
1 Organizar el equipo y entender el reto

- Formar **grupos de 2–3 personas**.
 - Entender que el objetivo es **simular un proceso real de selección de personal** para un puesto de programador/a multiplataforma.
 - El producto final debe demostrar **competencias técnicas + personales**.
-

2 Analizar el problema del supermercado

- El supermercado tiene **4 básculas**.
 - Cada báscula genera **tickets en formato .txt**.
 - Se generan **más de 600 tickets** diarios.
 - Es obligatorio cumplir la normativa **TicketBAI** (digitalización y envío a Ogasuna).
-

3 Leer automáticamente los tickets (.txt)

- Crear un programa que:
 - Lea **todos los tickets** de las carpetas de las básculas.
 - Se ejecute cuando el usuario lo ordene (manual, no automático continuo).
-

4 Guardar los tickets en una base de datos

- Diseñar una **base de datos central**.
- Guardar cada ticket con:
 - Fecha y hora
 - Datos del ticket
 - Vendedor/a
- **Reglas importantes:**
 - Los tickets **NO se pueden borrar nunca**.
 - El vendedor debe poder **modificarse**.
 - Los tickets ya importados no se duplican.

5 Generar XML para TicketBAI

- Por cada ticket (o lote):
 - Crear un **archivo XML válido** según el esquema oficial.
 - Validar el XML.
 - Guardarlo en una **carpeta central** como backup.
 - **Nunca borrar los XML.**
-

6 Enviar los XML a Ogasuna

- Cuando el XML sea correcto:
 - Enviarlo **automáticamente por email** a Ogasuna.
 - Registrar cada envío en una **hoja de cálculo** con:
 - Fecha
 - Hora
 - Nombre del archivo
 - Tamaño del archivo
-

7 Copias de seguridad

- Implementar un sistema de **backups**:
 - Base de datos
 - XML generados
 - La gestión del backup corre por cuenta del equipo.
-

8 Crear la web de visualización

- Desarrollar una **interfaz web** que permita:
 - Ver una **lista de tickets/XML** en una tabla.
 - Al hacer clic, ver el **detalle completo** del ticket.
- Usar código ya existente en:
 - **JavaScript**
 - **Bootstrap**

9 Estadísticas

- Proponer y generar **estadísticas útiles** a partir de los tickets:
 - Ventas por día
 - Por vendedor
 - Por sección, etc.
 - El supermercado deja libertad para decidir cuáles.
-

10 Documentación personal (parte de empleabilidad)

- Preparar y entregar:
 - **Curriculum (CV)** atractivo.
 - **Carta de presentación.**
 - Trabajar la **marca personal**:
 - Actitud
 - Comunicación
 - Trabajo en equipo
-



SEMANA 1 – Análisis y bases del proyecto

Objetivo: tener claro qué vamos a hacer y dejar los cimientos preparados.

✓ Tareas

1. Leer y entender bien el enunciado completo.
2. Repartir roles en el equipo (programación, BBDD, web, documentación).
3. Analizar el formato de los **tickets .txt**.
4. Diseñar:
 - Esquema de la **base de datos**.
 - Estructura de carpetas (básculas, XML, backups).
5. Crear el proyecto base:
 - Entorno de desarrollo.
 - Conexión a la base de datos.
6. Empezar:
 - Lectura de tickets .txt.
 - Inserción básica en la BBDD (sin lógica avanzada aún).
7. Borrador del CV y carta de presentación.

✓ Entrega interna:

- Diagrama de BBDD
- Programa que lea .txt y guarde tickets



SEMANA 2 – Lógica principal y TicketBAI

Objetivo: que el núcleo del sistema funcione.

✓ Tareas

1. Mejorar la importación de tickets:
 - Evitar duplicados.
 - Guardar fecha y hora.
2. Implementar gestión de **vendedores**:
 - Listado de tickets.
 - Cambio de vendedor.
3. Generación de **XML TicketBAI**:
 - Crear XML por ticket o lote.
 - Validarlo con el esquema.
4. Guardar XML en carpeta central (backup).
5. Preparar envío de XML por email (aunque sea simulado).
6. Registrar envíos en una hoja de cálculo.

✓ Entrega interna:

- XML válidos
- Tickets en BBDD correctamente gestionados



SEMANA 3 – Web y estadísticas

Objetivo: visualizar la información de forma clara.

✓ Tareas

1. Crear la **web**:
 - Tabla con lista de tickets/XML.
 - Vista de detalle al hacer clic.
2. Integrar JavaScript y Bootstrap proporcionados.
3. Conectar la web con la base de datos o XML.
4. Implementar **estadísticas**:
 - Ventas por día.
 - Ventas por vendedor.
 - Otras que decidáis.
5. Implementar copias de seguridad.

✓ Entrega interna:

- Web funcional
- Estadísticas visibles



SEMANA 4 – Pulido, documentación y defensa

Objetivo: dejarlo profesional y presentable.

✓ Tareas

1. Revisar errores y mejorar código.
2. Comprobar:
 - Que no se borra nada.
 - Que los backups funcionan.
3. Mejorar la interfaz web.
4. Finalizar:
 - CV
 - Carta de presentación
5. Preparar presentación:
 - Explicar problema → solución → demo.
6. Ensayar defensa oral.

✓ Entrega final:

- Aplicación completa
- Documentación
- Presentación y defensa

18 tik 6 aukerau eta gure proiektuarekin nola egiten dugun.