

Desarrollo de Software

GUATEMALA, 19 DE MAYO DE 2,018

¿Qué es?

- Secuencia de actividades.
- Conformado por un equipo.
- Apoyado sobre metodologías de desarrollo ágil.
- El objetivo es generar un producto.

SDLC

Ciclo de Vida del Desarrollo de Software
Es una serie de pasos.

Objetivos

- Software de alta calidad.
- Cumplimiento de entregas.
- Estimación de costos.

1 Requirements Analysis

2 Design

3 Development

4 Testing

5 Maintenance

Software
Development
Life Cycle



```
graph TD; 1[1 Requirements Analysis] --> 2[2 Design]; 2 --> 3[3 Development]; 3 --> 4[4 Testing]; 4 --> 5[5 Maintenance];
```

1. Comunicación

- Usuario realiza la petición de un producto de software.

2. Recolección de Solicitudes

- Reuniones con el cliente.
- Obtener la máxima cantidad de información.
- Clasificación de requisitos
 - Requisitos de usuario
 - Requisitos funcionales
 - Requisitos del sistema

2. Recolección de Solicitudes

Herramientas

- Analizando el software o sistema actual o obsoleto.
- Entrevistas.
- Consultando documentación o base de datos.
- Encuestas o cuestionarios.

3. Estudio de viabilidad

- Análisis de requisitos.
- Nivel Financiero.
- Nivel Práctico.
- Nivel Tecnológico.

Herramientas.

Algoritmos, presupuestos, documentación existente.

4. Análisis del Sistema

- Crear el mejor y más conveniente modelo de software.
- Limitaciones del software.
- Problemas relacionados con en el desarrollo.
- Problemas relacionados con la implementación.
- Impacto del proyecto.

4. Análisis del Sistema

Arquitectura de la información

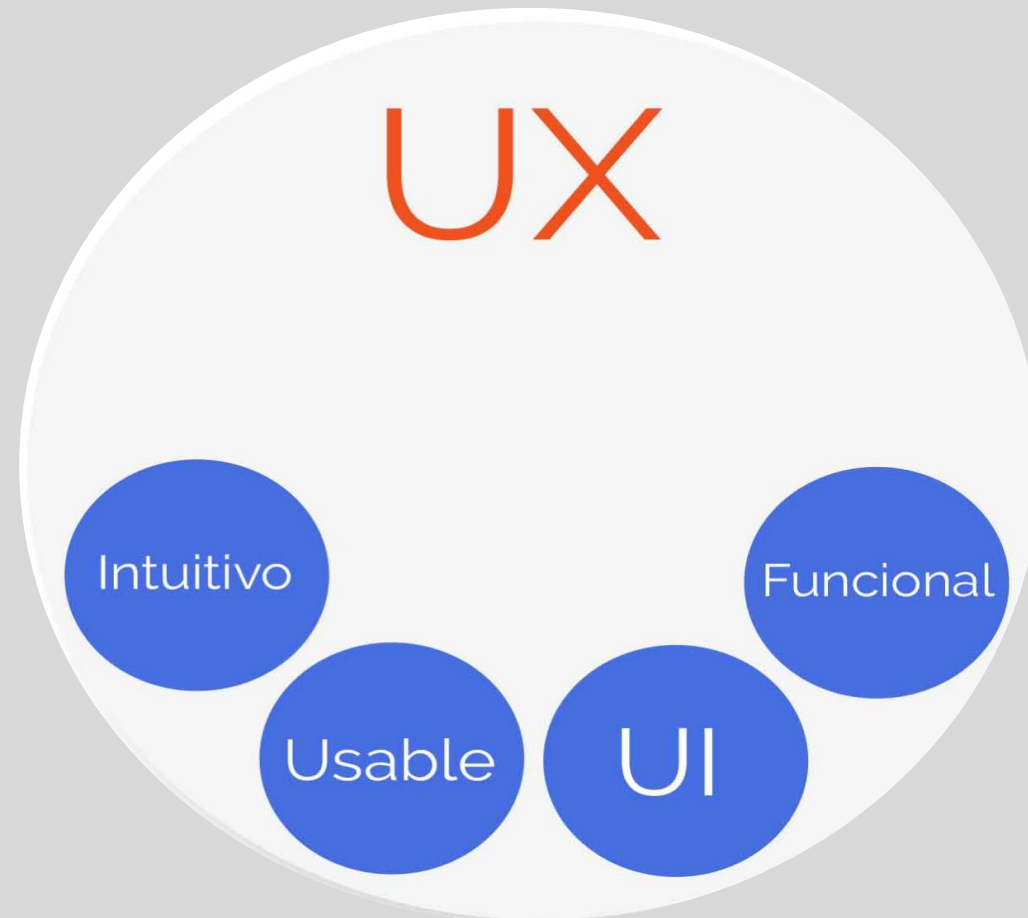
- Estructuras y esquemas organizativos: clasificación y estructura de la información.
- Sistema de etiquetado: representación de la información.
- Sistema de navegación: desplazamiento por la información.
- Sistema de búsqueda: buscar y encontrar la información.

4. Análisis del Sistema

UX (User Experience)

- Conocer al usuario final.
- Diseñar un producto que resuelva las necesidades del usuario.
- Mejorar la interacción.

4. Análisis del Sistema



4. Análisis del Sistema

Wireframe

- Guías visuales que representan la estructura de un sitio.
- Esquema del diseño de la página.
- Orden de contenido.
- Representar la funcionalidad, el comportamiento y la jerarquía de la información.

4. Análisis del Sistema

Tipos de Wireframe

- Wireframe Básicos.
Blanco y negro.
- Wireframe Anotados.
Notas breves.
- Wireframe de Flujo de Usuario.
Interactivo en visualizaciones estáticas.
- Wireframe Interactivos de Alta Definición.
Experimentar interacciones (toques, clics, deslizamientos)

4. Análisis del Sistema

Proceso de diseño de wireframe

1. Objetivo principal de la página web.
2. Dividirla en tres partes.
 - Encabezado
 - El cuerpo
 - Pie de página
3. Navegación (Enlaces, botones).
4. Anotaciones

DEMO

Balsamiq

A decorative horizontal bar at the bottom of the slide, consisting of a thin yellow line on top of a thicker blue line.

EJERCICIO

Balsamiq

Agenda Telefónica Web

Realizar el **wireframe** de un sistema que permita las siguientes funcionalidades:

- Registro del usuario.
- Autenticación del usuario.
- Almacenamiento de contactos.
- Agendar citas.

4. Análisis del Sistema

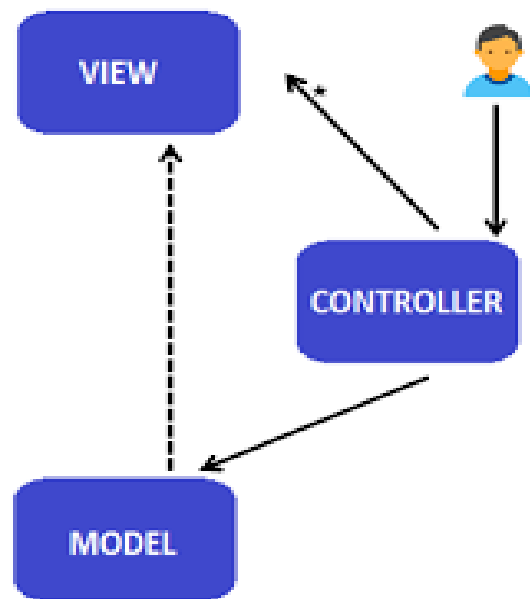
- Planificación
 - Tareas
 - Fecha de inicio
 - Fecha fin
 - Duración
- Recursos

5. Diseño

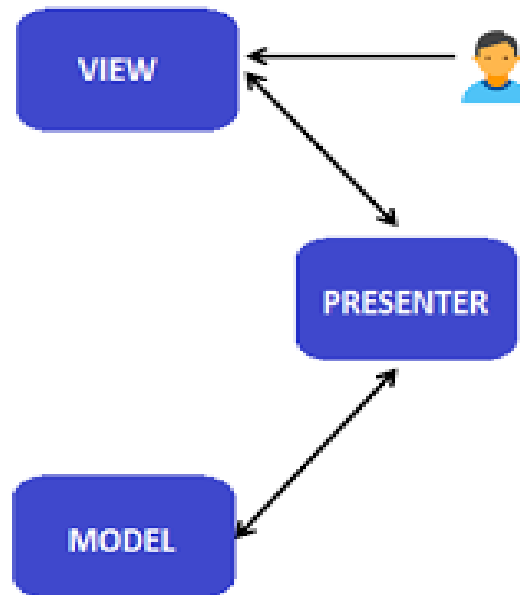
- Clasificación de Requisitos.
- Representa la calidad, rendimiento y el buen funcionamiento.
- Patrones de diseño
- UX/UI

5. Diseño

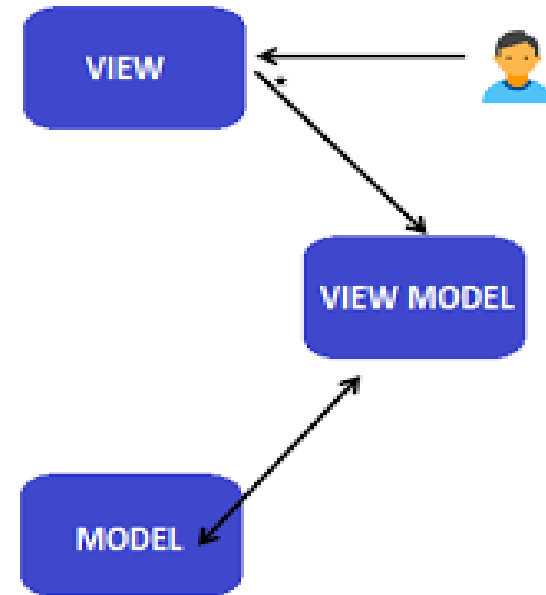
PATRONES DE DISEÑO



MVC

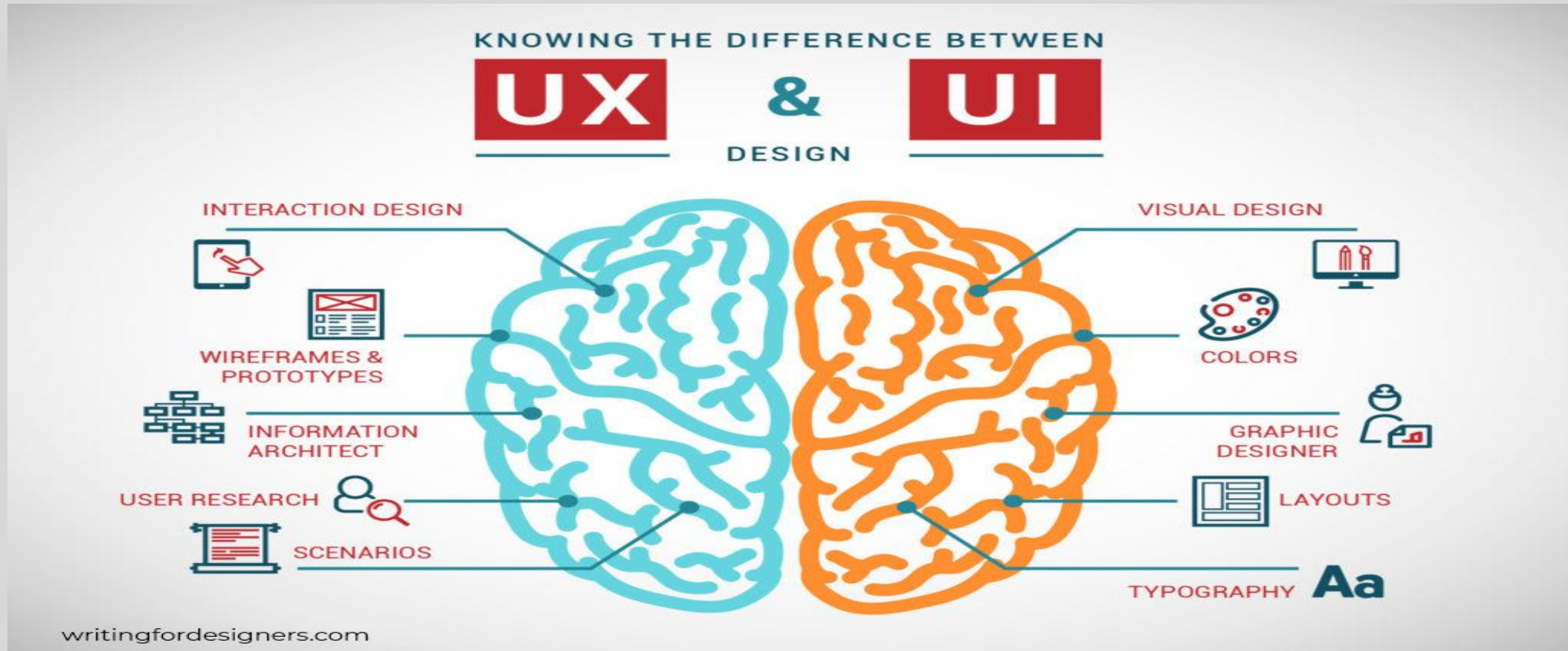


MVP



MVVM

5. Diseño



6. Codificación

- Denominada Fase de Programación
- Determinar el lenguaje de programación mas conveniente.
- Aplicar buenas prácticas de programación.
- Documentación interna, externa y de usuario.

7. Pruebas

- Comprobar que el software responda/realice correctamente las tareas indicadas.
- Realizadas por:
 - Desarrolladores
 - Personas Inexpertas
 - Equipo de QA

8. Integración

- Software externos (Base de datos, Chat, otros).

9. Implementación

- Poner a disposición del cliente el producto.
- Instalación del software.
- Adaptabilidad.
- Portabilidad.

10. Mantenimiento y Funcionamiento

- Eficiencia.
- Menor cantidad de errores.
- Capacitación.
- Documentación.

11. Disposición

- Falla en la ejecución.
- Software obsoleto.
- Actualización de software.

**Gracias por su
atención**
