

# Fase 3: Elaboración del producto BackLog

## 3.1 Generación de epicas

- **1. Gestión de géneros musicales personalizados**
  - **HU:** Como usuario, quiero seleccionar mis géneros musicales favoritos al registrarse, para recibir sugerencias de eventos y canciones alineadas a mis gustos.
  - **Descripción:** Permitir que los usuarios seleccionen y gestionen sus géneros musicales favoritos desde su perfil. Esta información se usará para personalizar eventos sugeridos y recomendaciones de canciones.
  - **Objetivo de negocio:** Aumentar la personalización de la experiencia y el engagement del usuario desde el primer uso.
- **2. Sistema de recomendaciones de canciones por usuarios**
  - **HU:** Como usuario, quiero recomendar canciones para los eventos, para contribuir a la playlist colaborativa del evento.
  - **Descripción:** Habilitar a los usuarios para sugerir canciones dentro del género permitido del evento. Las sugerencias quedarán visibles para otros asistentes y formarán parte de la playlist del evento.
  - **Objetivo de negocio:** Incentivar la participación activa y construir una experiencia colaborativa, generando mayor retención y sentido de comunidad.
- **3. Votación de canciones sugeridas por otros asistentes**
  - **HU:** Como usuario, quiero poder votar canciones recomendadas por otros usuarios para que las más populares suenen durante el evento.
  - **Descripción:** Implementar un sistema de votación para priorizar canciones sugeridas. Las más votadas se destacarán en el evento.
  - **Objetivo de negocio:** Mejorar la interacción entre usuarios, y permitir que los eventos respondan más al gusto colectivo, elevando la satisfacción general.
- **4. Notificaciones y recordatorios de eventos**
  - **HU:** Como usuario, quiero recibir notificaciones de nuevos eventos en base a mis preferencias para así estar al tanto de los eventos de mis lugares favoritos.
  - **Descripción:** Enviar notificaciones automáticas antes del evento, y alertas en caso de cambios. Configurables por el usuario.
  - **Objetivo de negocio:** Reducir el ausentismo, aumentar la asistencia efectiva y mantener el contacto constante con el usuario.
- **5. Visualización del perfil del evento**

- **HU:** Como usuario, quiero ver toda la información del evento (lugar, fecha, género, canciones recomendadas, asistentes), para decidir si me interesa participar.
- **Descripción:** Mostrar a los usuarios toda la información relevante de cada evento, incluyendo género, lugar y canciones propuestas.
- **Objetivo de negocio:** Mejorar la tasa de conversión en inscripciones al brindar claridad y transparencia sobre lo que ofrece cada evento.
- **6. Creación de eventos por parte de organizadores**
  - **HU:** Como organizador, quiero poder crear eventos y definir el género musical, para que los usuarios puedan asistir y sugerir canciones.
  - **Descripción:** Brindar una interfaz para que organizadores creen eventos, definan género musical y gestionen propuestas y votaciones.
  - **Objetivo de negocio:** Escalar la app permitiendo a terceros generar contenido y promover sus propios eventos bajo el ecosistema de la app.
- **7. Integración con Spotify o YouTube para escuchar las recomendaciones**
  - **HU:** Como usuario, quiero escuchar un fragmento o la canción completa desde la app cuando veo las sugerencias, para saber si me gusta.
  - **Descripción:** Permitir escuchar un preview o la canción completa desde la app usando APIs externas (Spotify, YouTube).
  - **Objetivo de negocio:** Aumentar la atraktividad y usabilidad de la app, facilitando la elección y aprobación de canciones sugeridas.
- **8. Sistema de badges o logros según participación**
  - **HU:** Como usuario, quiero recibir medallas o reconocimientos por participar activamente en eventos y sugerencias, para motivarme y ganar visibilidad.
  - **Descripción:** Crear un sistema de gamificación que premie a los usuarios que más participan (sugerencias, votos, asistencia).
  - **Objetivo de negocio:** Incentivar la fidelización de usuarios, el crecimiento de la comunidad y la generación de contenido.
- **9. Recomendación inteligente de eventos según historial**
  - **HU:** Como usuario, quiero recibir recomendaciones automáticas de eventos según los géneros y eventos anteriores en los que participé, para descubrir nuevas propuestas afines.
  - **Descripción:** Mostrar sugerencias automáticas de nuevos eventos basadas en comportamiento anterior (géneros favoritos, asistencia, votos).
  - **Objetivo de negocio:** Mejorar el retorno recurrente de usuarios y ofrecer una experiencia cada vez más relevante y personalizada.
- **10. Definición de Arquitectura Modular y Escalable**
  - **HU:** Como equipo técnico, quiero definir una arquitectura modular y escalable.
  - **Descripción:** Diseñar arquitectura basada en microservicios o separación de capas (backend, frontend, base de datos).
  - **Objetivo de negocio:** Permitir crecimiento sostenido y optimizar la performance futura.

- **11. Implementación de Logging y Monitoreo Centralizado**
  - **HU:** Como equipo técnico, quiero implementar logging y monitoreo centralizado.
  - **Descripción:** Integrar servicios de logs, alertas y métricas
  - **Objetivo de negocio:** Detectar fallas más rápido y optimizar el mantenimiento del sistema.

## 3.2 Refinamiento de épicas en features

- **1. Gestión de géneros musicales personalizados:**
  - Descomponer cada épica en features más manejables:
    - **Selección de géneros al momento de registrarse:** Permitir que al crear la cuenta, el usuario pueda seleccionar uno o varios géneros musicales favoritos.
    - **Edición de géneros desde el perfil:** Permitir que el usuario pueda cambiar sus géneros favoritos en cualquier momento desde su perfil.
    - **Sistema de guardado de géneros asociados al usuario:** Base de datos para almacenar y actualizar los géneros seleccionados por cada usuario.
    - **Personalización de eventos y canciones basados en géneros favoritos:** Algoritmo que priorice mostrar eventos y canciones recomendadas basados en esos géneros.
  - Documentar criterios de aceptación de alto nivel
    - Al registrarse, debe ser obligatorio seleccionar al menos un género.
    - Desde el perfil, el usuario debe poder agregar o quitar géneros cuando quiera.
    - El sistema debe guardar automáticamente los géneros seleccionados por el usuario.
    - Las recomendaciones de eventos y canciones deben basarse en los géneros seleccionados.
  - Priorizar según importancia para el negocio
    - Método MoSCoW: Must have (Debe), Should (Debería), Could (Podría) y Won't (No debe)
      - **Selección de géneros al momento de registrarse** -> Must have
      - **Edición de géneros desde el perfil** -> must have
      - **Sistema de guardado de géneros asociados al usuario** -> must have
      - **Personalización de eventos y canciones basados en géneros favoritos** -> should have
    - Método RICE: Reach (Alcance), Impact (Impacto), Confidence (Confianza), Effort (Esfuerzo)
      - **Selección de géneros al momento de registrarse** -> Reach: 10 // Impact: 9 // confident: 8 // effort: 3 // total:  $(10*9*8)/3 = 240$
      - **Edición de géneros desde el perfil** -> Reach: 8 // Impact: 7 // confident: 8 // effort: 5 // total:  $(8*7*8)/5 = 100,8$
      - **Sistema de guardado de géneros asociados al usuario** -> Reach: 8 // Impact: 9 // confident: 8 // effort: 4 // total:  $(8*9*8)/4 = 144$
      - **Personalización de eventos y canciones basados en géneros favoritos** -> Reach: 9 // Impact: 7 // confident: 9 // effort: 5 // total:  $(9*7*9)/5 = 113,4$
  - Estimación inicial: T-shirt sizing (XS, S, M, L, XL) o Planning Poker (1, 2, 3, 5, 8, 13, 21 y

34)

- Estimar el tamaño relativo de cada feature
  - **Selección de géneros al momento de registrarse** -> S
  - **Edición de géneros desde el perfil** -> M
  - **Sistema de guardado de géneros asociados al usuario** -> M
  - **Personalización de eventos y canciones basados en géneros favoritos** -> L
- Identificar features que requieren investigación técnica (spikes)
  - **Spike 1:** ¿Qué sistema o librería de UI usar para la selección de géneros (checkbox, multiselect, etc.)?
  - **Spike 2:** ¿Qué modelo de base de datos es más eficiente para almacenar géneros múltiples por usuario?
- Documentar supuestos y riesgos potenciales
  - En el word 1

## ● 2. Sistema de recomendaciones de canciones por usuarios

- Descomponer cada épica en features más manejables:
  - **Habilitar sugerencia de canciones:** Permitir que usuarios inscritos en un evento puedan proponer canciones.
  - **Visualización pública de sugerencias:** Mostrar todas las canciones sugeridas por los asistentes en la página del evento.
- Documentar criterios de aceptación de alto nivel
  - Solo usuarios inscritos pueden sugerir. Máximo de sugerencias por usuario configurable. Validar que las canciones sugeridas estén dentro del género definido para el evento.
  - Las sugerencias deben estar visibles para todos los asistentes. Incluir nombre de canción, artista y usuario que la propuso
- Priorizar según importancia para el negocio
  - Método MoSCoW: Must have (Debe), Should (Debería), Could (Podría) y Won't (No debe)
    - **Habilitar sugerencia de canciones** -> Must have
    - **Visualización pública de sugerencias** -> must have
  - Método RICE: Reach (Alcance), Impact (Impacto), Confidence (Confianza), Effort (Esfuerzo)
    - **Habilitar sugerencia de canciones** -> Reach: 7 // Impact: 9 // confident: 8 // effort: 6 // total:  $(7*9*8)/6=84$
    - **Visualización pública de sugerencias** -> Reach: 9 // Impact: 9 // confident: 8 // effort: 4 // total:  $(9*7*8)/4=162$
- Estimación inicial: T-shirt sizing (XS, S, M, L, XL) o Planning Poker (1, 2, 3, 5, 8, 13, 21 y 34)
  - Estimar el tamaño relativo de cada feature
    - **Habilitar sugerencia de canciones** -> L
    - **Visualización pública de sugerencias** -> M
  - Identificar features que requieren investigación técnica (spikes)
    - **Spike 1:** ¿Qué sistema o librería de UI usar para la selección de géneros (checkbox, multiselect, etc.)?
    - **Spike 2:** ¿Qué modelo de base de datos es más eficiente para

almacenar géneros múltiples por usuario?

- Documentar supuestos y riesgos potenciales
  - En el word 1

- **4. Notificaciones y recordatorios de eventos**

- Descomponer cada épica en features más manejables:
  - **Configuración de notificaciones en perfil:** El usuario puede activar o desactivar notificaciones de eventos desde su perfil. Permite elegir qué tipo de notificaciones recibir: recordatorios, cambios, nuevos eventos recomendados.
  - **Notificaciones de recordatorio de evento:** 24-48 horas antes del evento, enviar una notificación recordando fecha, hora y ubicación.
  - **Notificaciones de cambios en eventos:** Notificar a los usuarios si hay cambios en eventos a los que están inscriptos (hora, sede, cancelación).
  - **Notificaciones de nuevos eventos basados en preferencias:** Alertar automáticamente cuando se crea un nuevo evento que coincide con los géneros favoritos y ubicaciones frecuentes del usuario.
- Documentar criterios de aceptación de alto nivel
  - El usuario puede configurar qué notificaciones quiere recibir (obligatorio).
  - El sistema envía recordatorios automáticos antes de cada evento.
  - Si un evento cambia o se cancela, los usuarios inscriptos reciben una alerta inmediata.
  - Nuevos eventos compatibles con los gustos del usuario son notificados automáticamente.
- Priorizar según importancia para el negocio
  - Método MoSCoW: Must have (Debe), Should (Debería), Could (Podría) y Won't (No debe)
    - **Configuración de notificaciones en perfil** -> Must have
    - **Notificaciones de recordatorio de evento** -> should have
    - **Notificaciones de cambios en eventos** -> Must have
    - **Notificaciones de nuevos eventos basados en preferencias** -> Could
  - Método RICE: Reach (Alcance), Impact (Impacto), Confidence (Confianza), Effort (Esfuerzo)
    - **Configuración de notificaciones en perfil** -> Reach: 9 // Impact: 8 // confident: 6 // effort: 3 // total:  $(9*8*6)/3=144$
    - **Notificaciones de recordatorio de evento** -> Reach: 7 // Impact: 9 // confident: 7 // effort: 3 // total:  $(7*9*7)/3=147$

- **Notificaciones de cambios en eventos** -> Reach: 9 // Impact: 9 // confident: 8 // effort: 4 // total:  $(9*9*8)/4=162$
  - **Notificaciones de nuevos eventos basados en preferencias** -> Reach: 7 // Impact: 6 // confident: 8 // effort: 4 // total:  $(7*6*8)/4=84$
- Estimación inicial: T-shirt sizing (XS, S, M, L, XL) o Planning Poker (1, 2, 3, 5, 8, 13, 21 y 34)
  - Estimar el tamaño relativo de cada feature
    - **Configuración de notificaciones en perfil** -> L
    - **Notificaciones de recordatorio de evento** -> S
    - **Notificaciones de cambios en eventos** -> M
    - **Notificaciones de nuevos eventos basados en preferencias** -> M
  - Identificar features que requieren investigación técnica (spikes)
    - **Spike 1:** Integración con servicio de notificaciones.
    - **Spike 2:** Control de configuración de notificaciones.
  - Documentar supuestos y riesgos potenciales
    - En el word 1

## ● 8. Sistema de badges o logros según participación

- Descomponer cada épica en features más manejables:
  - **Definición de categorías de logros:** Definir tipos de badges: asistencia, sugerencias, votos, engagement general. Ej.: "Asistente Experto", "Rey de las Sugerencias", "Votante Activo".
  - **Sistema automático de asignación de badges:** Al alcanzar ciertos hitos (ej: asistir a 5 eventos, sugerir 10 canciones), el sistema asigna automáticamente los badges al perfil del usuario.
  - **Visualización de badges en el perfil:** Mostrar los logros del usuario en su perfil público como parte de su identidad.
- Documentar criterios de aceptación de alto nivel
  - Los badges estarán diferenciados por categorías.
  - El usuario recibe badges automáticos cuando cumple criterios de participación..
  - Los badges son visibles en el perfil público del usuario.
- Priorizar según importancia para el negocio
  - Método MoSCoW: Must have (Debe), Should (Debería), Could (Podría) y Won't (No debe)
    - **Definición de categorías de logros** -> should have
    - **Sistema automático de asignación de badges** -> must have
    - **Visualización de badges en el perfil** -> should have
  - Método RICE: Reach (Alcance), Impact (Impacto), Confidence (Confianza), Effort (Esfuerzo)
    - **Definición de categorías de logros** -> Reach: 8 // Impact: 7 // confident: 9 // effort: 4 // total:  $(8*7*9)/4= 126$
    - **Sistema automático de asignación de badges** -> Reach: 8 // Impact: 7 // confident: 8 // effort: 8 // total:  $(8*7*8)/8=56$
    - **Visualización de badges en el perfil** -> Reach: 8 // Impact: 6 // confident: 8 // effort: 6 // total:  $(8*6*8)/6=64$
- Estimación inicial: T-shirt sizing (XS, S, M, L, XL) o Planning Poker (1, 2, 3, 5, 8, 13, 21 y 34)

- Estimar el tamaño relativo de cada feature
  - **Definición de categorías de logros** -> S
  - **Sistema automático de asignación de badges** -> XL
  - **Visualización de badges en el perfil** -> M
- Identificar features que requieren investigación técnica (spikes)
  - **Spike 1:** Diseño visual de badges: Investigar y definir el estilo visual de los badges (minimalistas, coloridos, realistas, gamificados).
  - **Spike 2:** Definición de hitos y métricas: Validar métricas justas y alcanzables para la entrega de badges.
- Documentar supuestos y riesgos potenciales
  - En el word 1

## ● 10. Definición de Arquitectura Modular y Escalable

- Descomponer cada épica en features más manejables:
  - **Definición de estructura modular inicial:** Establecer la separación clara entre frontend, backend, y base de datos, con comunicación bien definida.
  - **Diseño de integración entre módulos:** Definir mecanismos de comunicación entre servicios (ej: API Gateway, colas de mensajería, Webhooks).
  - **Documentación de la arquitectura:** Crear documentación técnica detallada: diagramas de arquitectura, flujos de datos, contratos de APIs.
  - **Estrategia de escalabilidad horizontal:** Planificar cómo escalar los servicios críticos en función de la carga.
- Documentar criterios de aceptación de alto nivel
  - Se define y documenta una arquitectura inicial modular y escalable.
  - Se identifica la separación de responsabilidades de cada módulo o servicio.
  - Se establecen las bases para el crecimiento: escalabilidad horizontal y modularidad.
  - Se definen lineamientos de buenas prácticas para mantener la arquitectura a largo plazo.
- Priorizar según importancia para el negocio
  - Método MoSCoW: Must have (Debe), Should (Debería), Could (Podría) y Won't (No debe)
    - **Definición de estructura modular inicial** -> must have
    - **Diseño de integración entre módulos** -> must have
    - **Documentación de la arquitectura** -> should have
    - **Estrategia de escalabilidad horizontal** -> should have
  - Método RICE: Reach (Alcance), Impact (Impacto), Confidence (Confianza), Effort (Esfuerzo)
    - **Definición de estructura modular inicial** -> Reach: 9 // Impact: 10 // confident: 9 // effort: 5 // total:  $(9 \times 10 \times 9) / 5 = 162$
    - **Diseño de integración entre módulos** -> Reach: 8 // Impact: 9 // confident: 8 // effort: 4 // total:  $(8 \times 9 \times 8) / 4 = 144$
    - **Documentación de la arquitectura** -> Reach: 6 // Impact: 8 // confident: 7 // effort: 3 // total:  $(8 \times 6 \times 7) / 3 = 112$
    - **Estrategia de escalabilidad horizontal** -> Reach: 7 // Impact: 8 // confident: 7 // effort: 5 // total:  $(8 \times 6 \times 8) / 6 = 78$

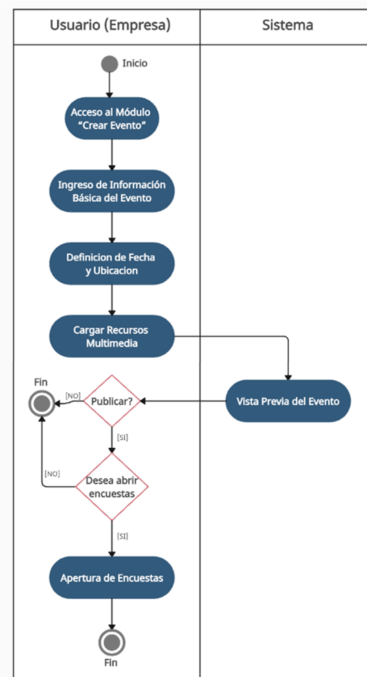
- Estimación inicial: T-shirt sizing (XS, S, M, L, XL) o Planning Poker (1, 2, 3, 5, 8, 13, 21 y 34)
  - Estimar el tamaño relativo de cada feature
    - **Definición de estructura modular inicial**-> M
    - **Diseño de integración entre módulos** -> L
    - **Documentación de la arquitectura** -> S
    - **Estrategia de escalabilidad horizontal** -> M
  - Identificar features que requieren investigación técnica (spikes)
    - **Spike 1:** Análisis de tecnologías a utilizar: Evaluar si conviene usar microservicios desde el principio o diseñar una arquitectura modular monolítica inicial con opción de desacople futuro.
    - **Spike 2:** Selección de herramientas de comunicación entre servicios: Investigar herramientas más adecuadas para orquestar microservicios.
    - **Spike 3:** Costos y complejidad de escalabilidad: Analizar el impacto económico y técnico de escalar la plataforma en infraestructura en la nube.
  - Documentar supuestos y riesgos potenciales
    - En el word 1



### 3.3 Diagrama Actividades y Estados

**Proceso:** Publicación de un Evento

**Objetivo:** Permitir que una empresa/organizador publique un evento para que los usuarios/clientes puedan verlo e interactuar con el

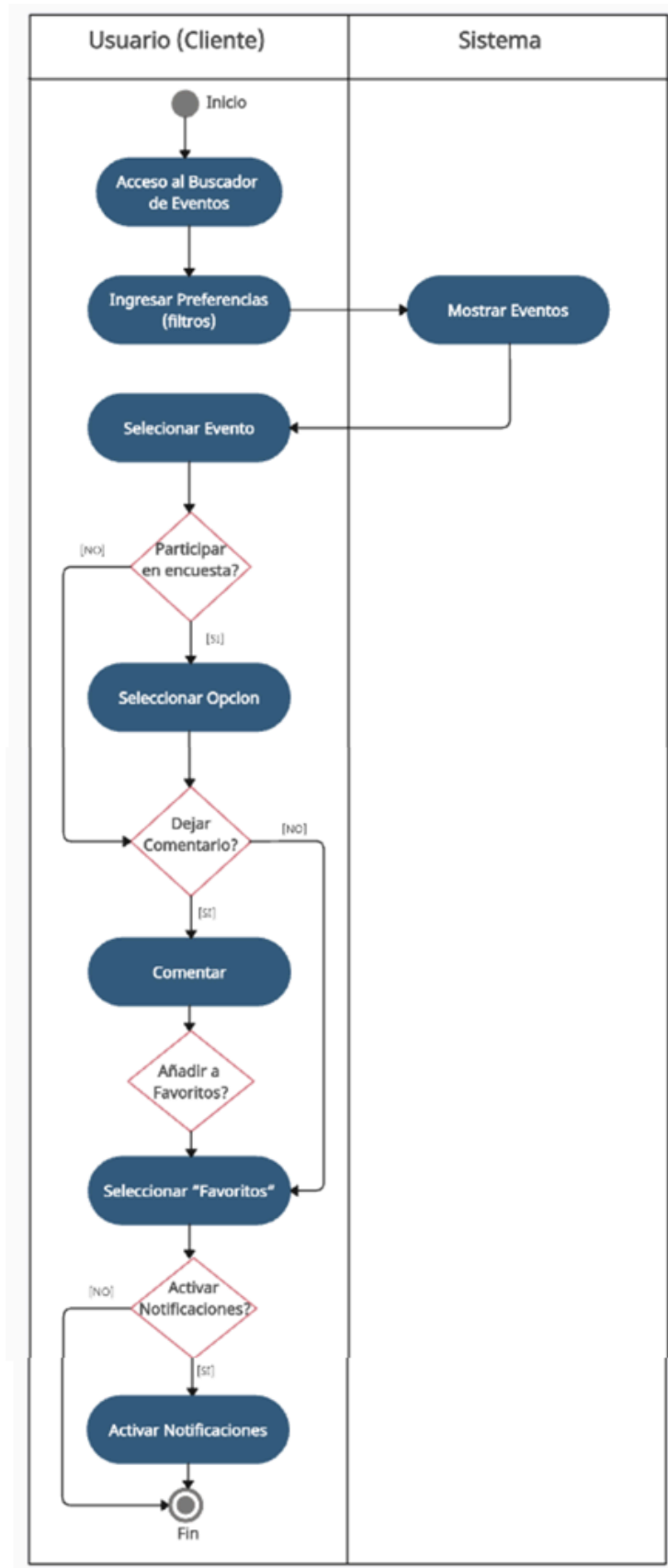


#### Etapas Detalladas (Secuencia de Actividades):

1. **Acceso al Módulo "Crear Evento":** El usuario (empresa) accede a la opción para iniciar la creación de un nuevo evento desde el panel correspondiente.
2. **Ingreso de Información Básica del Evento:** el usuario deberá ingresar el título del evento, una descripción detallada (información que se considere importante).
3. **Definición de Fecha y Ubicación:** El usuario establece la fecha, hora de inicio/fin y la ubicación (física) del evento
4. **Cargar Recursos Multimedia:** Se agregan archivos como imágenes promocionales, videos o documentos relevantes al evento.
5. **Vista Previa:** El sistema genera una previsualización con toda la información ingresada para revisión antes de publicar.
6. **Publicación del Evento:** El usuario decide si desea publicar el evento. En caso afirmativo, el evento se vuelve visible para los clientes.
7. **Apertura de Encuestas (Opcional):** Si el organizador lo desea, puede abrir encuestas para que los usuarios participen en la elección de aspectos del evento como temática, música o preferencias generales.

**Proceso:** Interacción del Usuario con un Event

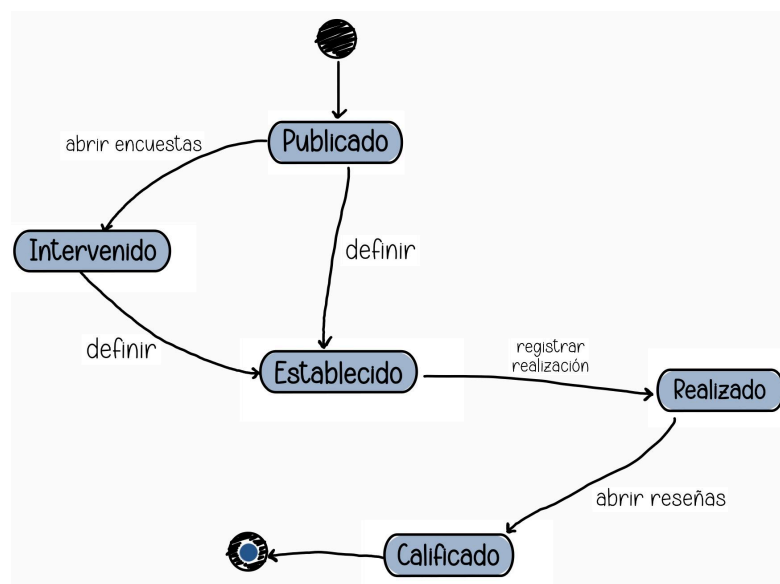
**Objetivo:** Permitir que un cliente explore, seleccione e interactúe con un evento publicado según sus intereses, brindando una experiencia personalizada previa al evento



## Etapas Detalladas (Secuencia de Actividades):

1. **Acceso al Buscador de Eventos:**  
El usuario accede al módulo o sección del sistema destinado a la exploración de eventos disponibles.
2. **Ingreso de Preferencias (Filtros):** el usuario introduce criterios de búsqueda como tipo de evento, ubicación, fecha, temática, etc., para obtener resultados más relevantes.
3. **Mostrar Eventos:** el sistema procesa los filtros y muestra una lista de eventos que coinciden con los criterios seleccionados.
4. **Seleccionar Evento:** el usuario elige un evento específico de la lista para conocer más detalles y opciones de interacción.
5. **Participar en Encuesta:** si el evento tiene encuestas activas, el sistema ofrece al usuario participar. El usuario puede optar por no responder o seleccionar una opción disponible.
6. **Dejar Comentario:** el sistema consulta al usuario si desea dejar una opinión o sugerencia. En caso afirmativo, se habilita un campo para ingresar el comentario.
7. **Añadir a Favoritos:** el usuario puede marcar el evento como favorito para guardarlo y acceder fácilmente más adelante.
8. **Activar Notificaciones:** el sistema ofrece la opción de activar notificaciones para recibir alertas futuras sobre cambios, recordatorios o nuevas interacciones relacionadas con el evento.

## Diagrama de Estado para “Evento”



### Descripción de Estados:

1. **Publicado:** un evento está en “Publicado” cuando el usuario(organizador) crea y publica el evento para que el resto de usuarios puedan visualizarlo.
2. **Intervenido:** un evento está en “Intervenido” cuando el usuario habilita las encuestas para el evento que creó. Permitiendo así que los usuarios puedan interactuar de distintas maneras.
3. **Establecido:** un evento está en “Establecido” cuando el usuario define cuando ya se terminó la interacción con el cliente (pre evento).
4. **Realizado:** el evento se encuentra en “Realizado” cuando ya transcurrió su fecha de realización.
5. **Calificado:** el evento se encuentra en “Calificado” cuando los distintos usuarios que asistieron al evento dejan una reseña/crítica sobre su experiencia.

## 3.4 Historias de Usuario por Feature

### *[HU-001] - Gestión de géneros musicales personalizados*

#### Descripción:

Como usuario,  
quiero seleccionar mis géneros musicales favoritos al registrarme  
para recibir sugerencias de eventos y canciones alineadas a mis gustos.

**Prioridad:** MoSCoW (Must have)

**Estimación:** M (Medium)

#### Criterios de Aceptación:

1. Dado que soy un usuario, cuando registro estoy registrando mi perfil, entonces debo poder seleccionar mis géneros musicales favoritos.
2. Dado que he seleccionado mis géneros musicales, cuando el sistema me muestra eventos recomendados, entonces estos deben estar alineados con los géneros que he elegido.

#### Notas Técnicas:

- Integrar con la base de datos de géneros musicales disponibles.
- Asegurar que los géneros seleccionados por el usuario se guarden en su perfil.
- Crear una API para personalizar las recomendaciones de eventos en base a los géneros seleccionados.

#### Reglas de Negocio Asociadas:

- [ID-RN-004] Los usuarios deben configurar sus preferencias antes de recibir sugerencias personalizadas.

**Riesgos Asociados:**

- [ID-RIESGO-001] El sistema podría no ofrecer recomendaciones precisas si el proceso de selección de géneros presenta dificultades técnicas en base a su desarrollo.

***[HU-002] - Sistema de recomendaciones de canciones por usuarios*****Descripción:**

Como usuario,  
quiero recomendar canciones para los eventos,  
para contribuir a la playlist colaborativa del evento.

**Prioridad:** MoSCoW (Should have)

**Estimación:** L (Large)

**Criterios de Aceptación:**

1. Dado que soy un usuario registrado, cuando entro en un evento, entonces debo ver un campo para sugerir canciones dentro del género permitido.
2. Dado que he sugerido una canción, cuando otros usuarios vean la playlist del evento, entonces mi sugerencia debe estar visible.

**Notas Técnicas:**

- Crear una interfaz de usuario para que los usuarios puedan enviar sus canciones recomendadas.
- Integrar con APIs de música para mostrar las canciones disponibles para sugerir.

**Reglas de Negocio Asociadas:**

- [ID-RN-004] Los usuarios deben configurar sus preferencias antes de recibir sugerencias personalizadas.

**Riesgos Asociados:**

- [ID-RIESGO-002] La interfaz de sugerencias puede no ser atractiva o ser poco intuitiva si no está bien diseñada.

***[HU-003] - Votación de canciones sugeridas por otros usuarios*****Descripción:**

Como usuario,  
quiero poder votar canciones recomendadas por otros usuarios,  
para que las más populares suenen durante el evento.

**Prioridad:** MoSCoW (Must have)

**Estimación:** M (Medium)

**Criterios de Aceptación:**

1. Dado que soy un usuario registrado, cuando veo una lista de canciones sugeridas en el evento, entonces debo poder votar por mis canciones favoritas.
2. Dado que he votado, cuando se muestre la lista final de canciones, entonces las más votadas deben ser destacadas.

**Notas Técnicas:**

- Implementar un sistema de votación con un límite de votos por usuario.
- Mostrar en tiempo real el ranking de las canciones más votadas.

**Reglas de Negocio Asociadas:**

- [ID-RN-004] Los usuarios deben configurar sus preferencias antes de recibir sugerencias personalizadas.
- [ID-RN-003] Solo usuarios autenticados pueden votar, comentar o marcar eventos como favoritos
- [ID-RN-005] Las empresas no podrán modificar votaciones ni comentarios de usuarios.

**Riesgos Asociados:**

- [ID-RIESGO-003] Los votos pueden ser manipulados si no se implementa una validación adecuada.

***[HU-004] - Notificaciones y recordatorios de eventos***

**Descripción:**

Como usuario,  
quiero recibir notificaciones de nuevos eventos en base a mis preferencias,  
para estar al tanto de los eventos de mis lugares favoritos.

**Prioridad:** MoSCoW (Must have)

**Estimación:** S (Small)

**Criterios de Aceptación:**

1. Dado que soy un usuario registrado y ya configuré mis preferencias, cuando se publique un evento que se alinee con mis gustos, entonces debo recibir una notificación.
2. Dado que un evento se acerca, cuando la fecha esté próxima, entonces debo recibir un recordatorio.

**Notas Técnicas:**

- Implementar un sistema de notificaciones permitiendo al usuario gestionarlas.

**Reglas de Negocio Asociadas:**

- [ID-RN-006] El sistema debe ser utilizado principalmente por navegadores de escritorio.
- [ID-RN-004] Los usuarios deben configurar sus preferencias antes de recibir sugerencias personalizadas.

**Riesgos Asociados:**

- [ID-RIESGO-004] La saturación de notificaciones podría generar molestias.

***[HU-005] - Creación de eventos por parte de empresas organizadoras.***

**Descripción:**

Como empresa organizadora,  
quiero poder crear eventos y definir el género musical,  
para que los usuarios puedan asistir y sugerir canciones.

**Prioridad:** MoSCoW (Must have)

**Estimación:** L (Large)

**Criterios de Aceptación:**

1. Dado que soy una empresa registrada, cuando accedo a la sección de creación de eventos, entonces debo poder definir el género musical, la fecha y la descripción del evento.
2. Dado que he creado un evento, cuando lo publico, entonces los usuarios deben poder ver y sugerir canciones para el evento.

**Notas Técnicas:**

- Crear un formulario para que los organizadores ingresen los detalles del evento.

**Reglas de Negocio Asociadas:**

- [ID-RN-002] El sistema solo permitirá la carga de eventos a cuentas registradas como "empresa".

**Riesgos Asociados:**

- [ID-RIESGO-005] El proceso de creación de eventos podría ser demasiado complejo si no se simplifica.

***[HU-006] - Visualización del perfil del evento***

**Descripción:**

Como usuario,  
quiero ver toda la información del evento,  
para decidir si me interesa participar.

**Prioridad:** MoSCoW (Must have)

**Estimación:** M (Medium)

**Criterios de Aceptación:**

1. Dado que soy un usuario registrado, cuando accedo al perfil de un evento, entonces debo poder ver detalles como lugar, fecha, género musical, canciones sugeridas y asistentes.

**Notas Técnicas:**

- Implementar una base de datos con la información relevante de los eventos.
- Crear una interfaz visual atractiva para mostrar los detalles del evento de manera clara.

**Reglas de Negocio Asociadas:**

- [ID-RN-003] Solo usuarios autenticados pueden votar, comentar o marcar eventos como favoritos.
- [ID-RN-005] Las empresas deben adaptar sus eventos basándose en la retroalimentación del público.

**Riesgos Asociados:**

- [ID-RIESGO-006] La información del evento podría estar desactualizada si no se actualiza en tiempo real.

**Nota:**

**Archivos Adjuntos: (HU-001 hasta HU-006)**

- Diagramas especificados en las fases anteriores: Diagrama contextual, de requerimientos y de clases.

## 3.5 División de historias

**Criterios considerados:**

- Se dividieron las historias **Large** (L) en subhistorias más pequeñas, que representan entre **1 y 3 días de trabajo**.
- Se procuró que cada historia cumpla con los criterios **INVEST**: Independiente, Negociable, Valiosa, Estimable, Small, Testable

**División de historias grandes:**

**[HU-002] - Sistema de recomendaciones de canciones por usuarios**

**Estimación original:** L (Large) → Subdividida en 3 subhistorias



- **[HU-002a] Interfaz para sugerir canciones**  
Como usuario, quiero acceder a un campo para sugerir canciones al ingresar a un evento, para participar en la playlist.  
**Estimación:** S (Small)  
**Dependencia:** ninguna
- **[HU-002b] Validación de género al sugerir**  
Como usuario, quiero que solo pueda sugerir canciones del género permitido del evento, para asegurar que la playlist sea coherente con el estilo del evento.  
**Estimación:** S (Small)  
**Dependencia:** HU-002a
- **[HU-002c] Visualización de sugerencias en playlist**  
Como usuario, quiero ver mi sugerencia reflejada en la playlist del evento, para confirmar que fue registrada correctamente.  
**Estimación:** S (Small)  
**Dependencia:** HU-002a

#### **[HU-005] - Creación de eventos por parte de empresas organizadoras**

**Estimación original:** L (Large) → Subdividida en 3 subhistorias

- **[HU-005a] Formulario de creación de evento**  
Como empresa, quiero poder acceder a un formulario para poder ingresar los datos básicos del evento, como el nombre, la descripción, la fecha y la ubicación.  
**Estimación:** M (Medium)  
**Dependencia:** ninguna
- **[HU-005b] Definición de género musical del evento**  
Como empresa, quiero definir el género musical al crear el evento, para que los usuarios puedan sugerir canciones relevantes para ese género.  
**Estimación:** S (Small)  
**Dependencia:** HU-005a
- **[HU-005c] Confirmación y publicación del evento**  
Como empresa, quiero recibir una confirmación visual al crear el evento, para asegurarse de que se haya publicado correctamente y que los usuarios pueden verlo y sugerir canciones.  
**Estimación:** S (Small)  
**Dependencia:** HU-005a

### **3.6 Subdivisión Técnica**

Equipo Técnico (6 personas):

- Backend: 2 personas
- Frontend Web: 2 personas
- QA (Testing): 1 persona
- Documentación/Coordinación Técnica: 1 persona (apoya donde se necesite)

#### Subtareas por área:

#### **Modelado de Datos (Backend)**

- Definir entidades: Usuario, Evento, Empresa, Comentario, Preferencia
- Diseñar relaciones entre entidades
- Crear la base de datos inicial

#### **Endpoints / API (Backend)**

- POST /registro: registro de usuarios
- POST /login: inicio de sesión
- GET /eventos: listado de eventos
- GET /eventos/{id}: detalle de evento
- POST /eventos: subir nuevo evento (empresa)
- POST /eventos/{id}/comentarios: comentar evento
- POST /eventos/{id}/votar: votar preferencias
- GET /empresa/{id}/estadísticas: ver estadísticas

#### **Interfaz UI/UX (Frontend)**

- Diseño de pantallas: Inicio, Login, Registro, Listado de Eventos, Panel de Empresa
- Componente de eventos con botones de votar y comentar
- Diseño responsive
- Integración con API

#### **Test Unitarios / Funcionales (QA)**

- Validación de formularios
- Testeo de endpoints
- Flujo completo de usuario
- Automatización básica

#### **Documentación Técnica**

- Diagrama de base de datos
- Lista de endpoints con ejemplos
- Manual para correr el proyecto
- Guía de colaboración

## 3.7 Planificación de Capacidad Inicial

### **Velocidad Inicial Estimada**

- Equipo de 6 personas.
- Estimación: 30 puntos por sprint.
- Duración del sprint: 1 semana.

### **Composición del Equipo**

- 2 Backend
- 2 Frontend
- 1 QA Tester
- 1 Coordinador/Documentador

### **Planificación de los Primeros 3 Sprints**

#### **Sprint 1: Fundaciones (Semana 1)**

- Historia 1: Registro/Login de usuarios
- Historia 2: Base de datos y relaciones
- Historia 3: Pantallas de login/registro
- Historia 4: Endpoint de autenticación
- Historia 5: Documentación inicial y pruebas

Objetivo: El usuario puede registrarse e iniciar sesión.

#### **Sprint 2: Interacción del Usuario (Semana 2)**

- Historia 1: Visualizar eventos
- Historia 2: Votar preferencias
- Historia 3: Comentar eventos
- Historia 4: Pruebas funcionales
- Historia 5: Documentación de flujo de usuario

Objetivo: El usuario puede votar y comentar en eventos.

#### **Sprint 3: Panel Empresa y Estadísticas (Semana 3)**

- Historia 1: Publicar evento
- Historia 2: Ver estadísticas del evento
- Historia 3: Diseño de panel de empresa
- Historia 4: Tests de funcionalidad completa
- Historia 5: Ajustes finales y documentación

Objetivo: Las empresas pueden publicar eventos y ver retroalimentación.

### **Métricas de Éxito por Sprint**

- Sprint 1: Registro/login funcional sin errores
- Sprint 2: Eventos visibles, votación y comentarios funcionales
- Sprint 3: Publicación de eventos y visualización de estadísticas