

Fase de Análisis en el Desarrollo de Software

La **fase de análisis** es una de las más importantes en el ciclo de vida del desarrollo de software. En esta etapa, se estudian **los requisitos del sistema**, se identifican las necesidades de los usuarios y se definen las bases para el diseño y desarrollo del software.

¿En qué consiste la fase de análisis?

1. Recopilación de Requisitos

- Se reúnen los requerimientos del cliente y los usuarios.
- Se identifican funcionalidades esenciales y restricciones.
- Se documentan las expectativas del sistema.

2. Análisis de Requisitos

- Se revisan los requisitos para detectar inconsistencias o ambigüedades.
- Se clasifican en **funcionales** (qué hace el sistema) y **no funcionales** (rendimiento, seguridad, etc.).

3. Modelado del Sistema

Se crean diagramas y modelos para representar la estructura y el comportamiento del sistema, como:

- **Casos de uso** (cómo interactúan los usuarios con el sistema).
- **Diagrama de clases** (estructura del sistema).
- **Diagrama de flujo de datos** (cómo fluye la información).
- **Diagrama de estados** (cómo cambia el sistema según las acciones del usuario).

4. Validación y Verificación

- Se revisan los modelos y requisitos con los interesados para confirmar que reflejan correctamente lo que se necesita.
- Se corrigen errores o malentendidos antes de pasar a la fase de diseño.

¿Por qué es importante la fase de análisis?

- Evita errores costosos en fases posteriores.
- Asegura que el sistema cumpla con las expectativas del cliente.
- Facilita la comunicación entre usuarios, analistas y desarrolladores.

En resumen, la fase de análisis es clave para entender qué debe hacer el software antes de empezar a construirlo. ¡Es como hacer los planos antes de construir una casa!.

Ejemplo Práctico de la Fase de Análisis

Imagina que una empresa quiere desarrollar una **aplicación móvil de pedidos de comida a domicilio** (tipo Uber Eats).

1. Recopilación de Requisitos (¿Qué necesita el cliente?)

El dueño del negocio dice:

- Los clientes deben poder **registrarse y hacer pedidos** desde la app.
- Los restaurantes deben recibir y gestionar los pedidos.
- Los repartidores deben ver qué pedidos deben entregar.
- El administrador debe ver estadísticas de ventas.

Ejemplo de requisitos:

- **Funcionales:** Registro de usuarios, sistema de pago, notificaciones en tiempo real.
- **No funcionales:** La app debe funcionar en iOS y Android, debe ser rápida y segura.

2. Análisis de Requisitos (¿Es viable lo que piden?)

Se revisa si todo está bien definido y si hay dudas:

- ¿Cómo pagarán los clientes? (Tarjeta, PayPal, efectivo)
- ¿Los restaurantes pueden rechazar pedidos?
- ¿Cuánto tiempo tiene un repartidor para aceptar un pedido?

Si hay dudas, se aclaran con el cliente antes de seguir.

3. Modelado del Sistema (Planificación con diagramas y esquemas)

Aquí se crean representaciones visuales para entender cómo funcionará el sistema:

Diagrama de Casos de Uso:

Muestra cómo interactúan los usuarios con el sistema:

- Cliente → **Realizar pedido**
- Restaurante → **Aceptar pedido**
- Repartidor → **Recoger pedido y entregarlo**
- Administrador → **Gestionar informes de ventas**

Fase de análisis

Mg. Hugo Ruiz Martinez

Instructor ADSO

Diagrama de Clases:

Define cómo se organizarán los datos en el sistema:

- **Usuario** (nombre, email, rol: cliente/restaurante/repartidor)
- **Pedido** (número de pedido, cliente, restaurante, estado)
- **Pago** (método de pago, monto, estado)

Diagrama de Flujo de Datos:

Muestra cómo viaja la información:

1. Cliente hace un pedido → 2. Restaurante lo recibe → 3. Se confirma el pago → 4. Repartidor lo entrega.

4. Validación y Verificación (¿Está todo correcto?)

Se revisa con los clientes y usuarios clave para confirmar que:

- Todos los requisitos están bien representados.
- Los diagramas reflejan correctamente lo que se necesita.
- No faltan funciones esenciales.

Si algo no está bien, se hacen cambios antes de pasar a la fase de **diseño y desarrollo**.

Conclusión

La fase de análisis ayuda a planificar bien el sistema antes de programarlo. Es como hacer los planos de una casa antes de construirla: evita errores y asegura que todo funcione correctamente.

Verificación de Modelos Realizados en la Fase de Análisis

Ahora que completamos la fase de análisis con los modelos de la aplicación de pedidos de comida, es momento de **verificarlos** para asegurarnos de que cumplen con los requisitos definidos en el informe.

1. Paso 1: Comparar los Modelos con los Requisitos

Objetivo: Revisar si los diagramas reflejan correctamente lo que se pidió.

Ejemplo de revisión:

- **Caso de Uso:** ¿Incluye todas las acciones del usuario? (Hacer pedido, pagar, recibir pedido).
- **Diagrama de Clases:** ¿Tiene todas las entidades necesarias? (Usuario, Pedido, Restaurante, etc.).
- **Diagrama de Flujo de Datos:** ¿Representa correctamente cómo viaja la información en el sistema?

Si algún requisito no está bien representado, se hacen ajustes en los modelos.

2. Paso 2: Validación con los Interesados

Objetivo: Mostrar los modelos a los clientes o usuarios clave para confirmar que reflejan lo que necesitan.

Ejemplo:

- El dueño del negocio revisa el diagrama de casos de uso y dice: **"Falta una opción para que los clientes puedan cancelar un pedido"** → Se debe agregar esta función en el modelo.
- Un repartidor revisa el flujo de datos y pregunta: **"¿Cómo sé cuándo un pedido está listo para recoger?"** → Se aclara con una notificación en el sistema.

Si se detectan errores o mejoras, se actualizan los modelos antes de seguir a la fase de diseño.

3. Paso 3: Corrección de Errores y Aprobación Final

Objetivo: Hacer ajustes y obtener la aprobación del equipo y los clientes.

Ejemplo:

Fase de análisis

Mg. Hugo Ruiz Martinez

Instructor ADSO

- Se agrega la opción de "Cancelar pedido" en el caso de uso.
- Se actualiza el diagrama de flujo de datos para reflejar las notificaciones de pedido listo.
- Se revisa que la estructura del sistema en el diagrama de clases sea coherente con los requisitos.

Una vez que todo está corregido y validado, se obtiene la aprobación final y se pasa a la **fase de diseño**.

Conclusión

La verificación de modelos asegura que lo analizado en la fase anterior **cumpla con los requisitos antes de avanzar**. Esto evita errores en el desarrollo y garantiza que el sistema funcione como se espera.