformulación del producto.

ANEXO C

- **ESTIMACIÓN PRELIMINAR DE UNIT ECONOMICS**

El presente anexo contiene una estimación preliminar de *Unit Economics* del proyecto BeatMap. Este análisis fue utilizado como referencia durante la planificación y no representa valores definitivos, sino aproximaciones destinadas a evaluar la viabilidad económica del proyecto.

- **C.1 Unidad de análisis**
 - Artista destacado.
 - Organizador que publica eventos.
- **C.2 Supuestos principales**
 - Plataforma en etapa temprana.
 - Crecimiento progresivo de usuarios.
 - Modelo de ingresos basado en suscripciones y destacados.
- **C.3 Estructura de ingresos**

Fuente	Valor estimado (CLP)
Destacado de evento	5.000
Suscripción organizador	15.000 / mes
Suscripción artista	10.000 / mes

- **C.4 Costos operacionales**

Concepto	Costo estimado (CLP)
Hosting y servicios	80.000
Herramientas y dominio	20.000
Difusión y marketing	50.000
Total	150.000

- **C.5 Observaciones**

Los valores presentados corresponden a una etapa exploratoria. No se consideran remuneraciones ni inversión externa. El objetivo fue evaluar sostenibilidad básica y no rentabilidad inmediata.