

BORROO

PLANIFICACIÓN DEL DESARROLLO



Devising a project (DP) – G4

Repositorio: <https://github.com/ISPP-2425-G4/>

20/02/2025

Miembros:

- David Blanco Mora
- Pablo Díaz Ordóñez
- Pablo Espinosa Naranjo
- Jesús Fernández Rodríguez
- Francisco Fernández Mota
- Javier García Rodríguez
- Miguel González Ortiz
- Álvaro Martín Muñoz
- Ignacio Naredo Bernardos
- Javier Nieto Vicioso
- Marco Padilla Gómez
- Miguel Palomo García
- Luis Javier Periañez Franco
- Alexander Picón Garrote
- Santiago Rosado Raya
- Julia Sánchez Márquez
- Alejandro Sevillano Barea

Índice

Planificación del desarrollo	2
Sprint 1 (Semanas 5 y 6)	2
Distribución del equipo Sprint 1	2
Backlog del Sprint 1	2
Calendario del Sprint 1 (2 semanas vista)	3
Definición de Hecho	5
Sprint 2 (Semanas 7 y 8)	5
Distribución del equipo Sprint 2	6
Backlog del Sprint 2	6
Sprint 3 (Semanas 9 y 10).....	8
Distribución del equipo Sprint 3	8
Backlog del Sprint 3	9

Planificación del desarrollo

Para poder llevar a cabo el desarrollo de la aplicación usaremos la metodología ágil Scrum. Para ello, vamos a dividir los casos de uso de la aplicación y todas las funcionalidades a implementar durante los tres sprints que se llevarán a cabo, atendiendo primero a los casos de uso core del MVP y, posteriormente al resto. En cada sprint, se llevará a cabo una designación de los responsables de cada tarea, quienes se encargarán de completar cada una de las funcionalidades.

Sprint 1 (Semanas 5 y 6)

Objetivo: Crear un prototipo funcional con los casos de uso principales (core) del MVP.

Durante el sprint 1, llevaremos a cabo el desarrollo de los casos de uso Core del producto mínimo viable (MVP):

- Listar objetos.
- Poner en alquiler un objeto.
- Alquilar un objeto.
- CRUD de objetos.

Despliegue técnico y pruebas:

- Infraestructura inicial en **GitHub + GitHub Project**.
- Despliegue del prototipo en **Google Cloud (App Engine)**.
- Inicio del plan de pilotaje (piloting plan) con usuarios seleccionados.

Distribución del equipo Sprint 1

El equipo del proyecto estará dividido en 3 subgrupos, cada uno de los cuales estarán encargados de casos de uso concretos.

Subgrupo	Responsabilidades
1	Listado de objetos
2	Puesta en alquiler y alquilar objetos
3	CRUD de objetos

Cada subgrupo trabajará en backend y frontend, asegurando integración completa.

Backlog del Sprint 1

1. Diagrama UML

- Desarrollar el diagrama UML con todas las entidades y relaciones.

2. Listar objetos (Subgrupo 1)

- Implementar listado de objetos con filtros (categoría, disponibilidad, fecha de disponibilidad, opción de “por empresas”, ...)

- Agregar paginación y ordenación en las consultas.
- Diseñar y desarrollar la interfaz de lista de objetos.
- Implementar barra de búsqueda y filtros.

3. Poner en alquiler un objeto (Subgrupo 2)

- Implementar endpoint para poner en alquiler un objeto.
- Formulario de publicación de objeto.
- Crear pantalla de confirmación y resumen del alquiler.

4. Alquilar un objeto (Subgrupo 2)

- Implementar endpoint para alquilar un objeto.
- Crear pantalla de confirmación y resumen del objeto alquilado y coste total.

5. CRUD de objetos (Subgrupo 3)

- Crear un objeto
- Actualizar un objeto
- Mostrar los detalles de un objeto
- Eliminar un objeto

Calendario del Sprint 1 (2 semanas vista)

Atendiendo a una duración de algo más de dos semanas en total, un primer vistazo a una planificación a dos semanas vista de la parte de desarrollo sería:

Día 1: Kickoff del Sprint

- Presentación general del sprint.
- Definición clara de tareas y objetivos.
- Explicación de dependencias entre subgrupos.
- Configuración del entorno de desarrollo (repositorio, GitHub Project, Google Cloud).
- Discusión sobre estándares de codificación y metodologías de trabajo.

Día 2-3: Diseño de UML y modelado de base de datos

- **Todos los subgrupos:**
 - Diagramas de clases para los modelos de datos.
 - Diagramas de secuencia para flujos principales.
 - Diagramas de casos de uso para cada funcionalidad.

- Validación y revisión en equipo de los modelos.

Día 4-5: Desarrollo inicial del backend y conexión con frontend

Subgrupo 1

- Implementación de endpoints y vistas iniciales.
- Implementar el listado y paginación de productos.

Subgrupo 2

- Implementación de endpoints y vistas iniciales.
- Desarrollo del formulario para publicación de objetos.
- Desarrollo de vista de confirmación y resumen del objeto en alquiler.

Subgrupo 3

- Implementación de endpoints y vistas iniciales.
- Create y Read de objeto.

Día 6-7: Desarrollo de funcionalidades avanzadas

Subgrupo 1

- Implementación de endpoints.
- Implementar el motor de búsqueda y filtrado.

Subgrupo 2

- Implementación de endpoints.
- Desarrollo de vista de confirmación y resumen y coste del objeto alquilado.

Subgrupo 3

- Implementación de endpoints.
- Delete y Update de objeto.

Día 8: Integración y pruebas unitarias

Todos los subgrupos:

- Pruebas unitarias en backend.
- Integración entre backend y frontend.

Día 9: Pruebas funcionales y corrección de errores

Todos los subgrupos:

- Pruebas de conexión entre frontend y backend.
- Corrección de errores detectados en integración.

Día 10: Revisión final y feedback

Todos los subgrupos:

- Revisión del código y feedback del equipo.
- Presentación del Sprint y ajustes finales.

Definición de Hecho

Para que una tarea se considere completa:

- Código subido y revisado en el repositorio.
- Pruebas unitarias implementadas.
- Integración entre backend y frontend validada.
- Revisión y feedback del equipo realizados.

Tener también en cuenta que se desarrollarán reuniones de seguimiento de proyecto, documentando todas las decisiones en la wiki de este. Cada subgrupo hará revisión de código entre pares antes de subir los cambios.

Sprint 2 (Semanas 7 y 8)

Objetivo: Completar la funcionalidad del MVP agregando administración, seguridad, pagos y nuevas funcionalidades clave.

Casos de uso principales:

- Sistema de pago
- Sistema de administración
- Registro e inicio de sesión
- Sistema de notificaciones
- Sistema de chat y mensajería
- Sistema de reseñas
- CRUD usuarios

Despliegue técnico y pruebas:

- Recepción de **feedback del piloto** para mejoras.
- Infraestructura en **GitHub + GitHub Project**.
- Despliegue en **Google Cloud (App Engine)**.

Distribución del equipo Sprint 2

Subgrupo	Responsabilidades
1	Sistema de pagos y depósitos, autenticación de usuarios, sistema de reseñas y calificaciones
2	Sistema de mensajería interna, gestión de contenido (moderación)
3	Sistema de notificaciones, CRUD de usuarios y objetos

Backlog del Sprint 2

1. Autenticación de usuarios (Subgrupo 1)

- **Backend:**
 - Crear el modelo Usuario en la base de datos.
 - Implementar endpoints de **registro e inicio de sesión** con validación de datos.
 - Manejo de sesiones y tokens de autenticación (JWT).
- **Frontend:**
 - Diseñar pantallas de **registro e inicio de sesión**.
 - Implementar formularios con validaciones de datos.
 - Conectar con la API y almacenar tokens de autenticación.

2. Sistema de pagos y depósitos (Subgrupo 1)

- Un usuario puede realizar un pago por el alquiler mediante pasarela de pago integrada.
- Se retiene un depósito, que se libera al finalizar el alquiler.

- **Backend:**
 - Implementación de **Stripe o PayPal** como pasarela de pago.
 - Creación del modelo Transacción para registrar los pagos.
 - Endpoint para procesar pagos y gestionar depósitos retenidos.
 - Validaciones de seguridad en las transacciones.
- **Frontend:**
 - Interfaz de usuario para **realizar pagos y ver confirmaciones**.
 - Implementación de alertas para pagos exitosos o fallidos.
 - Integración con la API de pagos.

3. Sistema de mensajería interna (Subgrupo 2)

- **Backend:**
 - Creación del modelo **Mensaje** para almacenar conversaciones.
 - Implementación de **WebSockets** para chat en tiempo real.
 - Endpoint para enviar y recibir mensajes.
- **Frontend:**
 - Diseño e implementación de chat entre usuarios.

- Integración de notificaciones de mensajes nuevos.

4. Sistema de notificaciones (Subgrupo 2)

- **Backend:**
 - Creación del modelo **Notificación** para almacenar alertas.
 - Implementación de **WebSockets** o **Firebase** para **notificaciones en tiempo real**.
 - Endpoint para gestionar preferencias de notificación.
- **Frontend:**
 - Interfaz para **visualizar notificaciones** y ajustar preferencias.
 - Implementación de **notificaciones push y alertas en la app**.

5. Sistema de reseñas y calificaciones (Subgrupo 3)

- Un usuario puede calificar a otro usuario tras completar un alquiler.
- Cada usuario tendrá un promedio de calificaciones visible en su perfil.
- Se podrán dejar comentarios adicionales junto con la calificación.

- **Backend:**
 - Creación del modelo **Calificación**
 - Endpoint para registrar calificaciones después de un alquiler.
 - Endpoint para obtener el promedio de calificaciones de un usuario.
 - Validación para que solo usuarios con alquiler finalizado puedan calificar.
 - Protección contra spam y reseñas falsas.
- **Frontend:**
 - Interfaz en el perfil de cada usuario con:
 - Promedio de calificaciones (estrellas y número total de reseñas).
 - Listado de reseñas con comentarios recientes.
 - Formulario para calificar a otro usuario tras finalizar el alquiler.
 - Restricción para que solo usuarios que han alquilado puedan calificar.

Funcionalidades para Administración

6. Gestión de contenido (Subgrupo 3)

- Administración puede eliminar publicaciones o reseñas que incumplan las normas.
- Moderación de comentarios y calificaciones en objetos.

- **Backend:**
 - Implementación de **roles de administración** con permisos específicos.

- Endpoint para **gestionar publicaciones, reseñas y comentarios**.
- Lógica para **suspender o bloquear usuarios infractores**.
- **Frontend:**
 - Panel de administración con listado de **reseñas y publicaciones**.
 - Opciones para **eliminar contenido ofensivo** o marcarlo para revisión.

7. CRUD de Usuarios y Objetos (Subgrupo 1, 2 y 3)

- Administración podrá **crear, editar, eliminar y suspender** usuarios.
- Administración podrá **gestionar los objetos en alquiler** (editar, eliminar, suspender).
- **Backend:**
 - Creación de endpoints para **gestión de usuarios** (CRUD completo).
 - Creación de endpoints para **gestión de objetos en alquiler**.
 - Validaciones de **roles y permisos** para restringir acceso.
- **Frontend:**
 - Panel de administración con **listado de usuarios y objetos**.
 - Opciones para **editar y suspender cuentas o anuncios**.

Sprint 3 (Semanas 9 y 10)

Objetivo: Refinamiento, optimización y prototipo final.

Casos de uso principales:

- Editar el perfil de usuario
- Gestión de suscripción premium
- Anunciar la necesidad de un alquiler

Despliegue técnico y pruebas:

- Estrategia inicial de marketing y comunicación.
- Infraestructura en GitHub + GitHub Project.
- Despliegue en Google Cloud (App Engine).

Distribución del equipo Sprint 3

Subgrupo	Responsabilidades
1	Subscripción premium, anunciar la necesidad de un alquiler, mejorar las funcionalidades anteriores y pruebas
2	Algoritmo de recomendación, mejorar las funcionalidades anteriores y pruebas

3	Edición de perfil de usuario, anunciar la necesidad de un alquiler, mejorar las funcionalidades anteriores y pruebas
---	--

Backlog del Sprint 3

1. Suscripción premium (Subgrupo 1)

- Los usuarios podrán acceder a beneficios exclusivos mediante una suscripción mensual.
- Se ofrecerán ventajas como mayor visibilidad de anuncios, reducción de comisiones y soporte prioritario.

- **Backend:**

- Implementación de Stripe o PayPal para pagos de suscripción.
- Creación del modelo **Suscripción** para gestionar planes y vencimientos.
- Endpoint para activar, renovar y cancelar suscripciones.
- Validaciones de acceso a funcionalidades premium.

- **Frontend:**

- Interfaz para elegir y gestionar planes de suscripción.
- Implementación de alertas y confirmaciones de pago.
- Etiquetas y distinción visual para usuarios premium.

2. Algoritmo de recomendación (Subgrupo 2)

- Se mostrarán sugerencias personalizadas de objetos en alquiler basadas en el comportamiento del usuario.

- **Backend:**

- Implementación de un modelo de recomendación basado en historial de búsquedas y alquileres.
- Endpoint para obtener recomendaciones personalizadas.
- Optimización de consultas a la base de datos para mejorar el rendimiento.

- **Frontend:**

- Sección de recomendaciones dentro de la aplicación.

3. Editar perfil de usuario (Subgrupo 3)

- Los usuarios podrán editar sus datos personales y preferencias.

- **Backend:**

- Implementación de endpoints para actualizar información del usuario (nombre, dirección, foto de perfil, etc.).
- Validaciones para asegurar la integridad de los datos.

- **Frontend:**

- Interfaz para editar la información del perfil.
- Implementación de validaciones en el formulario.
- Visualización de cambios en tiempo real tras la edición del perfil.

4. Anunciar la necesidad de un alquiler (Subgrupo 3):

- **Backend:**
 - Implementar endpoint para que los usuarios puedan publicar solicitudes de alquiler.
 - Validaciones para evitar solicitudes duplicadas o sin datos esenciales.
 - Endpoint para listar las solicitudes de alquiler activas.
- **Frontend:**
 - Crear formulario para que los usuarios anuncien qué objeto necesitan alquilar.
 - Diseñar pantalla con listado de solicitudes activas.
 - Implementar filtros para encontrar solicitudes según tipo de objeto y ubicación.

Pruebas de la aplicación

Cada grupo realizará una serie de pruebas (unitarias, de carga, regresión...) para comprobar el correcto funcionamiento de la aplicación.

Subgrupo 1 (Gestión de usuarios y administración):

- Pruebas de registro, inicio de sesión, suscripción premium, seguridad de autenticación, y usabilidad en el proceso de registro.
- Validación de integración entre autenticación y pagos.

Subgrupo 2 (Gestión de objetos, alquileres, mensajes y notificaciones):

- Validación de publicación de objetos, alquiler, mensajes, notificaciones en tiempo real, y pruebas de compatibilidad en dispositivos.
- Comprobación de seguridad en mensajes y protección de datos.

Subgrupo 3 (Gestión de alquileres, reseñas y administración):

- Pruebas de calificaciones, reseñas, moderación de contenido, y la funcionalidad de administración.
- Evaluación de rendimiento en el manejo de reseñas y validación de seguridad.