

DUIT

R1L2

Documento de plan de Proyecto

Autores:

Aleixo Fernández Cuevas
Cristo Manuel Navarro Martín

Nombre del fichero:

DAW_PRW_R1L2_UT01.0 – Documento de plan de proyecto

Fecha de esta versión:

01/02/2026

Ciclo Formativo:

Desarrollo de Aplicaciones Web – Semipresencial (DAWN)

Módulo:

Proyecto de Desarrollo de Aplicaciones Web (PRW)

Historial de revisiones

Fecha	Descripción	Autores
21/12/2025	Fase de análisis. Versión inicial	Aleixo F. Cuevas / Cristo N. Martín
10/01/2026	Primera corrección	Aleixo F. Cuevas / Cristo N. Martín
01/02/2026	Segunda corrección	Aleixo F. Cuevas / Cristo N. Martín

Índice

1. Resumen ejecutivo.....	4
2. Alcance del proyecto	4
2.1 Objetivo Principal.....	4
2.2 Objetivos Específicos.....	4
2.3. Alcance Incluido.....	4
2.4. Alcance Excluido	6
3. Requisitos del proyecto	6
3.1 Requisitos funcionales.....	6
3.2 Requisitos no funcionales.....	6
3.3 Requisitos técnicos.....	6
4. Estructura de trabajo y cronograma.....	7
5. Recursos y roles del equipo.....	7
5.1 Recursos humanos.....	7
5.2 Recursos técnicos.....	7
6. Gestión de riesgos.....	8
6.1. Plan de Mitigación.....	9
7. Conclusión	10
8. Reparto de tareas y responsabilidades.....	10

1. Resumen ejecutivo

Duit es una plataforma web que conecta a usuarios que necesitan servicios profesionales con profesionales cualificados que pueden ofrecerlos. La aplicación gestiona el ciclo completo del servicio: desde la publicación de solicitudes, postulaciones de profesionales, contratación, ejecución del trabajo hasta la valoración final bidireccional.

El proyecto aplica conocimientos del ciclo formativo de Desarrollo de Aplicaciones Web en un caso real, utilizando tecnologías modernas como Spring Boot 3, Java 21, PostgreSQL y desplegando en infraestructura cloud.

2. Alcance del proyecto

2.1 Objetivo Principal

Desarrollar una plataforma web funcional, segura y escalable que facilite la conexión entre clientes y profesionales de servicios, implementando el ciclo completo de gestión de servicios con sistema de valoraciones y transparencia.

2.2 Objetivos Específicos

1. **Objetivo Técnico:** Implementar una arquitectura MVC robusta con Spring Boot, JPA/Hibernate y PostgreSQL, siguiendo buenas prácticas de desarrollo.
2. **Objetivo Funcional:** Desarrollar todas las funcionalidades core del sistema (gestión de usuarios, solicitudes, postulaciones, trabajos y valoraciones).
3. **Objetivo de Seguridad:** Garantizar la protección de datos mediante Spring Security, encriptación BCrypt y control de acceso basado en roles.
4. **Objetivo de Usabilidad:** Crear una interfaz responsive e intuitiva con Thymeleaf y Bootstrap que funcione en todos los dispositivos.
5. **Objetivo Académico:** Aplicar los conocimientos adquiridos en el ciclo formativo en un proyecto integral y profesional.
6. **Objetivo de Despliegue:** Publicar la aplicación en producción con infraestructura cloud (Koyeb + Neon) accesible públicamente.

2.3. Alcance Incluido

Módulo de Usuarios

- Registro y autenticación
- Perfiles de usuario y profesional
- Sistema de roles (Cliente, Profesional, Administrador)
- Gestión de direcciones

Módulo de Solicitudes

- Creación y publicación de solicitudes
- Categorización de servicios
- Estados del ciclo de vida (Borrador, Publicada, En progreso, Completada, Cancelada, Expirada)
- Gestión completa CRUD

Módulo de Postulaciones

- Búsqueda de solicitudes por profesionales
- Postulación con precio y mensaje
- Aceptación/rechazo por parte del cliente
- Gestión de estados (Pendiente, Aceptada, Rechazada, Retirada)
- Gestión completa CRUD

Módulo de Trabajos

- Creación automática al aceptar postulación
- Seguimiento de trabajos activos
- Actualización de estados
- Vinculación con solicitud y postulación
- Gestión completa CRUD

Módulo de Valoraciones

- Sistema bidireccional (cliente ↔ profesional)
- Puntuación de 1 a 5 estrellas
- Comentarios opcionales
- Estados (Pendiente, Publicada, Oculta)
- Gestión completa CRUD

Panel de Administración

- Gestión de usuarios (CRUD)
- Gestión de categorías (CRUD)
- Visualización de estadísticas
- Moderación de valoraciones

Seguridad y Auditoría

- Control de acceso basado en roles
- Auditoría automática de cambios (created_by, created_date, last_modified_by, last_modified_date)
- Registro de accesos al sistema
- Encriptación de contraseñas con BCrypt

2.4. Alcance Excluido

- ❖ Sistema de pagos integrado (Stripe, PayPal)
- ❖ Chat en tiempo real o mensajería instantánea
- ❖ Aplicación móvil nativa (iOS/Android)
- ❖ Geolocalización con mapas interactivos
- ❖ Notificaciones push
- ❖ Validación de cuenta mediante correo electrónico de verificación

3. Requisitos del proyecto

3.1 Requisitos funcionales

- Registro y autenticación de usuarios.
- Creación y gestión de solicitudes.
- Postulación de profesionales.
- Gestión completa del trabajo.
- Valoración tras la finalización del servicio.

3.2 Requisitos no funcionales

- Seguridad de los datos.
- Accesibilidad desde distintos dispositivos.
- Rendimiento adecuado.
- Código mantenible y documentado.

3.3 Requisitos técnicos

- Backend en Java 21.
- Spring Boot 3.x.
- Base de datos PostgreSQL.
- Arquitectura MVC.
- Despliegue en infraestructura cloud.

4. Estructura de trabajo y cronograma

El proyecto **Duit** se ha desarrollado siguiendo la metodología **Kanban**, basada en la gestión visual del trabajo y en un flujo continuo de tareas.

Fase	Nombre	Descripción	Herramientas	Periodo
1	Análisis y planificación	Definición de alcance, requisitos y riesgos	Trello	Dic 2025
2	Diseño funcional y técnico	Prototipos y estructura del sistema	Figma	Dic 2025 Ene 2026
3	Desarrollo backend	Lógica de negocio y datos	VSC + Git	Ene 2026
4	Desarrollo frontend	Vistas y experiencia de usuario	VSC + Git	Ene 2026 Feb 2026
5	Pruebas y validación	Verificación funcional	Pruebas manuales	Feb 2026
6	Documentación y entrega	Memoria y entrega final	OnlyOffice / Markdown	Feb 2026

5. Recursos y roles del equipo

5.1 Recursos humanos

El proyecto es desarrollado por un equipo de dos desarrolladores con perfil Full Stack, que participan activamente en todas las fases del proyecto.

Recurso	Rol	Responsabilidades	Dedicación	Periodo
Aleixo Fernández Cuevas	Full Stack Developer	Arquitectura, backend, seguridad, despliegue y documentación	100%	Sept 2025 – Mar 2026
Cristo Manuel Navarro Martín	Full Stack Developer	Frontend, integración, apoyo backend y pruebas	100%	Sept 2025 – Mar 2026

5.2 Recursos técnicos

Herramientas de desarrollo

- VSC
- Git y GitHub
- PostgreSQL
- DBeaver
- Figma
- Trello

Infraestructura

- Hosting: Koyeb (Free Tier)
- Base de datos: Neon (Free Tier)

6. Gestión de riesgos

Riesgos Técnicos

ID	Riesgo	Probabilidad	Impacto	Severidad
RT-01	Problemas de compatibilidad entre versiones de librerías	Media	Medio	MEDIA
RT-02	Bugs críticos en producción	Media	Alto	ALTA
RT-03	Caída de servicios cloud gratuitos	Media	Alto	ALTA
RT-04	Vulnerabilidades de seguridad no detectadas	Baja	Alto	MEDIA
RT-05	Pérdida de datos por fallo en BD	Baja	Crítico	MEDIA

Riesgos de Proyecto

ID	Riesgo	Probabilidad	Impacto	Severidad
RP-01	Retraso en el cronograma	Media	Medio	MEDIA
RP-02	Scope creep (ampliación incontrolada de funcionalidades)	Alta	Medio	ALTA
RP-03	Falta de tiempo para testing exhaustivo	Alta	Medio	ALTA
RP-04	Cambios en requisitos a mitad de desarrollo	Media	Medio	MEDIA
RP-05	Problemas de salud o disponibilidad del desarrollador	Baja	Alto	MEDIA

Riesgos Externos

ID	Riesgo	Probabilidad	Impacto	Severidad
RE-01	Cambios en políticas de servicios cloud	Baja	Medio	BAJA
RE-02	Deprecación de tecnologías utilizadas	Baja	Medio	BAJA
RE-03	Limitaciones de tier gratuito superadas	Media	Medio	MEDIA

6.1. Plan de Mitigación

RT-01: Problemas de compatibilidad

- **Prevención:** Usar Spring Initializr para garantizar compatibilidad inicial
- **Mitigación:** Mantener dependencias actualizadas, revisar breaking changes
- **Contingencia:** Tener versiones de respaldo documentadas

RT-02: Bugs críticos en producción

- **Prevención:** Testing exhaustivo antes de despliegue
- **Mitigación:** Sistema de rollback rápido, logs detallados
- **Contingencia:** Hotfix inmediato, comunicación a usuarios

RT-03: Caída de servicios cloud

- **Prevención:** Backups regulares de BD
- **Mitigación:** Monitoreo de uptime, plan de migración documentado
- **Contingencia:** Servicios alternativos identificados (Render, Railway)

RT-04: Vulnerabilidades de seguridad

- **Prevención:** Usar frameworks actualizados, validaciones en servidor
- **Mitigación:** Revisión de seguridad, dependencias actualizadas
- **Contingencia:** Parche de seguridad urgente

RT-05: Pérdida de datos

- **Prevención:** Backups automáticos de Neon
- **Mitigación:** Backups manuales periódicos
- **Contingencia:** Restauración desde último backup

RP-02: Scope creep

- **Prevención:** Documento de alcance claro, priorización estricta
- **Mitigación:** Revisión semanal de scope, registro de cambios
- **Contingencia:** Funcionalidades adicionales movidas a roadmap futuro

RP-03: Falta de tiempo para testing

- **Prevención:** Testing continuo durante desarrollo
- **Mitigación:** Priorización de tests críticos
- **Contingencia:** Testing post-entrega, versión 1.1

7. Conclusión

Este documento de plan de proyecto describe los objetivos, el alcance, los requisitos y la planificación del desarrollo de la aplicación web **Duit**. Su objetivo es servir como guía durante el desarrollo del proyecto y ayudar a organizar el trabajo a lo largo del módulo de Proyecto Integrado del ciclo formativo de Desarrollo de Aplicaciones Web.

Definir desde el inicio los roles, las funcionalidades y los límites del sistema permite evitar errores y problemas durante el desarrollo, además de facilitar una mejor organización de las tareas. El uso de herramientas de planificación y diseño (Trello, Figma, Git) ha ayudado a que el proyecto avance de forma más ordenada, aunque no formen parte directa de las funcionalidades de la aplicación.

8. Reparto de tareas y responsabilidades

La elaboración del presente Documento de Plan de Proyecto se ha realizado de forma colaborativa por los dos miembros del equipo, participando ambos en la definición, redacción y revisión de su contenido.

- **Aleixo Fernández Cuevas:**
Elaboración del primer borrador del documento, definición de la estructura general, redacción de los apartados de alcance, planificación, descripción del sistema y revisión final del contenido, incorporando correcciones y mejoras.
- **Cristo Manuel Navarro Martín:**
Colaboración en la redacción de los apartados técnicos, especialmente los relacionados con seguridad, requisitos técnicos y gestión de riesgos, así como revisión del documento y aportación de mejoras sobre el contenido inicial.