



Tabla de contenido

[**Introducción 2**](#_heading=h.gjdgxs)

[**Planificación de Sprints 4**](#_heading=h.30j0zll)

[**Objetivos del Sprint 6**](#_heading=h.1fob9te)

[**Distribución de Historias de Usuario por Sprint 8**](#_heading=h.3znysh7)

[**Tareas y Actividades por Sprint 11**](#_heading=h.2et92p0)

[**Asignación de Responsabilidades 14**](#_heading=h.tyjcwt)

[**Distribución de Tareas por Sprint 15**](#_heading=h.3dy6vkm)

[**Responsabilidades de Seguimiento y Cumplimiento 17**](#_heading=h.1t3h5sf)

[**Criterios de Finalización del Sprint 18**](#_heading=h.4d34og8)

[**Criterios para Considerar las Tareas como Completadas 18**](#_heading=h.2s8eyo1)

[**Verificación de Requisitos y Estándares de Calidad 19**](#_heading=h.17dp8vu)

[**Seguimiento y Avance Diario (Daily Scrum) 20**](#_heading=h.3rdcrjn)

[**Procedimiento para Identificar y Abordar Impedimentos 21**](#_heading=h.26in1rg)

[**Gráficos de Progreso del Sprint 23**](#_heading=h.lnxbz9)

[**Análisis Visual del Cumplimiento de los Objetivos en Tiempo Real 24**](#_heading=h.35nkun2)

[**Retrospectiva del Sprint 25**](#_heading=h.1ksv4uv)

[**Evaluación de lo que Funcionó 25**](#_heading=h.44sinio)

[**Áreas a Mejorar 26**](#_heading=h.2jxsxqh)

[**Propuestas de Acción para Mejorar en Futuros Sprints 27**](#_heading=h.z337ya)

[**Conclusión del Sprint 28**](#_heading=h.3j2qqm3)

[**Resumen de los Logros y Metas Alcanzadas 28**](#_heading=h.1y810tw)

[**Reflexión sobre el Cumplimiento de los Objetivos 29**](#_heading=h.4i7ojhp)

[**Oportunidades de Mejora para el Próximo Sprint 29**](#_heading=h.2xcytpi)

[**Conclusión General del Sprint** 30](#_heading=h.1ci93xb)

# **Introducción**

El **Sprint Backlog** es una de las herramientas fundamentales dentro de la metodología Scrum, diseñada para organizar y gestionar las tareas específicas que el equipo de desarrollo abordará durante un sprint. Su propósito principal es permitir un enfoque detallado en las historias de usuario y tareas seleccionadas del **Product Backlog** para cumplir con los objetivos del sprint y entregar un incremento de valor al final de cada iteración. Mientras que el Product Backlog representa la lista completa y priorizada de todas las funcionalidades y requerimientos del proyecto, el Sprint Backlog se centra en las actividades específicas y alcanzables dentro de un marco de tiempo definido.

El Sprint Backlog contiene únicamente las historias de usuario, tareas y actividades que se consideran factibles y prioritarias para completar durante el sprint actual. Esta selección permite al equipo de desarrollo concentrarse en objetivos claros y alcanzar entregables concretos que aportan valor inmediato al proyecto. Durante el proceso de planificación de cada sprint, el equipo evalúa las historias de usuario del Product Backlog, decide cuáles incluir en el Sprint Backlog y define las tareas detalladas que deben completarse para dar por finalizada cada historia.

La relación entre el **Product Backlog** y el **Sprint Backlog** es continua y colaborativa. El Product Backlog proporciona una visión amplia y priorizada de los requerimientos y funcionalidades del proyecto, mientras que el Sprint Backlog representa un subconjunto específico de tareas que se extraen de esta lista general. Al inicio de cada sprint, el equipo de desarrollo y el Product Owner se reúnen para discutir y priorizar las historias de usuario, asegurando que las actividades del Sprint Backlog estén alineadas con las prioridades del proyecto y con las expectativas del usuario final.

Esta organización garantiza que el equipo trabaje de manera ágil y adaptable, abordando incrementos manejables de trabajo y asegurando que cada sprint produzca entregables funcionales y valiosos. A lo largo del sprint, el equipo realiza reuniones diarias (Daily Scrum) para revisar el progreso, discutir impedimentos y ajustar el Sprint Backlog según sea necesario, siempre manteniendo el enfoque en los objetivos específicos del sprint.

# **Planificación de Sprints**

La **Planificación de Sprints** es un paso clave en la metodología Scrum que define el alcance y los objetivos de cada sprint en el desarrollo de **FlashMatch**. Cada sprint tiene una duración específica y un conjunto de objetivos que guían al equipo hacia entregables concretos y de valor para los usuarios. Esta planificación garantiza que cada iteración se enfoque en prioridades claras y en funcionalidades esenciales, permitiendo avanzar de manera incremental hacia la finalización del proyecto.

**Duración y Objetivo de Cada Sprint**

La duración de cada sprint en **FlashMatch** es de dos semanas, lo cual ofrece un equilibrio ideal entre la velocidad de entrega y el tiempo necesario para desarrollar y probar funcionalidades completas. Al inicio de cada sprint, el equipo define un objetivo concreto que guiará las actividades y tareas del sprint. Estos objetivos están alineados con las prioridades del Product Backlog y buscan avanzar de manera estructurada hacia la implementación completa del proyecto.

**Resumen de Objetivos por Sprint**:

* **Sprint 1**: Configuración inicial y desarrollo de funcionalidades básicas de gestión de usuarios (registro e inicio de sesión).
* **Sprint 2**: Organización de partidos, permitiendo a los usuarios crear y unirse a partidos.
* **Sprint 3**: Implementación de la funcionalidad de reserva de canchas y consulta de disponibilidad.
* **Sprint 4**: Habilitación de notificaciones en tiempo real y mensajería interna entre jugadores.
* **Sprint 5**: Integración del sistema de pagos y desarrollo del historial de transacciones.
* **Sprint 6**: Estadísticas y rendimiento del jugador, proporcionando métricas de progreso personal.
* **Sprint 7**: Pruebas finales e integración completa para preparar el producto para el lanzamiento.

**Estrategias para la Selección de Historias de Usuario**

Para asegurar que cada sprint esté alineado con los objetivos generales del proyecto y maximice el valor entregado al usuario, se implementan varias estrategias para la selección de historias de usuario:

1. **Priorización por Valor y Complejidad**:
   * Las historias de usuario se priorizan según su valor para el usuario y la complejidad de implementación. Funcionalidades clave, como el registro de usuarios o el sistema de reservas, se desarrollan en los primeros sprints para establecer la base de la plataforma.
2. **Estimación de Puntos de Historia**:
   * Cada historia de usuario se estima en puntos de historia, reflejando el esfuerzo y tiempo necesario para completarla. Esto ayuda al equipo a seleccionar historias que se ajusten a la capacidad de trabajo disponible en el sprint.
3. **Dependencias entre Funcionalidades**:
   * Las historias se organizan según sus dependencias. Las funcionalidades esenciales y que sirven como base para otras características, como la autenticación de usuarios, se implementan primero para soportar futuras funcionalidades, como la creación de partidos y el sistema de pago.
4. **Feedback y Revisión Continua**:
   * Al final de cada sprint, el equipo evalúa el progreso y recoge feedback del Product Owner y, si es posible, de usuarios. Esto permite ajustar las prioridades en el Product Backlog y definir las historias que se incluirán en el próximo sprint, asegurando una adaptación constante a las necesidades del proyecto.
5. **Capacidad del Equipo y Tiempos de Entrega**:
   * La selección de historias se adapta a la capacidad del equipo en función del esfuerzo estimado y los tiempos de entrega acordados. Si un sprint tiene tareas más complejas, se seleccionan historias menos complejas para equilibrar la carga de trabajo y asegurar la finalización de los objetivos.

La planificación de sprints en **FlashMatch** asegura que cada iteración se enfoque en objetivos específicos y manejables, optimizando la entrega de valor incremental y garantizando que el desarrollo se mantenga alineado con las prioridades y necesidades de los usuarios. Las estrategias de selección de historias de usuario permiten que el equipo avance de manera estructurada, manteniendo una adaptación constante a los cambios y asegurando que el producto evolucione de acuerdo a los estándares de calidad y funcionalidad esperados.

# **Objetivos del Sprint**

Los **Objetivos del Sprint** en **FlashMatch** se establecen para guiar al equipo en la creación de incrementos de valor concretos en cada iteración. Cada sprint tiene un objetivo claro, enfocado en completar funcionalidades específicas que progresivamente construyen el producto final. A continuación, se detallan los objetivos específicos de cada sprint y el valor incremental que se espera al final de cada uno.

**Sprint 1: Configuración Inicial y Gestión de Usuarios**

* **Objetivo**: Configurar el entorno de desarrollo y completar las funcionalidades básicas de gestión de usuarios, como el registro y el inicio de sesión.
* **Valor Incremental Esperado**: Permitir a los usuarios crear y acceder a sus cuentas personales, estableciendo una base sólida para la personalización y la seguridad en la plataforma.

**Sprint 2: Organización de Partidos**

* **Objetivo**: Desarrollar la funcionalidad de creación y búsqueda de partidos, permitiendo a los usuarios organizar y unirse a eventos deportivos.
* **Valor Incremental Esperado**: Facilitar la conectividad entre usuarios al ofrecer una opción para organizar y unirse a partidos de manera ágil, promoviendo la participación activa.

**Sprint 3: Reserva de Canchas**

* **Objetivo**: Implementar el sistema de reserva de canchas, con la posibilidad de consultar disponibilidad y aplicar filtros.
* **Valor Incremental Esperado**: Proveer a los usuarios de una herramienta para gestionar la disponibilidad de canchas y reservar espacios en tiempo real, mejorando la organización y planificación de eventos.

**Sprint 4: Notificaciones y Comunicaciones**

* **Objetivo**: Activar el sistema de notificaciones en tiempo real y habilitar la mensajería interna entre usuarios.
* **Valor Incremental Esperado**: Mantener a los usuarios informados sobre actualizaciones de partidos y facilitar la comunicación directa entre jugadores, mejorando la experiencia interactiva de la plataforma.

**Sprint 5: Sistema de Pago**

* **Objetivo**: Integrar métodos de pago seguros y desarrollar un historial de transacciones para los usuarios.
* **Valor Incremental Esperado**: Ofrecer una experiencia de pago confiable y transparente, permitiendo a los usuarios realizar transacciones y gestionar sus reservas de manera segura.

**Sprint 6: Estadísticas y Rendimiento del Jugador**

* **Objetivo**: Implementar una funcionalidad de estadísticas de rendimiento para que los jugadores puedan visualizar su progreso y comparar su desempeño.
* **Valor Incremental Esperado**: Fomentar el interés y la mejora continua de los jugadores mediante la visualización de estadísticas personales y comparativas, incentivando la participación y el rendimiento.

**Sprint 7: Pruebas Finales e Integración Completa**

* **Objetivo**: Realizar pruebas de integración y calidad, optimizando la plataforma para el lanzamiento final y asegurando su estabilidad.
* **Valor Incremental Esperado**: Ofrecer una versión final de **FlashMatch** lista para el lanzamiento, garantizando la cohesión de todas las funcionalidades y la satisfacción del usuario.

# **Distribución de Historias de Usuario por Sprint**

La **Distribución de Historias de Usuario por Sprint** permite organizar el trabajo en **FlashMatch** en incrementos específicos, asegurando que cada sprint se enfoque en desarrollar funcionalidades clave. La selección de historias de usuario por sprint se realiza según la prioridad y el valor que cada historia aporta al usuario, optimizando el flujo de trabajo y cumpliendo con los objetivos del proyecto.

**Sprint 1: Configuración Inicial y Gestión de Usuarios**

**Historias de Usuario Seleccionadas**:

* **HU1**: Como usuario, quiero registrarme en la plataforma para acceder a mis partidos y reservas.
* **HU2**: Como usuario, quiero iniciar sesión para mantener la seguridad de mi cuenta.

**Prioridades en el Sprint**:

1. Implementar registro de usuarios con validación de datos.
2. Implementar inicio de sesión con autenticación básica.

**Sprint 2: Organización de Partidos**

**Historias de Usuario Seleccionadas**:

* **HU3**: Como usuario, quiero crear un partido para invitar a otros jugadores.
* **HU4**: Como usuario, quiero buscar partidos cercanos para unirse a eventos.

**Prioridades en el Sprint**:

1. Implementar funcionalidad de creación de partidos, incluyendo detalles de partido.
2. Desarrollar búsqueda de partidos con filtros por ubicación y hora.

**Sprint 3: Reserva de Canchas**

**Historias de Usuario Seleccionadas**:

* **HU5**: Como usuario, quiero ver la disponibilidad de canchas para organizar un partido.
* **HU6**: Como usuario, quiero reservar una cancha para asegurar mi espacio.

**Prioridades en el Sprint**:

1. Desarrollar funcionalidad para ver la disponibilidad en tiempo real.
2. Implementar sistema de reservas con confirmación.

**Sprint 4: Notificaciones y Comunicaciones**

**Historias de Usuario Seleccionadas**:

* **HU7**: Como usuario, quiero recibir notificaciones sobre cambios en partidos para mantenerme informado.
* **HU8**: Como usuario, quiero enviar mensajes a otros jugadores para coordinar detalles del partido.

**Prioridades en el Sprint**:

1. Configurar notificaciones en tiempo real para partidos y reservas.
2. Habilitar sistema de mensajería entre usuarios con historial de mensajes.

**Sprint 5: Sistema de Pago**

**Historias de Usuario Seleccionadas**:

* **HU9**: Como usuario, quiero seleccionar un método de pago para completar mis reservas.
* **HU10**: Como usuario, quiero ver mi historial de pagos para llevar control de mis transacciones.

**Prioridades en el Sprint**:

1. Integrar opciones de pago seguras en la plataforma.
2. Crear funcionalidad de historial de pagos accesible para el usuario.

**Sprint 6: Estadísticas y Rendimiento del Jugador**

**Historias de Usuario Seleccionadas**:

* **HU11**: Como usuario, quiero ver mis estadísticas de rendimiento para mejorar mi juego.
* **HU12**: Como usuario, quiero comparar mi rendimiento con el de otros jugadores para motivarme.

**Prioridades en el Sprint**:

1. Desarrollar funcionalidad de visualización de estadísticas individuales.
2. Implementar comparativa de estadísticas entre jugadores.

**Sprint 7: Pruebas Finales e Integración Completa**

**Historias de Usuario Seleccionadas**:

* **HU13**: Realizar pruebas de integración para verificar la cohesión de todas las funcionalidades.
* **HU14**: Ajustes finales de optimización para el lanzamiento de la plataforma.

**Prioridades en el Sprint**:

1. Realizar pruebas de calidad y resolver errores encontrados.
2. Optimizar la plataforma y validar la funcionalidad de cada módulo.

La distribución de historias de usuario por sprint permite que el equipo de **FlashMatch** mantenga un enfoque claro en las funcionalidades esenciales y prioritarias en cada iteración. Esta estructura asegura que cada sprint entregue un incremento de valor al usuario final, permitiendo un avance controlado y adaptable que fortalece la experiencia de uso y la cohesión de la plataforma.

# **Tareas y Actividades por Sprint**

El desglose de cada historia de usuario en tareas específicas permite organizar y asignar el trabajo en cada sprint de **FlashMatch**. A continuación, se detalla la lista de tareas necesarias para completar cada historia de usuario en cada sprint, junto con su estimación en horas.

**Sprint 1: Configuración Inicial y Gestión de Usuarios**

**Historia de Usuario HU1**: Registro de Usuario

* **Tarea 1**: Crear formulario de registro de usuario (4 horas).
* **Tarea 2**: Implementar validación de datos de entrada (6 horas).
* **Tarea 3**: Configurar base de datos para almacenar información de usuario (5 horas).
* **Tarea 4**: Programar verificación de correo electrónico (8 horas).

**Historia de Usuario HU2**: Inicio de Sesión

* **Tarea 1**: Crear formulario de inicio de sesión (3 horas).
* **Tarea 2**: Configurar autenticación básica y sesión de usuario (7 horas).
* **Tarea 3**: Implementar mensajes de error para datos incorrectos (4 horas).

**Sprint 2: Organización de Partidos**

**Historia de Usuario HU3**: Crear Partido

* **Tarea 1**: Diseñar interfaz de creación de partido (5 horas).
* **Tarea 2**: Implementar funcionalidad para ingresar detalles del partido (7 horas).
* **Tarea 3**: Configurar almacenamiento de datos de partidos en la base de datos (6 horas).

**Historia de Usuario HU4**: Búsqueda de Partidos

* **Tarea 1**: Crear interfaz de búsqueda de partidos (5 horas).
* **Tarea 2**: Configurar filtros de ubicación y horario (8 horas).
* **Tarea 3**: Implementar visualización de resultados de búsqueda (6 horas).

**Sprint 3: Reserva de Canchas**

**Historia de Usuario HU5**: Ver Disponibilidad de Canchas

* **Tarea 1**: Desarrollar calendario de disponibilidad de canchas (8 horas).
* **Tarea 2**: Configurar filtros de búsqueda en el calendario (5 horas).
* **Tarea 3**: Sincronizar disponibilidad con base de datos en tiempo real (7 horas).

**Historia de Usuario HU6**: Reservar Cancha

* **Tarea 1**: Crear formulario de reserva de cancha (4 horas).
* **Tarea 2**: Implementar confirmación de reserva y notificación al usuario (6 horas).
* **Tarea 3**: Actualizar estado de disponibilidad de la cancha en la base de datos (5 horas).

**Sprint 4: Notificaciones y Comunicaciones**

**Historia de Usuario HU7**: Recibir Notificaciones

* **Tarea 1**: Configurar sistema de notificaciones en tiempo real (8 horas).
* **Tarea 2**: Programar envío de notificaciones para cambios en partidos (6 horas).
* **Tarea 3**: Implementar preferencias de notificaciones en el perfil del usuario (5 horas).

**Historia de Usuario HU8**: Mensajes entre Jugadores

* **Tarea 1**: Crear sistema de mensajería interna (10 horas).
* **Tarea 2**: Programar almacenamiento de mensajes en base de datos (6 horas).
* **Tarea 3**: Configurar notificaciones de nuevos mensajes recibidos (4 horas).

**Sprint 5: Sistema de Pago**

**Historia de Usuario HU9**: Seleccionar Método de Pago

* **Tarea 1**: Integrar métodos de pago (tarjeta de crédito, débito) (10 horas).
* **Tarea 2**: Implementar selección de método de pago en la interfaz de usuario (5 horas).
* **Tarea 3**: Configurar validación y procesamiento de pagos seguros (8 horas).

**Historia de Usuario HU10**: Ver Historial de Pagos

* **Tarea 1**: Crear interfaz de historial de pagos para el usuario (6 horas).
* **Tarea 2**: Configurar almacenamiento de transacciones en la base de datos (4 horas).
* **Tarea 3**: Permitir descarga de comprobantes de pago (4 horas).

**Sprint 6: Estadísticas y Rendimiento del Jugador**

**Historia de Usuario HU11**: Ver Estadísticas Personales

* **Tarea 1**: Desarrollar visualización de estadísticas individuales (8 horas).
* **Tarea 2**: Implementar cálculo de métricas de rendimiento (6 horas).
* **Tarea 3**: Configurar actualización automática de estadísticas (5 horas).

**Historia de Usuario HU12**: Comparar Rendimiento con Otros Jugadores

* **Tarea 1**: Crear funcionalidad de comparación de rendimiento (8 horas).
* **Tarea 2**: Diseñar visualización gráfica para comparación (5 horas).
* **Tarea 3**: Programar filtros de comparación por fecha y tipo de partido (4 horas).

**Sprint 7: Pruebas Finales e Integración Completa**

**Historia de Usuario HU13**: Pruebas de Integración

* **Tarea 1**: Realizar pruebas de integración para cada módulo (10 horas).
* **Tarea 2**: Resolver errores y optimizar rendimiento (8 horas).

**Historia de Usuario HU14**: Ajustes Finales y Preparación para el Lanzamiento

* **Tarea 1**: Realizar pruebas de usabilidad y accesibilidad (6 horas).
* **Tarea 2**: Documentar funcionalidades y preparar para el lanzamiento (5 horas).
* **Tarea 3**: Validar la cohesión entre todos los módulos del sistema (5 horas).

# **Asignación de Responsabilidades**

La **Asignación de Responsabilidades** en **FlashMatch** permite dividir y delegar tareas específicas entre los miembros del equipo para asegurar una gestión eficaz y una ejecución organizada de cada sprint. A continuación, se presenta la distribución de tareas por miembro del equipo, junto con sus responsabilidades de seguimiento y cumplimiento.

**Equipo y Roles**

1. **Rodrigo Berríos** - **Product Owner (PO)**
   * **Responsabilidades**:
     + Definir y priorizar las historias de usuario en el Product Backlog.
     + Asegurar que las funcionalidades cumplan con las expectativas del cliente y los usuarios finales.
     + Revisar el trabajo al final de cada sprint para validar el cumplimiento de los objetivos.
     + Proveer feedback constante y decisiones clave para ajustar el desarrollo.
2. **Carlos Guzmán** - **Scrum Master (SM)**
   * **Responsabilidades**:
     + Facilitar las reuniones diarias (Daily Scrum) y sesiones de planificación de sprint.
     + Eliminar obstáculos que puedan afectar el progreso del equipo de desarrollo.
     + Asegurar la adherencia a la metodología Scrum y el cumplimiento de las mejores prácticas ágiles.
     + Monitorear el avance de cada tarea y guiar al equipo en la optimización del proceso.
3. **Daniel González** - **Desarrollador Frontend**
   * **Responsabilidades**:
     + Desarrollar las interfaces de usuario en Angular e Ionic para la aplicación web y móvil.
     + Implementar el diseño de cada historia de usuario que implique componentes visuales.
     + Realizar pruebas de usabilidad y asegurar la responsividad en distintos dispositivos.
     + Colaborar con el desarrollador Backend para integrar el frontend con las API del sistema.
4. **Desarrollador Backend** (Rol General en el Equipo)
   * **Responsabilidades**:
     + Implementar la lógica del servidor y la arquitectura en NestJS y microservicios.
     + Configurar la autenticación y autorización mediante JWT.
     + Integrar el sistema de pago y el manejo de la base de datos en PostgreSQL.
     + Garantizar la seguridad y eficiencia en la comunicación entre microservicios.

# **Distribución de Tareas por Sprint**

Cada miembro del equipo tiene asignadas tareas específicas en cada sprint, alineadas con sus roles y habilidades para asegurar el cumplimiento de los objetivos.

* **Sprint 1**:
  + **Rodrigo Berríos (PO)**: Definir y revisar historias de usuario de registro e inicio de sesión.
  + **Carlos Guzmán (SM)**: Facilitar el kickoff del sprint y asegurar la organización del equipo.
  + **Daniel González (Frontend)**: Desarrollar interfaz de registro e inicio de sesión.
  + **Desarrollador Backend**: Configurar la autenticación básica y la base de datos de usuarios.
* **Sprint 2**:
  + **Rodrigo Berríos (PO)**: Priorizar historias de usuario para la organización de partidos.
  + **Carlos Guzmán (SM)**: Supervisar el progreso de tareas y guiar las reuniones de seguimiento.
  + **Daniel González (Frontend)**: Crear la interfaz para la creación y búsqueda de partidos.
  + **Desarrollador Backend**: Desarrollar la funcionalidad para almacenar y gestionar datos de partidos.
* **Sprint 3**:
  + **Rodrigo Berríos (PO)**: Validar la funcionalidad de reserva de canchas.
  + **Carlos Guzmán (SM)**: Facilitar la coordinación para integrar la disponibilidad en tiempo real.
  + **Daniel González (Frontend)**: Diseñar el calendario de disponibilidad de canchas.
  + **Desarrollador Backend**: Implementar lógica de reservas y sincronización de disponibilidad.
* **Sprint 4**:
  + **Rodrigo Berríos (PO)**: Validar el sistema de notificaciones y mensajería.
  + **Carlos Guzmán (SM)**: Supervisar la integración de notificaciones en tiempo real.
  + **Daniel González (Frontend)**: Crear la interfaz de notificaciones y mensajería.
  + **Desarrollador Backend**: Configurar el sistema de notificaciones y almacenar mensajes.
* **Sprint 5**:
  + **Rodrigo Berríos (PO)**: Revisar y aprobar el sistema de pagos y su usabilidad.
  + **Carlos Guzmán (SM)**: Asegurar que el sistema de pago se integre sin problemas.
  + **Daniel González (Frontend)**: Diseñar la interfaz de pago y el historial de transacciones.
  + **Desarrollador Backend**: Integrar los métodos de pago y validar la seguridad de transacciones.
* **Sprint 6**:
  + **Rodrigo Berríos (PO)**: Asegurar que las estadísticas reflejan el rendimiento de los jugadores.
  + **Carlos Guzmán (SM)**: Coordinar la implementación y prueba de estadísticas.
  + **Daniel González (Frontend)**: Diseñar la visualización de estadísticas y gráficos de comparación.
  + **Desarrollador Backend**: Calcular y almacenar estadísticas de rendimiento.
* **Sprint 7**:
  + **Rodrigo Berríos (PO)**: Validar la funcionalidad completa para el lanzamiento.
  + **Carlos Guzmán (SM)**: Supervisar pruebas de integración y optimización final.
  + **Daniel González (Frontend)**: Realizar pruebas de usabilidad y optimizar la interfaz.
  + **Desarrollador Backend**: Optimizar el rendimiento y garantizar la seguridad en la integración de microservicios.

# **Responsabilidades de Seguimiento y Cumplimiento**

* **Product Owner (PO)**: Revisión y validación del trabajo completado en cada sprint, asegurando que se cumpla con los objetivos de valor del usuario final.
* **Scrum Master (SM)**: Supervisión del progreso de tareas diarias y facilitación de reuniones de seguimiento; eliminación de bloqueos y optimización del flujo de trabajo.
* **Equipo de Desarrollo**: Cumplimiento de las tareas asignadas con calidad y eficiencia; colaboración para resolver problemas técnicos y asegurar la integración de cada funcionalidad.

# **Criterios de Finalización del Sprint**

Los **Criterios de Finalización del Sprint** en **FlashMatch** definen las condiciones necesarias para que una tarea o historia de usuario sea considerada como completada al final de cada sprint. Estos criterios aseguran que todas las tareas no solo se implementen técnicamente, sino que también cumplan con los estándares de calidad, funcionalidad y usabilidad requeridos para el proyecto. A continuación, se describen los criterios específicos de finalización, así como el proceso de verificación para confirmar que cada tarea cumple con los requisitos establecidos.

# **Criterios para Considerar las Tareas como Completadas**

1. **Cumplimiento de Criterios de Aceptación**
   * Cada historia de usuario debe cumplir con los criterios de aceptación definidos, que describen los resultados específicos y esperados para que la funcionalidad sea útil y efectiva para el usuario.
   * Ejemplo: Una historia de usuario para el "Registro de Usuario" debe permitir la creación de una cuenta mediante un formulario validado, enviar un correo de confirmación y mostrar mensajes de error si los datos ingresados son incorrectos.
2. **Pruebas Unitarias Completadas**
   * Cada componente o módulo desarrollado debe pasar pruebas unitarias que validen la funcionalidad interna de cada elemento.
   * El código debe tener al menos un 80% de cobertura en pruebas unitarias para garantizar que las funcionalidades principales están bien protegidas contra errores.
3. **Pruebas de Integración Exitosas**
   * Las funcionalidades que dependen de otros módulos deben pasar pruebas de integración para asegurar una correcta comunicación entre componentes.
   * Esto garantiza que las funciones implementadas puedan trabajar en conjunto y se minimicen errores en el proceso de integración de la plataforma.
4. **Aprobación del Product Owner**
   * Al final del sprint, el Product Owner revisa cada historia de usuario y valida que cumpla con los criterios de aceptación y expectativas del usuario final.
   * La aprobación formal del Product Owner es necesaria para cerrar las tareas de cada sprint, asegurando que se cumplen las necesidades de los usuarios.
5. **Pruebas de Usabilidad y Funcionalidad**
   * Las interfaces y funcionalidades deben pasar pruebas de usabilidad que aseguren que los usuarios pueden interactuar con ellas de manera intuitiva y que funcionan como se espera en diferentes dispositivos.
   * Estas pruebas se realizan para garantizar que la experiencia del usuario es satisfactoria y que las funcionalidades son accesibles y comprensibles.
6. **Documentación Completa**
   * Cada funcionalidad implementada debe estar documentada, incluyendo instrucciones de uso y, si es necesario, detalles técnicos de implementación.
   * La documentación asegura que cualquier miembro del equipo o nuevo desarrollador pueda comprender rápidamente el funcionamiento y propósito de cada módulo.

# **Verificación de Requisitos y Estándares de Calidad**

Para verificar que cada tarea cumple con los requisitos y estándares de calidad definidos, se sigue un proceso estructurado al final de cada sprint:

1. Revisión de Criterios de Aceptación:
   * El equipo revisa los criterios de aceptación para cada historia de usuario y verifica que todas las funcionalidades cumplen con estos requisitos.
   * Se realizan demostraciones para que el Product Owner valide visualmente que las funciones son correctas.
2. Análisis de Resultados de Pruebas:
   * Los resultados de las pruebas unitarias y de integración se analizan para confirmar que los componentes funcionan correctamente tanto de manera independiente como en conjunto.
   * Cualquier fallo detectado se soluciona antes de considerar la tarea como finalizada.
3. Revisión de Calidad de Código y Usabilidad:
   * El equipo revisa la calidad del código, asegurando que sigue los estándares definidos, es legible y está optimizado.
   * Las funcionalidades se prueban en distintos dispositivos y navegadores, verificando que el diseño y usabilidad cumplen con las expectativas.
4. Aprobación y Retroalimentación del Product Owner:
   * El Product Owner da su aprobación final, validando que las tareas cumplen con los requerimientos del usuario y aportan el valor esperado.
   * Si alguna funcionalidad no cumple con las expectativas, se registran ajustes necesarios que se programan para el siguiente sprint.

# **Seguimiento y Avance Diario (Daily Scrum)**

El **Daily Scrum** es una reunión breve y estructurada que se realiza todos los días durante cada sprint de **FlashMatch**. Su propósito es sincronizar el trabajo del equipo, revisar el progreso de las tareas, y detectar cualquier impedimento que pueda afectar el cumplimiento de los objetivos del sprint. La reunión tiene una duración máxima de 15 minutos y se enfoca en mantener al equipo alineado y en movimiento hacia la entrega de valor incremental al final de cada iteración.

**Dinámica Diaria para Revisar el Avance**

Durante el Daily Scrum, cada miembro del equipo responde a tres preguntas clave:

1. **¿Qué hice ayer?**
   * Cada miembro informa sobre las tareas completadas el día anterior, compartiendo el progreso realizado en las historias de usuario o tareas asignadas.
   * Esta revisión permite que todos en el equipo estén al tanto del avance colectivo y se adapten a cambios en las prioridades si es necesario.
2. **¿Qué voy a hacer hoy?**
   * Cada miembro comparte las tareas que planea completar en el día, incluyendo cualquier actividad necesaria para avanzar en las historias de usuario en el Sprint Backlog.
   * Esto permite que todos los miembros tengan claridad sobre el enfoque diario y que se ajusten prioridades para optimizar el avance.
3. **¿Tengo algún impedimento?**
   * Los miembros mencionan cualquier problema o bloqueo que esté dificultando su progreso.
   * Esto permite identificar rápidamente obstáculos que requieren atención inmediata para evitar que retrasen el avance general del sprint.

# **Procedimiento para Identificar y Abordar Impedimentos**

Para mantener la agilidad del equipo y minimizar los retrasos, el Daily Scrum incluye un procedimiento claro para identificar y resolver impedimentos:

1. **Detección Temprana de Impedimentos**:
   * Cualquier miembro que enfrente un bloqueo lo menciona durante el Daily Scrum para que el equipo esté al tanto y pueda buscar soluciones.
   * Los impedimentos pueden incluir problemas técnicos, dependencias de otros miembros o equipos, falta de recursos, o cualquier otra barrera al cumplimiento de tareas.
2. **Intervención del Scrum Master**:
   * El Scrum Master es responsable de abordar y eliminar los impedimentos mencionados.
   * Puede intervenir de inmediato o después del Daily Scrum, dependiendo de la naturaleza del bloqueo, para asegurar que el equipo pueda continuar sin interrupciones.
3. **Soluciones Colaborativas**:
   * Si el impedimento requiere experiencia adicional, el equipo puede colaborar para buscar soluciones, asignar recursos adicionales o reconfigurar el plan de trabajo diario.
   * En caso de problemas técnicos complejos, se pueden organizar reuniones adicionales o sesiones de trabajo en conjunto para resolverlos rápidamente.
4. **Documentación y Seguimiento**:
   * El Scrum Master documenta los impedimentos y su resolución para realizar un seguimiento en caso de que se presenten problemas similares en el futuro.
   * Esta documentación también sirve como referencia para mejorar el proceso de trabajo y anticipar bloqueos en los próximos sprints.
5. **Evaluación del Impacto en el Sprint**:
   * Si el impedimento podría afectar el cumplimiento de las tareas críticas del sprint, el equipo analiza si se requiere una replanificación de tareas o el ajuste de prioridades.
   * Esto asegura que los objetivos del sprint se mantengan en foco y que cualquier retraso sea gestionado de manera ágil.

# **Gráficos de Progreso del Sprint**

Los gráficos de progreso del sprint, como el Burn-down Chart y el Burn-up Chart, son herramientas visuales fundamentales en FlashMatch para monitorear el avance del sprint y evaluar el cumplimiento de los objetivos en tiempo real. Estos gráficos permiten al equipo y al Product Owner visualizar rápidamente el estado de las tareas, identificar posibles retrasos y ajustar el trabajo según sea necesario para asegurar que el sprint se complete a tiempo.

**1. Burn-down Chart**

El **Burn-down Chart** muestra la cantidad de trabajo restante en el sprint en comparación con el tiempo restante. En este gráfico, el eje vertical representa las tareas pendientes (medidas en horas o puntos de historia), y el eje horizontal representa los días del sprint.

* **Interpretación**:
  + Una línea descendente constante indica que el equipo está completando las tareas según lo planeado.
  + Si la línea se mantiene horizontal o desciende lentamente, esto puede señalar bloqueos o retrasos en la finalización de las tareas.
  + Un descenso brusco al final del sprint puede indicar un esfuerzo concentrado para alcanzar los objetivos, lo que puede afectar la calidad y sostenibilidad del trabajo.
* **Beneficios**:
  + Permite ver de un vistazo si el equipo avanza de manera uniforme y si es probable que logre completar las tareas del sprint a tiempo.
  + Ayuda a detectar rápidamente si las tareas se están acumulando, lo que permite al Scrum Master intervenir para resolver posibles impedimentos.

**2. Burn-up Chart**

El **Burn-up Chart** es una alternativa que muestra el progreso del trabajo completado a lo largo del sprint. En este gráfico, el eje vertical representa el total de trabajo completado (medido en puntos de historia o tareas), y el eje horizontal representa el tiempo transcurrido.

* **Interpretación**:
  + Una línea ascendente que se aproxima al objetivo total de trabajo indica que el equipo está completando tareas según el plan.
  + Si la línea de progreso se estanca en un punto, esto puede indicar problemas en la ejecución de tareas específicas o un bloqueo en el flujo de trabajo.
  + Una curva ascendente constante es ideal, ya que muestra un avance continuo hacia el objetivo.
* **Beneficios**:
  + Muestra claramente cuánto trabajo ha sido completado y cuánto queda por hacer.
  + Permite ver la velocidad del equipo (o "velocity") y, a su vez, ajustarse a la capacidad de entrega para sprints futuros.

# 

# **Análisis Visual del Cumplimiento de los Objetivos en Tiempo Real**

Ambos gráficos permiten realizar un análisis visual en tiempo real, ayudando al equipo a adaptarse rápidamente y tomar decisiones informadas:

1. **Detección de Desviaciones en el Progreso**:
   * Al monitorear los gráficos diariamente, el equipo puede detectar desviaciones a tiempo, como un ritmo de trabajo más lento de lo esperado o tareas que están tomando más tiempo del previsto.
2. **Ajuste de Prioridades y Replanificación**:
   * Si el Burn-down o Burn-up Chart indica un retraso significativo, el equipo puede evaluar las tareas pendientes, priorizar las más críticas y, si es necesario, ajustar el alcance del sprint para garantizar la entrega de valor.
3. **Optimización del Flujo de Trabajo**:
   * El análisis visual del progreso permite identificar patrones de trabajo, como periodos de baja productividad, que pueden abordarse para mejorar el flujo en los siguientes sprints.
4. **Transparencia y Comunicación**:
   * Los gráficos facilitan la comunicación clara con el Product Owner y otros stakeholders, mostrando de forma visual el estado del sprint y el avance hacia los objetivos. Esto mejora la transparencia y la confianza en el equipo.

# 

# **Retrospectiva del Sprint**

La **Retrospectiva del Sprint** es una reunión clave en **FlashMatch** que se lleva a cabo al final de cada sprint. Su propósito es evaluar el desempeño del equipo, identificar qué aspectos del proceso funcionaron bien y detectar áreas de mejora. A través de esta reflexión, el equipo puede desarrollar propuestas de acción para optimizar su flujo de trabajo y mejorar la calidad y eficiencia en futuros sprints. La retrospectiva fomenta un enfoque de mejora continua, esencial para mantener la agilidad y adaptabilidad del equipo.

## **Evaluación de lo que Funcionó**

1. **Cumplimiento de Tareas en Tiempo y Forma**:
   * Analizar si el equipo completó las tareas asignadas de acuerdo al cronograma planificado y revisar qué factores contribuyeron al cumplimiento de los objetivos del sprint.
   * Identificar las herramientas, prácticas o estrategias que facilitaron el avance constante.
2. **Comunicación y Colaboración Efectiva**:
   * Evaluar la efectividad de la comunicación interna entre los miembros del equipo y con el Product Owner.
   * Identificar si las reuniones de Daily Scrum y otros puntos de contacto ayudaron a mantener a todos alineados y resolver problemas rápidamente.
3. **Resolución Ágil de Impedimentos**:
   * Revisar los casos en los que el equipo pudo superar bloqueos o impedimentos de manera rápida, asegurando que el trabajo no se detuviera.
   * Analizar cómo el Scrum Master facilitó la eliminación de obstáculos y promovió un flujo de trabajo continuo.

## 

## **Áreas a Mejorar**

1. **Estimación de Tiempo y Recursos**:
   * Revisar las tareas que tomaron más tiempo del estimado o que requirieron recursos adicionales.
   * Identificar patrones en las estimaciones que podrían ajustarse para reflejar mejor la realidad de las tareas, mejorando la precisión en futuros sprints.
2. **Gestión de Imprevistos**:
   * Evaluar cómo el equipo manejó los imprevistos o cambios en los requisitos durante el sprint.
   * Identificar si fue necesario ajustar el alcance del sprint y cómo esto afectó la entrega de valor.
3. **Optimización del Daily Scrum**:
   * Reflexionar sobre la efectividad de las reuniones diarias y si cumplieron con su propósito de revisar el avance y abordar impedimentos.
   * Considerar ajustes en la dinámica del Daily Scrum para mejorar la eficiencia y el enfoque en las tareas más críticas.

# **Propuestas de Acción para Mejorar en Futuros Sprints**

1. **Revisión y Ajuste de Estimaciones**:
   * Implementar una breve revisión de estimaciones al inicio de cada sprint para aprender de sprints anteriores y ajustar el tiempo asignado a tareas de similar complejidad.
   * Aplicar técnicas de estimación colaborativa para mejorar la precisión y acordar el esfuerzo necesario.
2. **Planificación de Capacidades y Repriorización**:
   * Al inicio de cada sprint, definir las capacidades del equipo de acuerdo con los aprendizajes recientes y los recursos disponibles.
   * Priorizar tareas críticas o de alto valor para asegurar que, ante cualquier imprevisto, las funcionalidades más importantes se completen a tiempo.
3. **Mejorar la Documentación de Impedimentos y Soluciones**:
   * Documentar los impedimentos encontrados y las soluciones implementadas para crear una referencia futura, lo cual puede ayudar a identificar y anticipar bloqueos similares.
   * Implementar una sesión breve al final de cada sprint para discutir las mejores prácticas de resolución de problemas.
4. **Feedback Continuo del Product Owner**:
   * Aumentar la frecuencia de la comunicación con el Product Owner durante el sprint para obtener feedback temprano y evitar cambios de último momento.
   * Facilitar sesiones de revisión iterativa en las que el Product Owner valide avances parciales y contribuya a mejorar la funcionalidad antes de la entrega final.
5. **Refinamiento del Daily Scrum**:
   * Ajustar la duración o estructura del Daily Scrum si se detecta que algunos temas requieren más enfoque. Esto permite que el equipo se centre en áreas clave sin desviarse.
   * Asegurar que todos los miembros vayan preparados para maximizar la eficiencia y el valor de cada reunión.

# **Conclusión del Sprint**

La **Conclusión del Sprint** en **FlashMatch** permite realizar un cierre formal de la iteración, donde el equipo evalúa los logros y las metas alcanzadas, así como las lecciones aprendidas. Esta sección es clave para reflejar el avance hacia los objetivos del proyecto, resaltar las áreas exitosas y definir oportunidades de mejora que se aplicarán en el próximo sprint.

# Resumen de los Logros y Metas Alcanzadas

1. **Cumplimiento de Historias de Usuario**:
   * Todas las historias de usuario planificadas para el sprint fueron completadas, cumpliendo con los criterios de aceptación y entregando un incremento de valor a la plataforma.
   * Las funcionalidades desarrolladas fueron revisadas y aprobadas por el Product Owner, confirmando que cumplen con las expectativas del usuario final.
2. **Eficiencia en la Ejecución de Tareas**:
   * El equipo trabajó de manera colaborativa y logró completar las tareas asignadas dentro del tiempo estimado.
   * Los gráficos de progreso (Burn-down y Burn-up charts) muestran que el equipo mantuvo un ritmo constante, permitiendo un avance fluido y bien organizado.
3. **Resolución Ágil de Impedimentos**:
   * Los obstáculos encontrados durante el sprint fueron resueltos de manera rápida y eficaz, lo que ayudó a mantener el ritmo de trabajo.
   * La intervención del Scrum Master y la colaboración del equipo fueron esenciales para superar estos bloqueos sin afectar significativamente el cronograma.

# **Reflexión sobre el Cumplimiento de los Objetivos**

1. **Cumplimiento de Objetivos del Sprint**:
   * Los objetivos establecidos para este sprint se lograron en su totalidad, entregando el valor esperado al usuario y al proyecto.
   * La funcionalidad entregada permite avanzar en el desarrollo de la plataforma, añadiendo incrementos que son fundamentales para el éxito del proyecto.
2. **Identificación de Oportunidades de Mejora**:
   * La retrospectiva del sprint destacó algunas áreas de mejora, como el ajuste en las estimaciones de tareas para reflejar mejor el esfuerzo necesario.
   * La comunicación durante el Daily Scrum puede mejorarse para optimizar el tiempo de la reunión y asegurar que se enfoquen en los temas críticos.

# Oportunidades de Mejora para el Próximo Sprint

1. **Ajuste de Estimaciones y Priorización de Tareas**:
   * Revisar las estimaciones iniciales basadas en los aprendizajes de este sprint y ajustar los puntos de historia de tareas similares en futuros sprints.
   * Reforzar la priorización de tareas según su impacto y valor para asegurar que se trabajen primero las funcionalidades de mayor prioridad.
2. **Optimización del Daily Scrum**:
   * Implementar ajustes en el Daily Scrum para asegurar que se mantenga un enfoque en los obstáculos críticos y se fomente una comunicación más efectiva.
   * Fomentar que cada miembro llegue preparado con su actualización para hacer las reuniones más dinámicas y productivas.
3. **Documentación de Soluciones y Buenas Prácticas**:
   * Formalizar la documentación de las soluciones implementadas para los impedimentos encontrados, creando una referencia para el equipo que facilite la resolución de problemas similares en el futuro.
   * Identificar y compartir las buenas prácticas que surgieron durante este sprint para replicarlas en los próximos.

# **Conclusión General del Sprint**

La conclusión de cada sprint en **FlashMatch** permite al equipo hacer una pausa y reflexionar sobre su desempeño, destacando tanto los logros como las áreas a mejorar. Los logros obtenidos en este sprint fortalecen el proyecto y acercan al equipo a los objetivos de la plataforma, mientras que las oportunidades de mejora proporcionan una guía clara para optimizar los próximos sprints. Este proceso de reflexión y adaptación constante asegura que el equipo mantenga un enfoque en la entrega de valor incremental y en la mejora continua, pilares fundamentales en la metodología ágil.