



XUNTA  
DE GALICIA

CONSELLERÍA DE CULTURA,  
EDUCACIÓN, FORMACIÓN  
PROFESIONAL E UNIVERSIDADES



IES SAN MAMEDE

Rúa do Castelo N° 3. 32700. Maceda (Ourense)

Tlf: 988 788 561 ✉ [ies.san.mamede@edu.xunta.gal](mailto:ies.san.mamede@edu.xunta.gal)

🌐 <http://www.iesanmamede.com>

# Saborify

<https://github.com/RodrigoGB12/ProyectoFinalCurso>

**Nome Alumno/a:** Rodrigo Gajardo Blanco

**Curso:** 2º DAM

**Módulo:** *Proxecto Final Ciclo*



## Contido

<b>Contido</b>	<b>2</b>
<b>1. Introducción</b>	<b>3</b>
<b>2. Obxectivos</b>	<b>3</b>
Objetivos alcanzados:	4
<b>3. Situación previa</b>	<b>4</b>
<b>4. Tecnoloxías empregadas</b>	<b>4</b>
<b>5. Solución proposta</b>	<b>4</b>
<b>6. Planificación do proxecto</b>	<b>4</b>
<b>7. Desenvolvemento e execución</b>	<b>5</b>
<b>8. Conclusións e reflexións</b>	<b>5</b>
<b>9. Bibliografía e Webgrafía</b>	<b>5</b>
<b>10. Anexo I – Manual técnico de instalación ou posta en marcha</b>	<b>5</b>
<b>11. Anexo II – Documentación de uso (manuais usuario)</b>	<b>5</b>
<b>12. Anexo III – Outra documentación</b>	<b>5</b>

## 1. Introducción

El proyecto elegido fue “Saborify” una app de catering donde los usuarios podran consultar sus pedidos y crear nuevos viendo asi el catalogo completo y los administradores podran ver los datos de todos los clientes, platos, pedidos etc.

## 2. Obxectivos

### Objetivos centrados en el usuario y el problema a resolver

- **Facilitar la gestión de pedidos y menús:** Permitir que los administradores de catering organicen los pedidos de manera eficiente sin depender de procesos manuales.
- **Reducir errores en la planificación:** Automatizar el registro y modificación de pedidos para evitar confusiones en la gestión de eventos.
- **Mejorar la comunicación con los clientes:** Proporcionar un sistema en el que los clientes puedan consultar menús, realizar pedidos y hacer modificaciones de manera sencilla.
- **Asegurar el acceso y la seguridad de la información:** Implementar un sistema de autenticación que garantice que solo los usuarios autorizados puedan gestionar la información del servicio de catering.
- **Optimizar el tiempo y los recursos:** Minimizar las tareas repetitivas y manuales, permitiendo a los administradores enfocarse en mejorar la experiencia del cliente.
- **Hacer accesible la digitalización del servicio de catering:** Ofrecer una plataforma económica y adaptable que pueda ser utilizada por pequeños y medianos negocios sin altos costos de inversión.

1.

## Tecnologías Utilizadas

## 3. Situación previa

En la actualidad, la gestión eficiente de los servicios de catering es fundamental para garantizar la calidad y satisfacción de los clientes. Con el crecimiento del sector gastronómico y el aumento de la demanda de eventos personalizados, surge la necesidad de contar con herramientas digitales que faciliten la administración de pedidos, menús y clientes.

Tradicionalmente, la gestión de estos servicios se ha llevado a cabo mediante procesos manuales o con herramientas básicas como hojas de cálculo y sistemas no especializados. Esto genera ineficiencias en la organización, retrasos en los pedidos y una mayor probabilidad de errores en la planificación de eventos y control de inventarios. Además, la falta de una plataforma centralizada dificulta la comunicación entre clientes, proveedores y administradores, lo que puede afectar la calidad del servicio ofrecido.

Ante esta problemática, muchas empresas de catering buscan soluciones tecnológicas que permitan automatizar estos procesos, optimizar recursos y mejorar la experiencia tanto para los clientes como para los gestores del negocio. Sin embargo, las opciones comerciales disponibles suelen ser costosas o poco adaptables a las necesidades específicas de cada empresa.

## 4. Tecnoloxías empregadas

### Backend (API REST - Java)

- **Java 17:** Lenguaje de programación moderno que ofrece mejoras en rendimiento, seguridad y funcionalidad.
- **Spring Boot:** Framework de Java que simplifica el desarrollo de aplicaciones web al proporcionar configuraciones predeterminadas y una integración sencilla con otros componentes del ecosistema Spring.
- **Spring Data JPA:** Biblioteca que facilita la interacción con bases de datos mediante la abstracción de consultas SQL y la integración con ORM como Hibernate.
- **Hibernate:** Framework ORM que permite mapear objetos de Java a tablas de bases de datos, reduciendo la complejidad en la gestión de datos.
- **H2:** Base de datos ligera y embebida utilizada para entornos de desarrollo, ideal para pruebas rápidas sin necesidad de configuración avanzada.
- **JWT (JSON Web Token):** Mecanismo de autenticación basado en tokens que permite gestionar sesiones de usuario de manera segura y sin necesidad de almacenar datos de sesión en el servidor.
- **Swagger:** Herramienta que facilita la documentación y prueba de APIs REST, proporcionando una interfaz interactiva para explorar y probar endpoints.

### Frontend (Angular)

- **Angular 17:** Framework moderno para el desarrollo de aplicaciones web SPA (Single Page Applications) que permite crear interfaces dinámicas y modulares con una estructura mantenible.
- **TypeScript:** Lenguaje de programación basado en JavaScript que añade tipado estático, mejorando la robustez y mantenibilidad del código.
- **RxJS:** Biblioteca para la programación reactiva que permite manejar eventos y flujos de datos asíncronos de manera eficiente.
- **Bootstrap:** Framework de diseño frontend que proporciona una colección de componentes y estilos predefinidos para desarrollar interfaces modernas y responsivas con facilidad.

## 5. Solución proposta

Este proyecto busca ofrecer una solución integral a través de una aplicación web compuesta por un backend desarrollado en Java con Spring Boot y un frontend en Angular. La API proporciona funcionalidades para la gestión de usuarios, autenticación, administración de menús y control de pedidos, asegurando un sistema robusto y escalable. Gracias al uso de tecnologías modernas como JWT para la seguridad, MySQL para la persistencia de datos y Swagger para la documentación, la aplicación garantiza un alto rendimiento y facilidad de mantenimiento.

El frontend, construido con Angular y Angular Material, ofrece una interfaz intuitiva y atractiva para que los usuarios puedan interactuar de manera eficiente con la plataforma. Utilizando Bootstrap y RxJS, se logra una experiencia fluida y dinámica, adaptada a las necesidades de los clientes y administradores del servicio de catering.

Con este proyecto, se pretende simplificar la gestión de los servicios de catering, mejorando la organización y optimización de los procesos internos. Además, se busca proporcionar una herramienta adaptable y de fácil implementación para pequeñas y medianas empresas del sector.

## 6. Planificación do proxecto

- Planificación do proxecto cun diagrama de Gantt
- Descripción do modelo de desenvolvemento de software a implementar.
- Análise do proxecto:
  - Diagrama de casos de usos.
  - Universo de discurso da base de datos.
  - Diagrama de Entidad-Relación.
- Deseño do proxecto:
  - Modelo relacional da base de datos.
  - Diagrama de clases.
  - Diagramas de fluxo de cada caso de uso.
  - Mockups da interface.
- Presupuesto completo (Hardware software e recursos humanos)

Concepto	Especificaciones/Requerimentos	Costo Aproximado
<b>Hardware</b>		
Procesador	Intel Core i5 / AMD Ryzen 5 (o superior)	250€
Memoria RAM	8 GB (recomendado 16 GB)	80€ - 150€
Almacenamiento	256 GB SSD (recomendado 512 GB SSD o superior)	60€ - 120€
Placa Base	Compatible con procesador y memoria RAM	100€ - 200€
Conectividad	Acceso a internet estable	20€/mes
<b>Software</b>		
Sistema Operativo	Windows 10/11, macOS, Linux (Ubuntu recomendado)	Gratuito
IDE Backend	IntelliJ IDEA Community Edition o VS Code	Gratuito
IDE Frontend	Visual Studio Code con extensiones de Angular	Gratuito
Java 17	OpenJDK	Gratuito
Spring Boot	Framework de código abierto	Gratuito
Node.js y npm	Para gestión de dependencias de Angular	Gratuito
Angular CLI	Para la creación y administración del frontend	Gratuito

## 7. Desenvolvemento e execución

Diagrama de despregamento e descripción xeral do funcionamento da aplicación.

Completar cos detalles técnicos que fose preciso resolver documentados ao longo do desenvolvemento.

## 8. Conclusións e reflexións

Valoración global do proxecto, resume do aprendido e das dificultades atopadas.

## 9. Bibliografía e Webgrafía

[Formato APA](#)



## 10. Anexo I – Manual técnico de instalación ou posta en marcha

## 11. Anexo II – Documentación de uso (manuais usuario)

## 12. Anexo III – Outra documentación

Calquera material adicional relevante (documentación da aplicación, manual de uso, capturas de pantalla, fragmentos de código, etc.).