**Guía a Rational Unified Process**

**Proyecto: Desparches**

**Integrantes**: Carlos Daza, Sara Castelblanco, Alison Tinoco, Laura Montaña, Daniela Pérez

**Resumen**

Este documento aplica de manera exhaustiva el Proceso Unificado de Rational (RUP) al proyecto **Desparches**, una plataforma digital innovadora diseñada para conectar a usuarios con eventos y promociones exclusivas en su entorno geográfico inmediato. Se describen detalladamente las cuatro fases fundamentales de RUP: Inicio, Elaboración, Construcción y Transición, junto con los artefactos específicos y flujos de trabajo correspondientes a cada fase. La implementación metodológica busca garantizar un desarrollo estructurado, iterativo e incremental que minimice riesgos y optimice recursos durante todo el ciclo de vida del proyecto.

**1. Introducción**

El Proceso Unificado de Rational (RUP) constituye una metodología de desarrollo de software robusta, iterativa e incremental que proporciona un marco disciplinado para la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Este marco metodológico se estructura en cuatro fases secuenciales (Inicio, Elaboración, Construcción y Transición) y nueve flujos de trabajo que se ejecutan con distinta intensidad a lo largo de las fases.

En el contexto del proyecto **Desparches**, RUP ofrece ventajas significativas frente a metodologías más rígidas o menos estructuradas. La plataforma **Desparches** tiene como objetivo principal facilitar el descubrimiento, acceso y participación en eventos y promociones locales, creando un ecosistema digital que beneficie tanto a usuarios que buscan experiencias de valor como a comercios y organizadores que desean aumentar su visibilidad.

La complejidad inherente al desarrollo de **Desparches** radica en:

* La integración de múltiples fuentes de datos sobre eventos y promociones
* La implementación de sistemas de geolocalización precisos
* El desarrollo de mecanismos de filtrado y recomendación personalizados
* La creación de interfaces intuitivas para diversos perfiles de usuario
* La implementación de medidas de seguridad para transacciones y datos personales

La aplicación de RUP permitirá gestionar esta complejidad a través de un enfoque incremental, donde cada iteración proporcionará un producto parcialmente funcional que se refinará progresivamente hasta alcanzar la versión final para lanzamiento al mercado.

**2. Bases teóricas**

**2.1 Fundamentos de RUP**

El Proceso Unificado de Rational se fundamenta en tres premisas esenciales que resultan particularmente relevantes para el desarrollo de **Desparches**:

1. **Desarrollo dirigido por casos de uso**: Los casos de uso actúan como hilo conductor que guía el proceso de desarrollo, desde la captura de requisitos hasta las pruebas finales. En **Desparches**, los casos de uso fundamentales incluyen la búsqueda geolocalizada de eventos, la visualización de promociones por categorías, la reserva/compra de entradas, la valoración post-evento, y la interacción social con otros usuarios.
2. **Arquitectura centrada en el sistema**: RUP establece una arquitectura robusta en etapas tempranas que permite visualizar el sistema holísticamente. Para **Desparches**, se implementará una arquitectura multicapa con separación clara entre presentación, lógica de negocio y acceso a datos, facilitando la escalabilidad y el mantenimiento futuro.
3. **Desarrollo iterativo e incremental**: El proyecto se divide en múltiples iteraciones cortas que generan incrementos funcionales evaluables. **Desparches** se desarrollará en iteraciones de 2-3 semanas, priorizando funcionalidades según su valor para usuarios y patrocinadores.

**2.2 Características diferenciadoras de RUP aplicadas a Desparches**

* **Adaptabilidad metodológica**: RUP permite ajustar el proceso según la escala y complejidad del proyecto. Para **Desparches**, se implementará una versión adaptada que refuerce la documentación en componentes críticos (seguridad, geolocalización) mientras agiliza procesos en funcionalidades secundarias.
* **Trazabilidad bidireccional**: RUP facilita el seguimiento desde requisitos hasta implementación y viceversa. En **Desparches**, esto resulta crucial para verificar que características como "descubrimiento de eventos cercanos" se implementen correctamente en todos los componentes del sistema.
* **Gestión proactiva de riesgos**: La identificación temprana y monitorización continua de riesgos permite mitigarlos antes de que impacten negativamente. Para **Desparches**, se prestará especial atención a riesgos relacionados con la integración de APIs externas, rendimiento en picos de demanda, y protección de datos de usuarios.
* **Verificación continua de calidad**: Las evaluaciones de calidad se integran en cada fase e iteración. En **Desparches**, se implementarán métricas específicas para usabilidad móvil, precisión de geolocalización y tiempo de respuesta en búsquedas complejas.

**3. Las fases**

**3.1 Inicio**

**3.1.1 Objetivos detallados**

* **Delimitar el alcance del proyecto**: Definir con precisión qué incluirá y qué excluirá **Desparches** en su primera versión. Se establecerá que la plataforma abarcará eventos culturales, gastronómicos, deportivos y promociones comerciales en áreas urbanas específicas, excluyendo inicialmente eventos rurales o de nichos muy especializados.
* **Identificar y priorizar casos de uso críticos**: Determinar qué funcionalidades son esenciales para la viabilidad del proyecto. Para **Desparches**, los casos de uso críticos incluyen:
  + Registro y autenticación de usuarios
  + Búsqueda geolocalizada de eventos
  + Visualización detallada de eventos/promociones
  + Reserva/compra de entradas o cupones
  + Valoración post-experiencia
* **Proponer una arquitectura candidata**: Diseñar una arquitectura preliminar que soporte los escenarios principales. Para **Desparches**, se propone una arquitectura cliente-servidor con aplicaciones nativas para iOS/Android, backend en microservicios (Node.js), base de datos PostgreSQL con extensiones geoespaciales, y servicios AWS para almacenamiento y procesamiento.
* **Estimar recursos necesarios**: Cuantificar los recursos humanos, tecnológicos y financieros requeridos. El proyecto **Desparches** requerirá un equipo multidisciplinar compuesto por:
  + 2 desarrolladores frontend (React Native)
  + 3 desarrolladores backend (Node.js, Python)
  + 1 especialista en UX/UI
  + 1 analista de datos
  + 1 ingeniero DevOps
  + 1 gerente de proyecto

La duración estimada será de 10 meses con un presupuesto aproximado de 150,000 USD.

* **Identificar y evaluar riesgos iniciales**: Catalogar potenciales amenazas al éxito del proyecto. Para **Desparches**, los riesgos principales incluyen:
  + Competencia directa de plataformas consolidadas (Eventbrite, Meetup)
  + Dificultad para conseguir masa crítica de usuarios y proveedores de eventos
  + Precisión insuficiente en algoritmos de geolocalización
  + Problemas de rendimiento en picos de demanda
  + Resistencia de comercios locales a la adopción tecnológica

**3.1.2 Productos (Artefactos) específicos**

* **Documento de visión del negocio**: Contiene la descripción detallada de **Desparches**, incluyendo:
  + Declaración del problema: "Los usuarios urbanos carecen de una plataforma unificada para descubrir experiencias locales relevantes, mientras que comercios y organizadores de eventos tienen dificultades para alcanzar su público objetivo."
  + Posicionamiento del producto: "**Desparches** será la aplicación de referencia para el descubrimiento contextual de experiencias locales, diferenciándose por su enfoque integrador (eventos + promociones) y su sistema avanzado de recomendaciones basado en preferencias y ubicación."
  + Características principales: Geolocalización precisa, filtros avanzados, recomendaciones personalizadas, reservas integradas, sistema de valoraciones, notificaciones contextuales.
  + Restricciones: Dependencia de conexión a internet, cobertura inicial limitada a áreas urbanas principales, necesidad de masa crítica de usuarios y proveedores.
* **Modelo inicial de casos de uso**: Documenta con diagramas UML y descripciones textuales los casos de uso críticos de **Desparches**:
  + CU-001: Registrar usuario (Actor: Usuario no registrado)
  + CU-002: Buscar eventos cercanos (Actor: Usuario registrado)
  + CU-003: Filtrar eventos por categoría (Actor: Usuario registrado)
  + CU-004: Visualizar detalles de evento (Actor: Usuario registrado)
  + CU-005: Reservar/comprar entrada (Actor: Usuario registrado)
  + CU-006: Publicar evento (Actor: Organizador)
  + CU-007: Ofrecer promoción (Actor: Comercio)
* **Especificación adicional de requisitos no funcionales**: Define los atributos de calidad que debe cumplir **Desparches**:
  + Rendimiento: Tiempo de respuesta máximo de 2 segundos para búsquedas geoespaciales; capacidad para soportar 10,000 usuarios concurrentes.
  + Usabilidad: Tiempo de aprendizaje inferior a 5 minutos para usuarios novatos; satisfacción de usuario superior a 4/5 en pruebas de usabilidad.
  + Fiabilidad: Disponibilidad del 99.5%; tiempo medio entre fallos superior a 200 horas.
  + Seguridad: Protección de datos conforme a RGPD; autenticación de doble factor para transacciones; cifrado de datos sensibles.
  + Escalabilidad: Capacidad para escalar horizontalmente según demanda; arquitectura preparada para expansión a nuevas regiones.
* **Glosario técnico y de dominio**: Establece la terminología unificada para el proyecto **Desparches**:
  + Evento: Actividad programada con fecha, hora y ubicación específicas (conciertos, exposiciones, torneos, etc.)
  + Promoción: Oferta comercial con condiciones y periodo de validez definidos.
  + Geofencing: Tecnología que define perímetros virtuales para detectar usuarios en zonas específicas.
  + Experiencia: Cualquier actividad o promoción que el usuario puede disfrutar.
  + Check-in: Proceso de confirmación de asistencia a un evento.
  + Recompensa: Beneficio otorgado a usuarios por su participación o fidelidad.
  + Organizador: Entidad o individuo que crea y gestiona eventos en la plataforma.
  + Comercio: Negocio que ofrece promociones a través de la plataforma.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Flujo de Trabajo** | **Productos Clave** | **Inicio (I)** | **Elaboración (R)** | **Construcción (R)** | **Transición (R)** |
| Administración del Proyecto | Plan de desarrollo | X | X | X | X |
|  | Caso de negocio | X |  |  |  |
|  | Lista de riesgos | X | X | X | X |
| Requisitos | Modelo de casos de uso | X | X |  |  |
|  | Visión | X | X |  |  |
|  | Especificación adicional | X | X |  |  |
|  | Glosario | X | X |  |  |
| Análisis y Diseño | Modelo de diseño | X | X |  |  |
|  | Documentación de la arquitectura SW | X |  |  |  |
| Implementación | Modelo de implementación | X | X | X |  |
| Test | Plan de test | X | X |  |  |
| Despliegue | Plan de despliegue | X |  |  |  |

**Leyenda:**

* **I (Inicio)**: Artefacto iniciado en esta fase
* **R (Refinamiento)**: Artefacto refinado o actualizado en esta fase

**3.2 Elaboración**

**3.2.1 Objetivos detallados**

* **Definir, validar y estabilizar la arquitectura**: Refinar la arquitectura propuesta y comprobar su viabilidad mediante prototipos funcionales. Para **Desparches**, se validará especialmente:
  + La integración entre componentes frontend y backend
  + El rendimiento de consultas geoespaciales
  + La escalabilidad del sistema de notificaciones
  + La consistencia de datos en operaciones concurrentes
* **Mitigar riesgos técnicos significativos**: Abordar sistemáticamente los riesgos más críticos identificados en la fase inicial. Para **Desparches**, se implementarán prototipos específicos para:
  + Evaluar la precisión de geolocalización en entornos urbanos densamente edificados
  + Probar el rendimiento del sistema de búsqueda con volúmenes de datos realistas
  + Verificar la integración con APIs de pago y servicios de mapas
* **Completar el análisis del dominio del problema**: Profundizar en la comprensión del contexto empresarial y técnico. Para **Desparches**, esto incluye:
  + Análisis detallado del comportamiento de usuarios potenciales mediante entrevistas y focus groups
  + Estudio de competidores directos e indirectos
  + Identificación de patrones de uso en aplicaciones similares
  + Mapeo completo de tipos de eventos y promociones a incluir
* **Establecer un plan detallado para la fase de construcción**: Definir iteraciones, asignar recursos y establecer criterios de éxito. El plan para **Desparches** contemplará:
  + 6-8 iteraciones de 3 semanas cada una
  + Desarrollo incremental de funcionalidades priorizadas según valor para el usuario
  + Integración continua con pruebas automatizadas
  + Revisiones de calidad al final de cada iteración

**3.2.2 Productos (Artefactos) específicos**

* **Modelo completo de casos de uso (80-90%)**: Ampliación del modelo inicial para incluir todos los casos de uso relevantes para **Desparches**:
  + Casos de uso base (ya identificados en fase de inicio)
  + Casos de uso adicionales:
    - CU-008: Valorar experiencia realizada
    - CU-009: Compartir evento en redes sociales
    - CU-010: Recibir notificaciones contextuales
    - CU-011: Gestionar perfil y preferencias
    - CU-012: Visualizar historial de actividades
    - CU-013: Obtener recomendaciones personalizadas
    - CU-014: Guardar eventos/promociones favoritos
    - CU-015: Gestionar métodos de pago
    - CU-016: Reportar contenido inapropiado
    - CU-017: Contactar con soporte técnico
* **Descripción arquitectónica detallada**: Documentación exhaustiva de la arquitectura de **Desparches** con múltiples vistas:
  + **Vista lógica**: Diagramas de clases UML que definen:
    - Modelo de dominio (Usuario, Evento, Promoción, Categoría, Valoración, etc.)
    - Servicios de aplicación (GeolocationService, RecommendationService, NotificationService, etc.)
    - Componentes de infraestructura (Authentication, Persistence, Caching, Logging, etc.)
  + **Vista de proceso**: Diagramas de actividad y secuencia para operaciones críticas:
    - Flujo de descubrimiento y reserva de eventos
    - Proceso de publicación y aprobación de eventos
    - Mecanismo de recomendaciones personalizadas
  + **Vista de implementación**: Organización del código en paquetes y componentes:
    - Estructura de la aplicación móvil (React Native)
    - Organización de microservicios backend
    - Dependencias entre componentes
  + **Vista de despliegue**: Infraestructura técnica para entornos de desarrollo, pruebas y producción:
    - Configuración de servidores y balanceadores de carga
    - Arquitectura de bases de datos (incluyendo réplicas y particionamiento)
    - Servicios cloud utilizados (AWS, Firebase, etc.)
  + **Vista de datos**: Modelo de datos normalizado con entidades principales:
    - Esquema relacional para datos estructurados
    - Estrategia para almacenamiento de datos no estructurados (MongoDB)
    - Mecanismos de caching (Redis) para optimizar rendimiento
* **Prototipo arquitectónico ejecutable**: Implementación funcional que valida los aspectos críticos de la arquitectura de **Desparches**:
  + Frontend móvil con interfaces principales implementadas
  + API REST para operaciones fundamentales (búsqueda, filtrado, visualización)
  + Integración con servicios de geolocalización y mapas
  + Mecanismo básico de autenticación y autorización
  + Persistencia de datos esenciales

El prototipo permitirá demostrar las capacidades de búsqueda geolocalizada, visualización de detalles de eventos y el flujo básico de reserva, validando así la viabilidad técnica de los componentes críticos.

* **Plan detallado de desarrollo**: Documento que especifica:
  + Planificación de iteraciones para la fase de construcción:
    - Iteración 1: Autenticación y perfil de usuario
    - Iteración 2: Búsqueda y visualización de eventos/promociones
    - Iteración 3: Sistema de filtros y categorización
    - Iteración 4: Mecanismo de reservas y pagos
    - Iteración 5: Sistema de valoraciones y comentarios
    - Iteración 6: Recomendaciones personalizadas
    - Iteración 7: Notificaciones y alertas
    - Iteración 8: Integración con redes sociales
  + Asignación de recursos por iteración
  + Métricas de seguimiento y criterios de éxito
  + Estrategia de control de calidad y pruebas
  + Plan de gestión de riesgos actualizado
* **Lista actualizada de riesgos**: Versión refinada con riesgos residuales y nuevos riesgos identificados durante la elaboración:
  + Categorización según impacto y probabilidad
  + Estado actual de mitigación para cada riesgo
  + Indicadores tempranos para monitorizar riesgos activos
  + Planes de contingencia para riesgos de alta prioridad

**3.3 Construcción**

**3.3.1 Objetivos detallados**

* **Implementar todas las funcionalidades requeridas**: Desarrollar de manera sistemática todas las características planificadas para **Desparches**, siguiendo el plan de iteraciones establecido:
  + Completar los módulos frontend para Web, iOS y Android
  + Implementar todos los microservicios backend
  + Desarrollar el sistema de recomendaciones basado en IA
  + Integrar sistemas de pago y reservas
  + Implementar mecanismos de gamificación y fidelización
* **Optimizar recursos y procesos de desarrollo**: Maximizar la eficiencia durante la implementación de **Desparches**:
  + Establecer prácticas de código limpio y estándares de desarrollo
  + Implementar automatización para pruebas unitarias, de integración y end-to-end
  + Utilizar técnicas de integración continua y entrega continua (CI/CD)
  + Aplicar refactorización sistemática para mantener la calidad del código
  + Realizar revisiones de código periódicas entre pares
* **Lograr versiones incrementales estables**: Generar releases intermedias para validación y pruebas:
  + Versión alfa (uso interno): Implementación de funcionalidades core
  + Versión beta (usuarios seleccionados): Sistema completo con limitaciones controladas
  + Release candidate: Versión casi final para pruebas extensivas
* **Mantener la integridad de la arquitectura**: Asegurar que la implementación respete el diseño arquitectónico:
  + Realizar verificaciones de conformidad arquitectónica periódicas
  + Documentar y justificar cualquier desviación respecto al diseño original
  + Actualizar la documentación arquitectónica según sea necesario

**3.3.2 Productos (Artefactos) específicos**

* **Modelos completos y actualizados**:
  + **Modelo de casos de uso finalizado**: Documentación completa (100%) de todos los casos de uso de **Desparches**, incluyendo:
    - Flujos básicos y alternativos
    - Precondiciones y postcondiciones
    - Requisitos especiales
    - Puntos de extensión
  + **Modelo de análisis refinado**: Diagramas y documentación que representan con precisión:
    - Clases de análisis (boundary, control, entity)
    - Relaciones entre clases
    - Responsabilidades de cada clase
    - Trazabilidad a requisitos
  + **Modelo de diseño detallado**: Especificación técnica completa que incluye:
    - Diagramas de clases detallados con atributos, métodos y visibilidad
    - Patrones de diseño aplicados (Observer, Factory, Strategy, etc.)
    - Mecanismos de persistencia y transaccionalidad
    - Estrategias de optimización
  + **Modelo de implementación completo**: Documentación de la organización del código:
    - Estructura de paquetes y dependencias
    - Gestión de bibliotecas externas
    - Configuración de entornos
    - Procedimientos de despliegue
  + **Modelo de despliegue finalizado**: Especificación técnica para la infraestructura:
    - Topología de red
    - Configuración de servidores
    - Balanceo de carga y alta disponibilidad
    - Estrategia de copias de seguridad y recuperación
* **Arquitectura íntegra y validada**: Documentación arquitectónica completa y actualizada que refleja el sistema tal como ha sido implementado:
  + Decisiones arquitectónicas documentadas con sus justificaciones
  + Registro de cambios respecto al diseño original
  + Evaluación de atributos de calidad (rendimiento, seguridad, escalabilidad)
  + Documentación de interfaces entre componentes
* **Manual inicial de usuario**: Guía comprensiva para usuarios de **Desparches**:
  + **Para usuarios finales**:
    - Proceso de registro y configuración inicial
    - Navegación y descubrimiento de eventos/promociones
    - Reserva y pago
    - Gestión de perfil y preferencias
    - Valoración y compartición de experiencias
    - Preguntas frecuentes y solución de problemas comunes
  + **Para organizadores y comercios**:
    - Creación y gestión de eventos/promociones
    - Establecimiento de precios y disponibilidad
    - Visualización de estadísticas y análisis
    - Gestión de reservas y pagos
    - Comunicación con participantes
* **Prototipo operacional beta**: Versión completa pero no definitiva de **Desparches** para pruebas con usuarios reales:
  + Funcionalidades implementadas al 95-100%
  + Integración con servicios externos completada
  + Datos de ejemplo realistas pre-cargados
  + Instrumentación para obtener métricas de uso
  + Mecanismos de reporte de errores implementados
* **Material de soporte para pruebas beta**: Recursos para facilitar las pruebas con usuarios seleccionados:
  + Guía de instalación y configuración
  + Escenarios de prueba sugeridos
  + Formularios para reporte de errores
  + Encuestas de satisfacción y usabilidad
  + Canales de comunicación para feedback

**4. Requisitos**

**4.1 Elicitación de requerimientos funcionales**

El proceso de elicitación de requerimientos funcionales se beneficia del proyecto Desparches de varias maneras. Al tratarse de una plataforma que integra diversas funcionalidades para permitir a los usuarios descubrir eventos y promociones en su entorno, la claridad en las funciones del sistema facilita la identificación de los requerimientos específicos que debe cumplir el proyecto.

**4.2 Identificación Precisa de Funcionalidades Necesarias**

El sistema de Desparches ofrece funcionalidades específicas, como la búsqueda de eventos, la visualización de promociones y la interacción con mapas en tiempo real. Estos detalles permiten identificar de manera precisa las funcionalidades que deben ser incluidas en el sistema. La elicitación de requerimientos se beneficia de esta claridad, ya que al saber qué acciones deben realizar los usuarios, es más fácil definir qué debe hacer el sistema para cumplir con esas necesidades.

**4.3 Alineación con las Necesidades del Usuario**

El proyecto involucra usuarios que interactúan con una base de datos, un mapa interactivo y un algoritmo de recomendación. Estos componentes permiten ofrecer una experiencia personalizada para el usuario, lo que hace que el proceso de elicitación de los requerimientos esté alineado con las expectativas de los usuarios. Al conocer los diferentes roles de los usuarios y sus interacciones con la plataforma,se pueden identificar sus necesidades específicas y asegurarse de que el sistema cumpla con ellas.

**4.4 Definición Clara de Interacciones y Flujos**

Desparches también describe cómo los usuarios interactúan con el sistema para realizar tareas como guardar eventos, agregar reseñas o gestionar su perfil. Estos flujos de trabajo proporcionan una base sólida para recopilar los requisitos, ya que permiten identificar las acciones que deben estar disponibles y cómo deben organizarse dentro del sistema. La elicitación de estos requerimientos ayuda a garantizar que la experiencia del usuario sea fluida y eficiente.

**4.5 Priorización de Funcionalidades**

Al contar con una descripción clara de las funcionalidades del sistema, el proceso de elicitación también puede ayudar a priorizar las características más importantes. Por ejemplo, funcionalidades como la búsqueda de eventos y la interacción con el mapa en tiempo real pueden ser consideradas prioritarias para mejorar la experiencia del usuario. Esto asegura que el proyecto se enfoque en lo más relevante desde el principio.

**4.6 Evitar Ambigüedades**

Finalmente, la descripción detallada de cómo funciona el sistema ayuda a evitar ambigüedades en los requisitos, ya que proporciona un marco claro para entender las interacciones que se deben incluir. Esto reduce el riesgo de malentendidos durante la fase de desarrollo y asegura que el producto final cumpla con las expectativas y objetivos del proyecto de manera precisa.

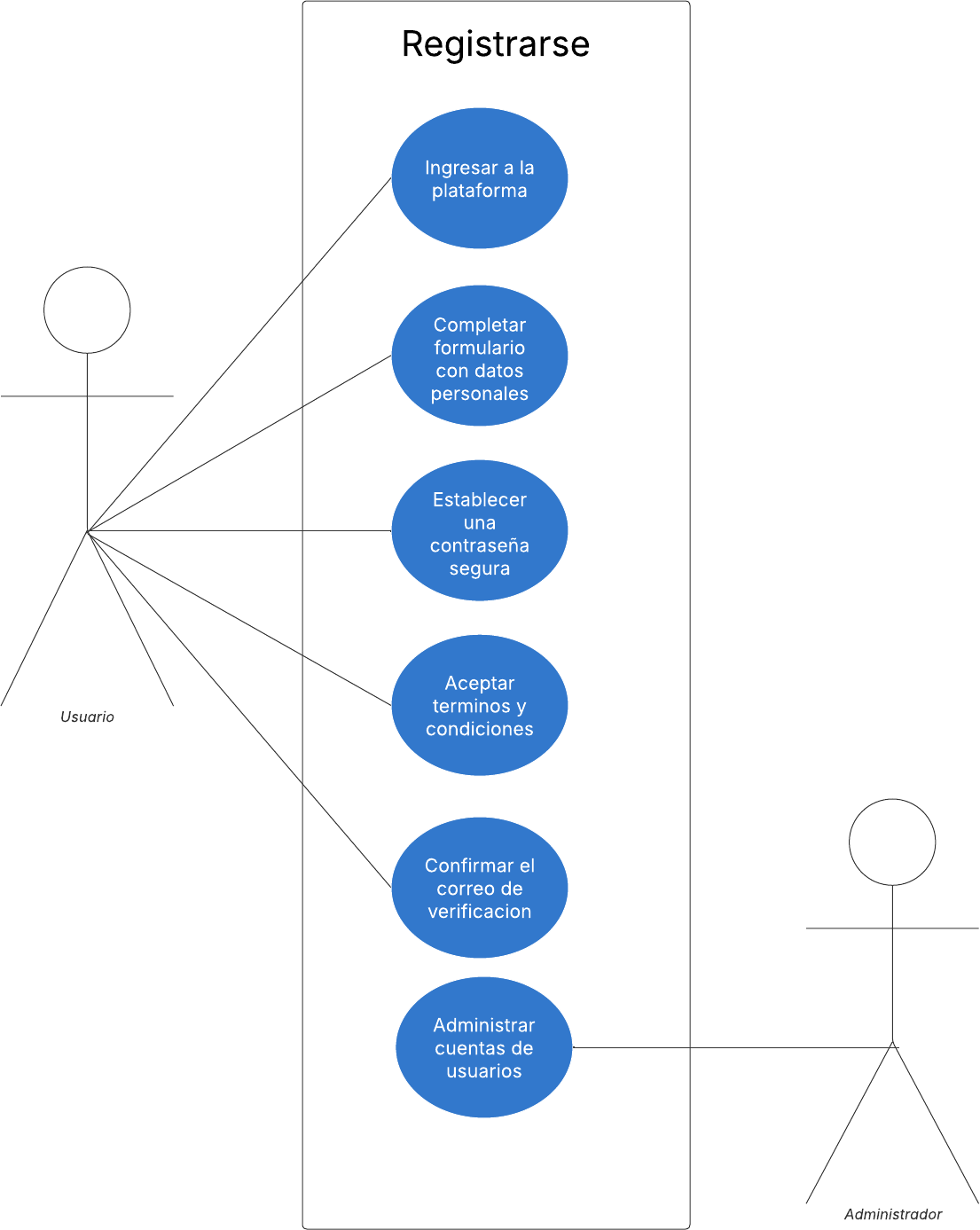
**4.7 Diagramas de Casos de Uso para Representar los Requerimientos Funcionales del Sistema**

1. Búsqueda de eventos: El sistema debe permitir a los usuarios buscar eventos utilizando filtros o categorías predefinidas, como tipo de evento, fecha o ubicación.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. Registro de usuarios**:** El sistema debe permitir que los usuarios se registren creando una cuenta personal mediante un proceso sencillo y seguro.



1. Visualización de eventos y promociones: Los usuarios deben poder ver una lista de eventos y promociones disponibles, con información relevante sobre cada uno.

Gráfico, Diagrama

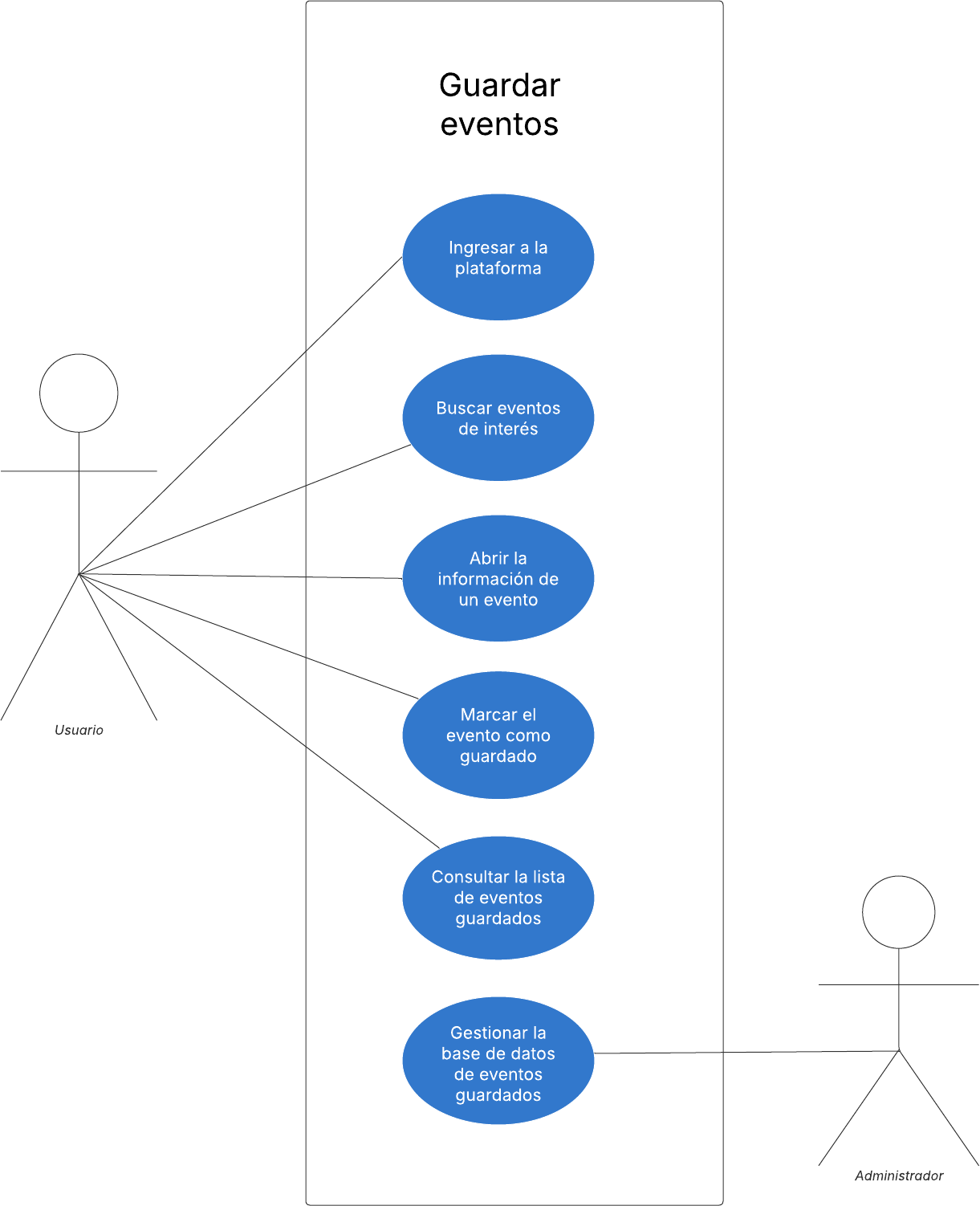
El contenido generado por IA puede ser incorrecto.

1. Navegación entre páginas: Los usuarios deben poder navegar fácilmente entre las siguientes páginas del sistema: página de inicio, página del mapa interactivo y página de perfil del usuario.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. Guardar eventos: Los usuarios deben tener la opción de guardar eventos de su interés para poder consultarlos más tarde.



1. Interacción con el mapa en tiempo real: El sistema debe permitir a los usuarios interactuar con un mapa en tiempo real donde se visualicen los eventos o actividades que están ocurriendo en su entorno.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. Visualización de información detallada de eventos: Los usuarios deben poder acceder a la información detallada de cada evento o actividad, incluyendo descripción, horarios, lugar, y otros detalles relevantes

Diagrama

El contenido generado por IA puede ser incorrecto..

1. Ajuste de preferencias y edición de perfil: El sistema debe permitir a los usuarios ajustar sus preferencias y editar su información personal desde su perfil.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. Lista de eventos guardados y asistidos: El perfil del usuario debe mostrar una lista de los eventos guardados y aquellos en los que ha asistido.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. Agregar reseñas a eventos**:** Los usuarios deben poder agregar reseñas y valoraciones a los eventos que hayan asistido, compartiendo su experiencia con otros usuarios.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. Ver preguntas frecuentes y soporte técnico: El sistema debe permitir a los usuarios acceder a una sección de preguntas frecuentes (FAQ) y ponerse en contacto con soporte técnico para resolver cualquier inconveniente.

Diagrama

El contenido generado por IA puede ser incorrecto.

* 1. Procesos aplicados en Desparches

Los procesos en "Desparches" pueden entenderse como los pasos o flujos de trabajo que permiten que el sistema funcione correctamente. A continuación, se describen los procesos principales del sistema, aplicados al proyecto.

1. Registro y autenticación de usuarios
   * Entrada: Un usuario accede a la plataforma e inicia sesión o se registra.
     + Proceso:
     + El usuario ingresa sus datos en el formulario de registro.
     + El sistema valida la información y la almacena en la base de datos.
     + Si la cuenta ya existe, se notifica al usuario.
     + Si el registro es exitoso, el usuario puede personalizar sus preferencias.
   * Salida: El usuario puede acceder a su perfil y comenzar a explorar la plataforma.
2. Publicación de eventos y promociones por empresas
   * Entrada: Una empresa inicia sesión y accede a su panel de control.
     + Proceso:
     + La empresa selecciona la opción de "Crear evento" o "Publicar promoción".
     + Llena un formulario con los detalles (nombre, descripción, fecha, ubicación, imagen, etc.).
     + El sistema valida la información y almacena el evento en la base de datos.
   * Se actualiza el mapa interactivo y la lista de eventos.

* Salida: El evento o promoción es visible para los usuarios en la plataforma.

1. Búsqueda y recomendación de eventos
   * Entrada: Un usuario busca eventos o promociones en la plataforma.
     + Proceso:
     + El usuario ingresa palabras clave o usa filtros (categoría, ubicación, fecha).
     + El sistema consulta la base de datos y obtiene los eventos relevantes.
     + Un algoritmo de recomendación sugiere eventos basados en las preferencias del usuario.
     + Se muestran los eventos en un mapa interactivo y en una lista detallada.
   * Salida: El usuario encuentra eventos y promociones cercanas a sus intereses.
2. Interacción con el mapa en tiempo real
   * Entrada: Un usuario desea ver eventos en su ubicación.
     + Proceso:
     + El usuario accede al mapa interactivo.
     + Se obtiene la ubicación actual del usuario (con permiso de GPS).
     + Se filtran los eventos más cercanos y se muestran en el mapa con marcadores.
     + El usuario puede hacer clic en un evento para ver más información.
   * Salida: El usuario visualiza eventos y promociones en su entorno.
3. Gestión de eventos guardados y asistidos
   * Entrada: Un usuario quiere guardar eventos o revisar su historial de asistencia.
     + Proceso:
     + El usuario selecciona un evento y elige la opción "Guardar".
     + El evento se almacena en su perfil dentro de "Eventos guardados".
     + Si asiste a un evento, el sistema lo registra automáticamente.
     + La lista de eventos asistidos se actualiza en su perfil.
   * Salida: El usuario puede ver eventos guardados y su historial de asistencia.
4. Agregar reseñas y valoraciones
   * Entrada: Un usuario desea compartir su experiencia sobre un evento.
     + Proceso:
     + El usuario selecciona un evento asistido.
     + Escribe una reseña y califica el evento con estrellas.
     + El sistema valida y almacena la reseña en la base de datos.
     + Otros usuarios pueden ver la reseña dentro de la información del evento.
   * Salida: La reseña y calificación quedan publicadas para ayudar a otros usuarios.
5. Acceso a soporte técnico y preguntas frecuentes
   * Entrada: Un usuario tiene una duda o problema con la plataforma.
     + Proceso:
     + El usuario accede a la sección de preguntas frecuentes (FAQ).
     + Si no encuentra la respuesta, elige la opción de contactar soporte técnico.
     + Completa un formulario de ayuda o accede a un chat en vivo.
     + El equipo de soporte revisa el caso y responde con una solución.
   * Salida: El usuario recibe asistencia y su problema es resuelto.
   1. Prototipos aplicados a Desparches

Para el desarrollo de Desparches, se implementaron diversos prototipos en Figma con el objetivo de diseñar y optimizar la experiencia del usuario antes de su implementación final. Se crearon dos versiones principales del prototipo:

1. Versión Web: Diseñada para su uso en navegadores de escritorio, enfocada en una navegación intuitiva y una visualización óptima de eventos.
   1. Versión Móvil: Adaptada a dispositivos móviles, con una interfaz optimizada para exploración rápida y geolocalización eficiente.
   2. Roles en los Prototipos

Se definieron dos roles principales dentro de la plataforma, cada uno con accesos y funcionalidades específicas:

* + - Usuarios:
      * Acceden a la plataforma para descubrir eventos y promociones.
      * Pueden explorar el mapa interactivo en tiempo real.
      * Guardan eventos de interés y reciben recomendaciones personalizadas.
    - Administradores (Empresas/Organizadores):
      * Gestionan el contenido de la plataforma.
      * Crean, modifican o eliminan eventos y promociones.
      * Supervisan la actividad y autenticidad de los eventos.

Los prototipos en Figma han permitido visualizar y estructurar la interfaz de Desparches, asegurando una experiencia de usuario intuitiva y eficiente. Con estos diseños, se ha logrado representar fielmente las funcionalidades clave, facilitando futuras mejoras e integraciones tecnológicas.

* 1. Aplicación de Requerimientos en el Prototipo de Desparches

1. Pestaña de inicio:
   * Usuario

Interfaz de usuario gráfica, Sitio web

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación, Sitio web

El contenido generado por IA puede ser incorrecto.

# La pestaña de inicio de la interfaz del usuario permite a los usuarios buscar eventos y promociones mediante filtros predefinidos como tipo de evento, fecha, ubicación y precio, mostrando resultados en tiempo real. Además, la plataforma ofrece una visualización dinámica de eventos y promociones activas, organizada de manera intuitiva y con una sección de destacados basada en las preferencias del usuario.

* + Interfaz de usuario gráfica

    El contenido generado por IA puede ser incorrecto.Administradores:

Interfaz de usuario gráfica, Aplicación, Sitio web

El contenido generado por IA puede ser incorrecto.

# La pestaña de inicio para administradores permite gestionar de manera eficiente los eventos y promociones de la plataforma. En esta sección, los administradores pueden visualizar los eventos creados, filtrarlos por categoría, fecha o popularidad, y acceder a detalles específicos de cada evento. Además, cuentan con botones de administración de los eventos, garantizando que la información publicada sea precisa y actualizada.

1. Interfaz de usuario gráfica, Aplicación

   El contenido generado por IA puede ser incorrecto.Pestaña mapa interactivo:

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

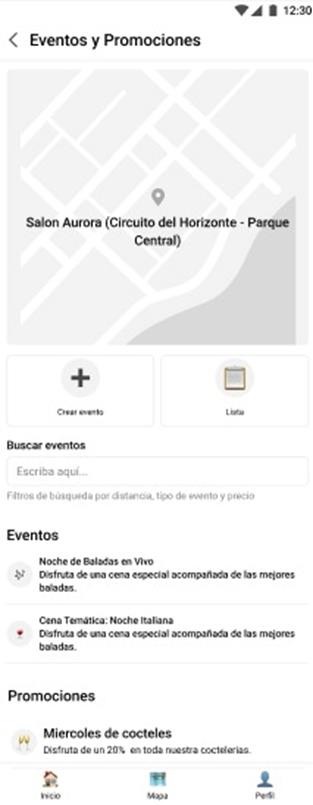
# La pestaña del mapa interactivo en Desparches permite a los usuarios explorar eventos y actividades en su entorno en tiempo real. A través del mapa, pueden visualizar los eventos disponibles, desplazarse por distintas ubicaciones y acceder a información detallada de cada evento.

* 1. Interfaz de usuario gráfica, Aplicación

     El contenido generado por IA puede ser incorrecto.Pestaña de creación y administración de eventos (Administrador):

Interfaz de usuario gráfica, Aplicación, Sitio web

El contenido generado por IA puede ser incorrecto.



En estas pestañas, el administrador puede visualizar una lista de los eventos y promociones que ha creado. Además, mediante el botón "Crear Proyecto", puede acceder a la página de creación de eventos e ingresar detalles como el nombre, ubicación y fecha del evento.

1. Pestaña de perfil

# Usuario:

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

La pestaña de perfil del usuario permite gestionar su experiencia en la plataforma de manera personalizada. Desde esta sección, los usuarios pueden editar su información personal, ajustar sus preferencias y acceder a una lista de eventos guardados y asistidos. Además, tienen la opción de agregar reseñas y valoraciones a los eventos en los que han participado. También pueden consultar la sección de preguntas frecuentes y contactar al soporte técnico en caso de necesitar ayuda.

# Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico El contenido generado por IA puede ser incorrecto.Administrador:

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

La pestaña de perfil del administrador permite gestionar su configuración y acceder a herramientas clave de la plataforma. Desde esta sección, el administrador puede ajustar sus preferencias, visualizar el panel de administración, consultar la lista de eventos creados y revisar las reseñas recibidas. Además, cuenta con accesos directos a la sección de preguntas frecuentes y soporte técnico para resolver cualquier inquietud.

* 1. Interfaz de usuario gráfica, Aplicación

     El contenido generado por IA puede ser incorrecto.Pestaña de inicio de sesión:

La página de inicio de sesión permite a los usuarios ingresar sus credenciales y acceder a la plataforma. Al validar el usuario y la contraseña, el sistema identifica si la cuenta corresponde a un administrador o a un usuario, dirigiéndolos a la interfaz correspondiente según su rol.