



Proyecto de Desarrollo de Aplicaciones Web (PRW) / Desarrollo de Aplicaciones Web

Área de Informática y Nuevas Tecnologías

Ficha Técnica

SmartLining

[Cod]. SMTLNG

Nombre del fichero:	DAW_PRW_[SMTLNG]_1.4. Ficha técnica.pdf
----------------------------	---

Fecha de esta versión:	11/01/2024
-------------------------------	------------

Historial de revisiones

Fecha	Descripción	Autor
18/01/2026	Confección y subida del documento	Aitor Aridane Peña Sánchez

ÍNDICE

1 INTRODUCCIÓN.....	5
2 REQUISITOS TÉCNICOS.....	5
2.1 Plataforma y herramientas de desarrollo software.....	5
2.2 Plataforma de Ejecución.....	7
3 Bases de Datos.....	7
4 Interfaces externas.....	8
5 Seguridad.....	8
6 Control de versiones.....	8
7 Observaciones.....	8

1 INTRODUCCIÓN

Este documento, es parte de la fase de diseño del proyecto y complementa la documentación funcional y el diseño del modelo de datos previamente definidos. Su finalidad es servir como referencia y justificación técnica tanto para el desarrollo del sistema como para exponer las decisiones técnicas adoptadas, asegurando la coherencia entre los objetivos del proyecto, los requisitos funcionales y la arquitectura tecnológica seleccionada.

2 REQUISITOS TÉCNICOS

2.1 Plataforma y herramientas de desarrollo software

2.1.1. Requisitos front-end

A continuación se enumeran las diferentes tecnologías de desarrollo front-end, versiones y herramientas de apoyo al desarrollo:

I) Tecnologías / framework

HTML5 | v5

Lenguaje de marcas utilizado para la estructura semántica de la aplicación web.

CSS3 | v3

Lenguaje de estilos empleado para la presentación visual, y junto a html5 constituyen la maquetación y el diseño del sistema.

JavaScript | ES2025

Lenguaje base para la interacción dinámica en el navegador.

TypeScript | v5.x

Lenguaje tipado utilizado para mejorar la mantenibilidad, robustez y escalabilidad del código front-end.

React | v19.x

Framework de desarrollo front-end basado en componentes, utilizado para la creación de interfaces dinámicas y reactivas, especialmente adecuado para la visualización en tiempo real del estado de las colas.

II) Otras Herramientas de desarrollo, depuración, etc

npm | v11.x

Vite | v7.x

ESLint | v9.x

Prettier | v3.x

Visual Studio Code | v1.108+

2.1.2. Requisitos back-end

A continuación se enumeran las diferentes tecnologías de desarrollo back-end, versiones y herramientas de apoyo al desarrollo: [Sustituya lo indicado entre corchetes por la información adecuada]

I) Plataforma de desarrollo [Node.js]

a) Framework, runtime y versiones

Node.js | v25.x

Entorno de ejecución JavaScript del lado del servidor.

Express.js | v5.x

Framework web minimalista utilizado para el desarrollo de la API REST del sistema.

TypeScript | v5.x

Lenguaje principal del backend, proporcionando tipado estático y mayor control del código.

b) Esquema de desarrollo

El backend se desarrolla siguiendo un esquema modular inspirado en el patrón Modelo-Vista-Controlador (MVC), adaptado al uso de Express:

- Controladores para la gestión de peticiones HTTP.
- Servicios para la lógica de negocio.
- Repositorios para el acceso a datos.
- Rutas para la definición de endpoints REST.

Este enfoque favorece la separación de responsabilidades y la escalabilidad del sistema.

c) IDE

Visual Studio Code | v1.85+, con extensiones para TypeScript, Git y depuración.

d) Otras dependencias

Prisma | v7.2 – ORM para la gestión de la base de datos relacional

mysql2 | v3.x – conector de la aplicación con la base de datos MySQL

dotenv | v17.x – gestión de variables de entorno

cors | v2.x – control de acceso entre dominios

jsonwebtoken | v9.x – autenticación basada en JWT

bcrypt | v6.x – cifrado de contraseñas

2.2 Plataforma de Ejecución

2.2.1. Servidor

A continuación se indican las características técnicas de mínimos para la ejecución de la aplicación descrita en el proyecto:

Servidor web y versión:

Node.js con servidor HTTP integrado mediante Express.

Sistema operativo y versión:

Ubuntu Server 22.04 LTS para despliegues. MacOS en local.

Runtime / Intérprete y versión:

Node.js v25.x LTS

Observaciones a la configuración:

La aplicación se ejecuta inicialmente en entorno local sobre MacOS, utilizando variables de entorno para la gestión de la configuración sensible (credenciales de base de datos, puertos de escucha y claves de autenticación).

La arquitectura del sistema está diseñada para permitir su despliegue futuro en entornos de pre-producción y producción sobre servidores VPS (IONOS u otros proveedores equivalentes), sin requerir modificaciones estructurales en el código.

2.2.2. Alojamiento físico/ Entornos de ejecución

A continuación, se indican los directorios donde se ubicarán los ficheros de la aplicación desplegada:

Desarrollo:

Servidor local con ejecución de la aplicación mediante Node.js y base de datos MySQL en entorno local.

Pre-Producción / Producción:

En caso de despliegue este sería en un servidor VPS con Ubuntu Server 22.04 LTS.

3 Bases de Datos

Base de datos: MySQL

Versión: 8.0.x

Esquema: smartlining

La base de datos relacional permite implementar el modelo entidad-relación definido y garantiza la integridad de los datos necesarios para el análisis de afluencia y tiempos de atención.

4 Interfaces externas

La aplicación se comunica mediante los WebServices creados en el front. Estos son utilizados por el frontend para acceder a los datos del sistema.

5 Seguridad

Las principales medidas de seguridad implementadas o previstas son:

- Autenticación mediante JWT para usuarios empleados y administradores.
- Separación de roles (cliente, empleado, administrador).
- Cifrado de contraseñas mediante bcrypt.
- Uso de variables de entorno para datos sensibles.
- Validación de entradas en la API y control de CORS.

6 Control de versiones

Se utiliza **Git** como sistema de control de versiones.

- Repositorio alojado en GitHub.

7 Observaciones

El diseño técnico descrito en este documento junto con los documentos de alcance funcional y diseño del modelo de datos del proyecto dan lugar a SmartLining. El stack de tecnologico seleccionado nos permite desarrollar una aplicación web totalmente escalable, mantenible y alineada con los objetivos analíticos y operativos definidos.