

**MEMORIA RETO**

**TÍTULO: RETO BURGER JAM - 2025**

**AUTORES:Izan Conejos, Marcos Lahuerta y Alejandro Comenge**

**FECHA: 31/03/2025 - 19/05/2025**

**Curso 2024-2025**

**1º GS Desarrollo de Aplicaciones Web**

### 

[**INGLÉS**](#_ei5r6c86ybkt) **3**

[**IPE I**](#_uof8v968r1t4) **3**

[**LENGUAJE DE MARCAS**](#_crawzkh1534x) **3**

[**BASES DE DATOS**](#_si6213a3yo5x) **4**

[**PROGRAMACIÓN**](#_lae0qf9vu85a) **4**

[**ENTORNOS DE DESARROLLO /NUBE**](#_j89y7zn9zepu) **4**

[**DIGITALIZACIÓN**](#_xxk25u8tf2vb) **5**

[**Fortalezas (Factores Internos Positivos)**](#_6ta590l0g1r5) **6**

[**Debilidades (Factores Internos Negativos)**](#_vveayn33hx3b) **6**

[**Oportunidades (Factores Externos Positivos)**](#_7tdgec9u5lu) **6**

[**Amenazas (Factores Externos Negativos)**](#_uk3mxgsh57uh) **7**

[**1. Desarrollo del Frontend**](#_wixl7hmwvdts) **8**

[**2. Presentación Final**](#_po7gag67l2hw) **8**

[**3. Módulo de Empleo y Vacante**](#_fajwwfzb3f1a) **8**

[**4. Gestión de la Base de Datos**](#_xis84wfyiwcp) **9**

[**5. Arquitectura y Backend**](#_ps208x278mci) **9**

[**6. Uso de Git y GitFlow**](#_usth0tblg6pp) **9**

[**7. Contenerización y Despliegue**](#_etkmmoqovjn9) **9**

[**8. Seguridad y Protección de Datos**](#_ahv8nh4tc89d) **10**

[**9. Plan de Digitalización**](#_wl9idcsvz5et) **10**

[**10. Gestión y Seguimiento del Proyecto**](#_otbp3eo1hind) **10**

[**SISTEMAS INFORMÁTICOS**](#_o5545qqam67) **10**

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### **INGLÉS**

En esta asignatura, lo que hicimos fue elaborar el contenido del proceso de staff y preparamos una presentación final sobre nuestro establecimiento, sabiendo que teníamos que poner las cosas importantes que aparece en nuestra web y podría ayudar a conocernos, la plataforma que utilizamos fue canva ya que es mucho más didáctica y mejor visualmente que otras plataformas.Asimismo, trabajamos los contenidos del apartado frontend que nos basamos en traducir la página web en cuanto a sus apartados como los html, JavaScript y por último los css.

### **IPE I**

Desarrollamos un apartado específico dentro de la web titulado “Trabaja con nosotros”, enfocado a la orientación y búsqueda de empleo. En él incluimos un anuncio de vacante para el perfil de desarrollador de aplicaciones web, detallando los requisitos mínimos exigibles y aspectos a valorar.

### 

### **LENGUAJE DE MARCAS**

La creación de la página web se realizó utilizando html, css y javascript.Estuvimos creando líneas de código en los html para ver cuál encaja más con lo que queríamos que se basara nuestro establecimiento, después para los detalles visuales utilizamos css en el que podíamos determinar muchas opciones que nos gustaban y al final lo pudimos ajustar muy bien.Utilizamos también JavaScript en el que podíamos hacer mucho más didáctico la página web en lo que es esencial.

Definimos roles diferenciados (cliente y trabajador), incluyendo una zona privada con permisos especiales para añadir, editar o eliminar productos o servicios.Utilizamos Git y GitFlow para el control de versiones, lo que nos permitió trabajar en equipo de manera coordinada y organizada.

### 

### **BASES DE DATOS**

Realizamos un estudio detallado de la base de datos, comenzando por el modelo entidad-relación, seguido del modelo relacional y el modelo físico, empleando herramientas adecuadas para su diseño , en este caso usamos draw.io. Documentamos la base de datos con un diccionario de datos y desarrollamos las sentencias SQL necesarias para el soporte de la web. Luego añadimos las tablas create y los insert y comprobamos su funcionamiento en el MySQL Workbench.

### **PROGRAMACIÓN**

En programación empezamos primero intentando identificar que estructura necesitábamos, una vez que sabíamos que estructura crear implementamos el maven e instalamos el tomcat. Los metimos en el intellij y empezamos a crear los diferentes paquetes que necesitábamos, aparte también hicimos un pom.xml al que pusimos las dependencias necesarias. Hicimos una api y nos atascamos a la ahora de que funcionara .

### **ENTORNOS DE DESARROLLO /NUBE**

Implementamos el consumo de APIs en JavaScript mediante Fetch, pero desgraciadamente no nos dio tiempo a terminarla para el lunes 19/05/25, distribuyendo las peticiones para evitar centralización. El despliegue del proyecto se realizó en la nube: el frontend en contenedor, el backend en EC2 y la base de datos en RDS, utilizando además un nombre DNS personalizado para la presentación final.

Lo que nos permitió abrir la página web mediante la dirección IP pública el cual modificamos en el apartado de default y DockerFile, aparte tuvimos que meter los html,css y los JavaScript.

### **DIGITALIZACIÓN**

Utilizamos una herramienta digital seleccionada por el equipo para crear un tablero de seguimiento del proyecto, lo que facilitó la organización de tareas. Además, elaboramos un Plan de Digitalización para la empresa, incluyendo un análisis DAFO, definición de objetivos SMART, alineación con los objetivos estratégicos y un plan de implantación, todo orientado a mejorar la eficiencia y competitividad de la empresa mediante la tecnología.

**Dafo**



### **Fortalezas (Factores Internos Positivos)**

* Equipo multidisciplinar con conocimientos en desarrollo web, bases de datos, y despliegue en la nube.
* Uso de tecnologías actuales: HTML5, CSS3, JavaScript, Java (Servlets), Git, Docker, AWS.
* Enfoque realista del proyecto con aplicación práctica en orientación laboral.
* Organización mediante tableros Kanban (Jira, Trello).
* Aplicación de buenas prácticas como GitFlow, arquitectura por capas y seguridad desde el diseño.
* Presentación y documentación en inglés, útil para la empleabilidad.

### **Debilidades (Factores Internos Negativos)**

* Posible falta de experiencia del equipo con herramientas cloud (AWS, RDS, EC2, Docker).
* Riesgo de mala gestión del tiempo y sobrecarga de tareas si no se distribuyen correctamente.
* Complejidad técnica elevada que puede llevar a errores si no se prueban todos los casos posibles.
* Falta de recursos reales para pruebas de rendimiento o seguridad a gran escala.

### **Oportunidades (Factores Externos Positivos)**

* Mejora directa del portafolio profesional de los alumnos.
* Alta demanda laboral de perfiles con conocimientos en tecnologías usadas en el proyecto.
* Posibilidad de reutilizar el proyecto como base para futuros desarrollos personales o empresariales.
* Fomento de soft skills (trabajo en equipo, gestión de proyectos, presentación en público).
* Desarrollo en inglés: ventaja competitiva para empleos internacionales o remotos.

### **Amenazas (Factores Externos Negativos)**

* Cambios imprevistos en los requisitos o el entorno técnico (problemas con AWS, licencias, etc.).
* Posibles errores en el despliegue o funcionamiento en entornos reales (nube, DNS).
* Brechas de seguridad si no se aplican correctamente los principios de "Security by Design".
* Falta de conexión estable entre servicios o problemas en contenedores Docker si no se configuran bien.
* Riesgo de evaluación baja si la presentación final no comunica bien los logros del proyecto.

**SMART**

* ****

### **1. Desarrollo del Frontend**

* **Objetivo SMART:**  
   Diseñar y desarrollar una página web responsive, intuitiva y semánticamente correcta en HTML5 y CSS que incluya roles diferenciados (cliente y trabajador), navegación entre pantallas y una sección “Trabaja con nosotros”, en inglés, antes del 10 de junio de 2025.

### **2. Presentación Final**

* **Objetivo SMART:** Realizar una presentación oral final en inglés que muestre el funcionamiento completo de la web, sin uso de PowerPoint, el día pactado para la defensa del proyecto, cumpliendo con los requisitos funcionales establecidos.

### **3. Módulo de Empleo y Vacante**

* **Objetivo SMART:** Publicar en la sección “Trabaja con nosotros” un anuncio para desarrollador web con requisitos mínimos y criterios de valoración claros, redactado en inglés, antes del 31 de mayo de 2025.

### **4. Gestión de la Base de Datos**

* **Objetivo SMART:** Diseñar e implementar una base de datos relacional para la web, con modelo E/R, modelo lógico y físico, documentación con diccionario de datos y scripts SQL de soporte y CRUD, todo listo para conexión y funcionamiento completo antes del 5 de junio de 2025.

### **5. Arquitectura y Backend**

* **Objetivo SMART:** Desarrollar una arquitectura backend basada en Servlets y DAO con conexión a base de datos desplegada en AWS (RDS), utilizando buenas prácticas y gestión de excepciones, con funcionalidades completas antes del 7 de junio de 2025.

### 

### **6. Uso de Git y GitFlow**

* **Objetivo SMART:** Gestionar el control de versiones y flujos de trabajo del proyecto mediante Git y GitFlow desde el inicio del desarrollo hasta la finalización del proyecto, con commits regulares y ramas estructuradas correctamente.

### **7. Contenerización y Despliegue**

* **Objetivo SMART:** Contenerizar la aplicación (Frontend y Backend) usando Docker, subir las imágenes a Docker Hub y desplegarla en la nube (EC2 para Backend, RDS para BD y contenedor para Frontend), con nombre DNS configurado, todo completado antes del 8 de junio de 2025.

### **8. Seguridad y Protección de Datos**

* **Objetivo SMART:** Aplicar principios de Seguridad by Design e implementar medidas básicas de protección de datos (cifrado, autenticación, prevención de ataques y cumplimiento con GDPR) desde el diseño inicial hasta la entrega final del proyecto.

### **9. Plan de Digitalización**

* **Objetivo SMART:** Elaborar un plan de digitalización que incluya análisis DAFO, definición de objetivos SMART, alineación estratégica, plan de implantación y despliegue, completado y entregado como parte del proyecto antes del 5 de junio de 2025.

### **10. Gestión y Seguimiento del Proyecto**

* **Objetivo SMART:** Utilizar una herramienta de gestión de proyectos (Trello, Jira, etc.) con un tablero compartido para el equipo donde se refleje el avance, tareas y roles, actualizado al menos 3 veces por semana durante toda la duración del proyecto.

### **SISTEMAS INFORMÁTICOS**

### Trabajamos en la contenerización del proyecto con Docker, generando imágenes del frontend (HTML) y, de forma opcional, del backend en Java