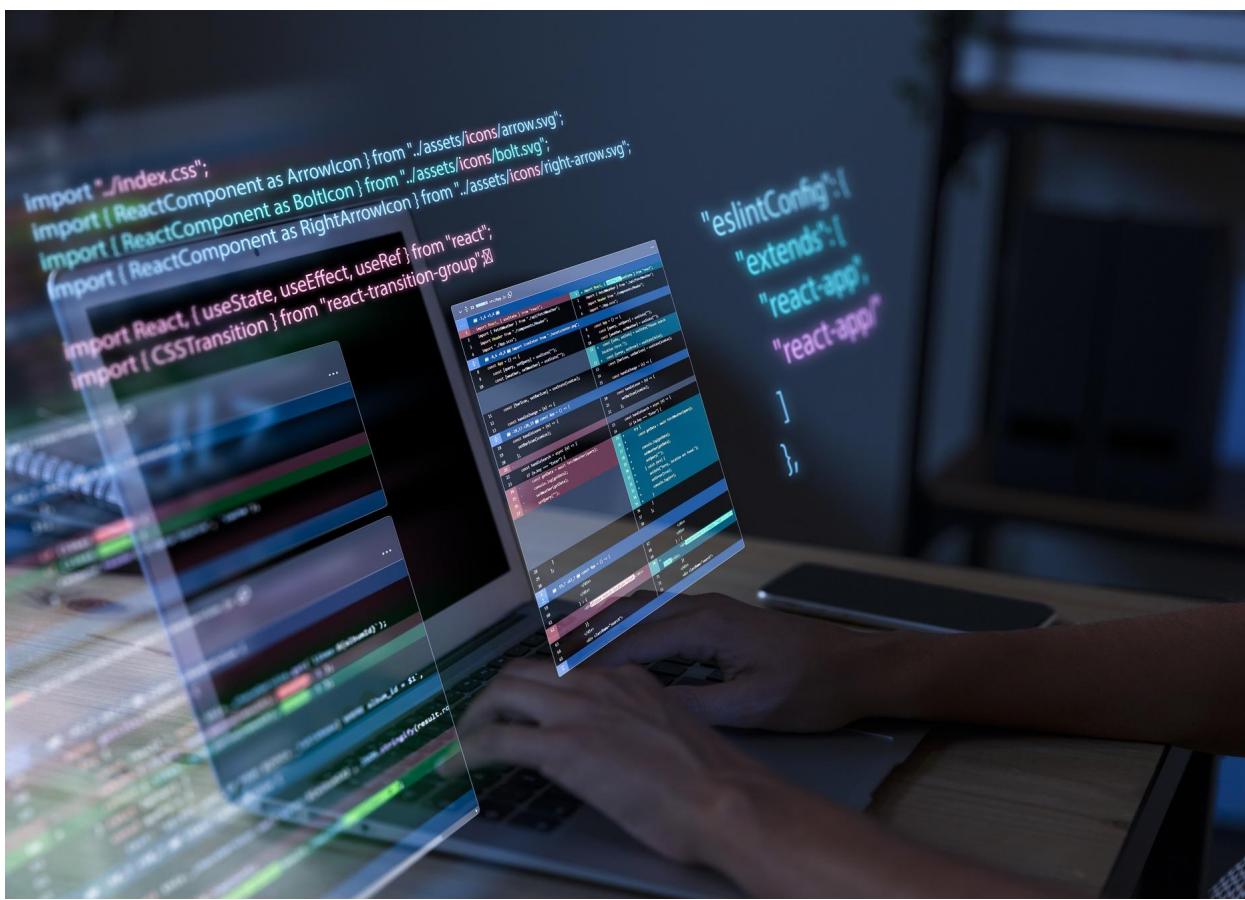


Desarrollo de software

1.4 Desarrollo de aplicaciones



Autor: Lisa ERIKSEN

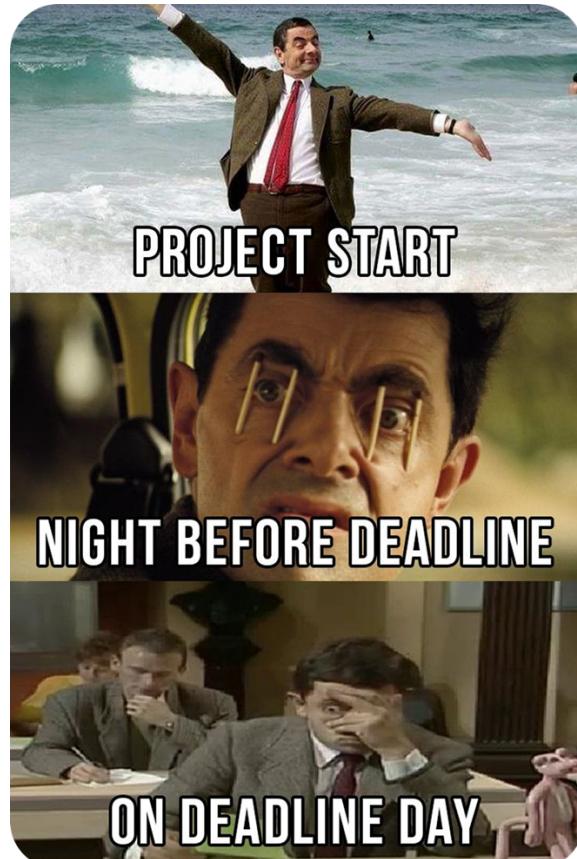
Fecha: 2025 / 2026

1. Objetivo de aprendizaje

- Comprender las **fases del ciclo de vida** del software.
- Identificar y diferenciar los **distintos modelos de desarrollo** de software.
- Aplicar un modelo de desarrollo a un caso práctico.
- Evaluar las ventajas y desventajas de cada modelo de desarrollo.

2. Introducción

El desarrollo de aplicaciones sigue un proceso estructurado conocido como el **ciclo de vida del software**, que abarca desde la identificación de necesidades hasta el mantenimiento del producto final. Este proceso se adapta según la metodología utilizada, lo que influye en la forma de abordar cada fase y la relación con los clientes.



3. Fases en el Desarrollo de una Aplicación

3.1 Análisis

👉 **Actividad:** Ordena las fases del ciclo de vida del software: Lee los siguientes pasos del desarrollo de software, ordénalos correctamente (del 1 al 6) y encuentre su título:

Orden	Título	Descripción
1		
2		
3		
4		
5		
6		

Títulos: Implementación (programación) / Pruebas / Diseño / Despliegue / Análisis de requisitos / Mantenimiento

Descripciones:

- Mantenimiento Resolver errores, actualizar librerías, añadir nuevas funciones y garantizar que el sistema siga siendo útil y seguro.
- Identificar qué se necesita, quién lo pide y con qué restricciones. Aquí se definen funcionalidades, reglas de negocio, limitaciones técnicas y prioridades.
- Diseño Transformar los requisitos en un plan técnico. Incluye la arquitectura del sistema, modelos de datos, diagramas de clases, flujos de usuario y prototipos.
- Escribir el código que materializa el diseño.
- Pruebas Validar que el software funciona como se esperaba y que cumple los requisitos. Se hacen pruebas unitarias, de integración, de aceptación y de seguridad.
- Despliegue Poner el software en un entorno productivo, accesible a los usuarios finales.

3.2 Diseño

Se define cómo será la aplicación. Es como planear una receta y los pasos a seguir. Incluye:

- **Arquitectura del sistema:** Describe la estructura del sistema y las interacciones entre componentes (*cliente, servidor, base de datos y API...*)

Ejemplo: En una aplicación web la arquitectura puede incluir 3 partes



- **Diseño de la base de datos:** Define cómo se organizarán y conectarán los datos.

Ejemplo: Tabla Users en la base de datos:

#	Name	Type
1	id	int(11)
2	username	varchar(255)
3	firstName	varchar(255)
4	lastName	varchar(255)

- **Diseño de la interfaz de usuario:** Cómo interactuarán los usuarios con el sistema.

Ejemplo: La interfaz del **portal de alumnos** permite consultar notas.



👉 **Actividad:** Escoge una idea de aplicación web (o inventa la tuya):

- 🎮 **GameHub** — una app para crear y compartir minijuegos con tus amigos
- 🐾 **PetMatch** — tipo Tinder, pero para adoptar mascotas
- 🍔 **FoodQuest** — una app que te da retos culinarios y te premia por probar nuevos platos
- 🎧 **MoodBeats** — crea listas de música según tu estado de ánimo
- 🌍 **EcoHéroes** — Juego donde ganas puntos por hacer acciones ecológicas (reciclar...)

1. Fase de Análisis

👉 **Preguntas:** Identifica las necesidades, usuarios y funcionalidades principales.

- a. ¿Cómo se llama tu aplicación?
- b. ¿Qué problema o necesidad resuelve tu app? (*descubrir música...*)
- c. ¿Quiénes serán los usuarios principales? (*estudiantes, gamers...*)
- d. ¿Qué tipo de interacciones tendrá el usuario con la app? Describe al menos 3.
- e. ¿Qué dispositivos usará el usuario?
- f. ¿Qué datos necesitará guardar la aplicación?
- g. ¿Qué limitaciones o requisitos técnicos hay que tener en cuenta? (*conexión a Internet, compatibilidad con Android/iOS, privacidad de datos, etc.*)

2. Fase de Diseño

👉 **Preguntas:** Crea el plan técnico de tu aplicación.

- a. Describe la arquitectura del sistema ¿qué partes tendrá?
(*Ejemplo: cliente móvil, servidor, base de datos, API externa...*)
- b. Crea una pequeña tabla de base de datos para tu app (con 3–5 campos). Precisa:
 - El nombre de la tabla
 - Los nombres y los tipos de datos (int, varchar...) de las columnas.
- c. **Reto creativo:** Crea una maqueta sencilla de la página de inicio (Home) de tu app con:
 - El nombre o logo de tu app.
 - Un botón principal (por ejemplo Empezar, Descubrir música...)
 - 2 o 3 elementos visuales clave (menú, imagen, lista, etc.)

Puedes usar: <https://www.figma.com>

3. Fase de codificación

👉 **Actividad:** En este ejercicio vas a escribir un programa de **login** que pida un nombre de usuario y una contraseña para entrar en tu app.



Paso 1 – Crea tu archivo para programar

Crea una carpeta con el nombre de tu app

Abre la carpeta en VS code y crea un nuevo archivo llamado login.py.

Paso 2 – Definir los datos correctos

Define al principio del programa el nombre de usuario y la contraseña “oficiales”.

💡 Pista: usa dos variables en mayúsculas.

Paso 3 – Pedir los datos al usuario

Haz que el programa pregunte el nombre de usuario y la contraseña.

💡 Pistas:

- Usa la función input() para leer texto del usuario.
- Guarda lo que el usuario escribe en dos variables.

Paso 4 – Verificar si son correctos

Comprueba si el usuario y la contraseña coinciden con los valores correctos.

💡 Pistas:

- Usa un if con una condición doble (if ... and ...)

Paso 5 – Añadir intentos limitados

Ahora mejora el programa: el usuario solo puede intentarlo 3 veces.

💡 Pistas:

- Usa una variable intentos = 0.
- Crea un bucle while intentos < 3
- Dentro del bucle, pide los datos y comprueba si son correctos.
- Si son incorrectos, muestra cuántos intentos quedan.
- Si acierta, muestra un mensaje de éxito y usa break para salir del bucle.

Paso 6 – Mensaje final

Cuando el usuario falla 3 veces, muestra un mensaje final: *Cuenta bloqueada. Intenta más tarde.*

4. Fase de implementación

👉 **Preguntas:** Busca tres plataformas gratuitas donde se pueda desplegar (publicar) una aplicación web o móvil.