

Guía Proyecto

Es importante considerar que el/la tutor/a:

- Realizará un seguimiento de la evolución del proyecto a través de Git.
- La extensión mínima de la documentación ha de ser de 30 páginas, sin contar los anexos.
- Si se detecta copia, plagio y/o el uso de IA en la elaboración del proyecto, este será calificado con un 0.

1. Introducción

En esta sección, el alumnado debe abordar los siguientes puntos:

- **Planteamiento del problema y justificación:** Describir claramente el problema a resolver, la relevancia del proyecto y las razones que justifican su desarrollo.
- **Objetivos:** Establecer los objetivos generales y específicos del proyecto, indicando qué se pretende lograr.
- **Diagrama de Gantt:** Incluir un cronograma detallado con las fases del proyecto, sus actividades y la duración estimada.
- **Resumen bilingüe (inglés-español):** Elaborar un resumen del proyecto en ambos idiomas con una extensión mínima de 20 líneas. Debe reflejar el propósito del proyecto, sus objetivos principales y una visión general de la metodología utilizada.

2. Análisis

En esta fase, el alumnado debe profundizar en los aspectos técnicos y conceptuales del proyecto mediante:

- **Requisitos funcionales y no funcionales:** Identificar y detallar las funcionalidades de la aplicación (requisitos funcionales) y las características de calidad (requisitos no funcionales).
- **Estudio del Arte:** Investigar y analizar soluciones existentes relacionadas con el proyecto para contextualizar el trabajo.

- **Justificación de viabilidad:** Evaluar la viabilidad técnica (herramientas y tecnologías disponibles), económica (costes asociados) y operativa (factibilidad de implementación y mantenimiento).

3. Diseño

En esta etapa, el alumnado debe elaborar y documentar los siguientes elementos:

- **Árbol de navegación:** Diseñar un esquema jerárquico que represente la estructura de navegación de la aplicación.
- **Wireframe-Prototipo:** Crear un prototipo visual (boceto) que refleje la interfaz del usuario y la estructura de las principales pantallas de la aplicación.
- **Diagramas BBDD (EER):** Diseñar el diagrama Entidad-Relación Extendido (EER) que represente la estructura de la base de datos.
- **Casos de uso:** Identificar e ilustrar los principales casos de uso de la aplicación, detallando los actores y sus interacciones.
- **Diagramas de clase:** Elaborar los diagramas de clase que representen la estructura del código y las relaciones entre los objetos de la aplicación.

4. Programación e Indicadores de Calidad

En esta sección, el alumnado debe documentar la implementación y los controles de calidad del proyecto mediante:

- **Implementación:** Describir el proceso de desarrollo de la aplicación, detallando el software y las tecnologías utilizadas, la organización del código y los módulos principales. Se debe proporcionar una visión clara de cómo está construida la aplicación y cómo se gestiona su despliegue:
 - **Software y tecnologías utilizadas:** Especificar las **herramientas y tecnologías** empleadas tanto en el desarrollo como en el **despliegue** de la aplicación: Lenguaje/s de programación, Entorno de desarrollo, Control de versiones, Despliegue de la aplicación web (AWS, Servidores, APIs, etc.).
 - **Arquitectura de la aplicación:** Describir el modelo arquitectónico empleado (MVC, MVVM u otros) y justificar su elección.

- **Organización del código y Módulos principales:** Describir la organización del proyecto, mostrando la relación entre los archivos y su función; así como, explicar los componentes clave de la aplicación y cómo interactúan entre sí: Interfaz de usuario, Lógica de negocio, etc.
- **Seguimiento:** Describir el proceso de validación y control de calidad de la aplicación.
 - Se ha de detallar los diferentes **tipos de pruebas** realizadas durante el desarrollo:
 - **Pruebas unitarias:** Validan el comportamiento de funciones o módulos individuales
 - **Pruebas de integración:** Verifican que los distintos módulos de la aplicación (interfaz, API, base de datos) interactúan correctamente.
 - **Pruebas de sistema:** Evalúan la aplicación en su conjunto, incluyendo la interfaz, el rendimiento y el comportamiento en distintos entornos (navegadores y dispositivos).
 - Documentar los **casos de prueba** que verifican las principales funcionalidades de la aplicación, los **resultados obtenidos** y las **correcciones realizadas**, en caso de realizarse. El objetivo es garantizar el correcto funcionamiento de todas las funcionalidades y la estabilidad del sistema.

A continuación, un ejemplo de tabla para organizar estos casos:

ID	Funcionalidad	Descripción del caso de prueba	Entrada	Resultado esperado	Resultado obtenido	Estado
CP-01	Añadir producto	Verificar que se añade un producto a la base de datos	Nombre: "Producto1" Descripción: "Descripción producto"	El producto se guarda y se muestra	El producto se guarda correctamente	<input checked="" type="checkbox"/> Pasado
CP-02	Eliminar producto	Comprobar que un producto se borra de un listado	Hacer clic en el botón "Borrar"	El producto desaparece de la interfaz	Error: No se borra	<input type="checkbox"/> Fallo
CP-03	Validación del formulario	Comprobar la validación de campos vacíos	Dejar campos vacíos	Mostrar un mensaje de error	Se muestra el mensaje de error	<input checked="" type="checkbox"/> Pasado

5. Conclusiones

En esta última sección, el alumnado debe:

- **Resumir los resultados obtenidos** y su alineación con los objetivos planteados.
- **Reflexionar** sobre las **dificultades** encontradas y las **soluciones** adoptadas.
- **Proponer líneas futuras** de mejora o extensión del proyecto.

6. Anexos

*El apartado de **anexos** incluye documentación complementaria que proporciona información adicional y detallada sobre el proyecto. Estos documentos sirven como referencia para facilitar el uso, la comprensión y el mantenimiento de la aplicación.*

En general, los anexos pueden contener manuales, guías técnicas, esquemas u otros recursos relevantes para garantizar una correcta implementación y comprensión del proyecto.

Posibles ejemplos de anexos, serían:

- **Manual de usuario:** Explica cómo utilizar la aplicación de forma clara y accesible para los usuarios finales, proporcionando instrucciones paso a paso, ejemplos prácticos y soluciones a problemas comunes.
- **Manual del desarrollador:** Dirigido a programadores o futuros mantenedores del proyecto. Incluye detalles técnicos, arquitectura del sistema, estructura del código, procedimientos de instalación, configuración y pautas para realizar actualizaciones o mejoras.

CALENDARIO

Fase	Descripción	Fecha
Introducción	<ul style="list-style-type: none"> - Planteamiento del problema y justificación - Objetivos - Diagrama de Gantt - Resumen bilingüe (mín. 20 líneas) 	10 - 16 de marzo
Análisis	<ul style="list-style-type: none"> - Requisitos funcionales y no funcionales - Estudio del Arte - Justificación de viabilidad (técnica, económica y operativa) 	17 - 23 de marzo
Diseño	<ul style="list-style-type: none"> - Árbol de navegación - Wireframe-Prototipo - Diagramas BBDD (EER) - Casos de uso - Diagramas de Clase, etc. 	24 de marzo - 06 de abril
Programación e indicadores de calidad	<ul style="list-style-type: none"> - Implementación 	07 de abril - 25 de mayo
Programación e indicadores de calidad	<ul style="list-style-type: none"> - Seguimiento 	26 de mayo - 04 de junio
Conclusiones		
ENTREGA		04 de junio