



HERNÁNDEZ GARCÍA ALFREDO

DESARROLLO DE SISTEMAS WEB

UNIDAD-2

ACTIVIDAD-2.3: PROPUESTA DEL PROCESO DE FRONT-END

ASESORA: URSULA DEL PILAR GONZÁLEZ ROBLES

OBJETIVO DEL DISEÑO DEL FRONTEND

El objetivo del proceso de diseño del **Frontend** es construir una interfaz **intuitiva, accesible, moderna y de alto rendimiento** que permita a los usuarios — desarrolladores, empresas y reclutadores— **interactuar eficientemente** con las funcionalidades del sistema: gestión de perfiles, publicación de ofertas, búsqueda de empleo, mensajería y notificaciones en tiempo real.

El diseño debe ofrecer una **experiencia fluida**, centrada en la usabilidad y la adaptabilidad a distintos dispositivos (desktop, tablet, móvil), garantizando así la **satisfacción del usuario final** y la **eficiencia del proceso de reclutamiento**.

Justificación del proceso de diseño del Frontend

El proceso de diseño se plantea siguiendo principios de **Diseño Centrado en el Usuario (UCD)** y metodologías **ágiles** (Scrum), para asegurar que el producto final responda a las necesidades reales de los usuarios y evolucione de manera iterativa y validada.

A continuación, se detallan las fases justificadas del proceso:

1. Investigación y Análisis de Requisitos

Justificación:

Antes de definir la interfaz, es fundamental comprender el contexto del usuario y los objetivos del portal.

Actividades:

- Identificación de perfiles de usuario:
 - Desarrolladores (buscan empleo, muestran habilidades, gestionan su portafolio).
 - Empresas (publican vacantes, revisan candidatos).
 - Reclutadores (filtran, contactan y gestionan procesos de selección).
- Identificación de requerimientos funcionales y no funcionales.
- Análisis de experiencia de usuario esperada (fluidez, personalización, interacción social).

Resultado:

Mapa de requisitos de usuario y definición de **personas** (arquetipos de usuario).

2. Arquitectura de la Información y Definición de Flujos

Justificación:

Una estructura de información clara mejora la **navegación** y la **comprensión** del sistema.

Se definen los **flujos de usuario** para las principales funcionalidades, asegurando una experiencia coherente y lógica.

Actividades:

- Definición de jerarquías de contenido y rutas (Vue Router).

Resultado:

Estructura de navegación validada con el Product Owner.

3. Diseño de Interfaz y Prototipado (UI/UX)

Justificación:

El diseño visual es esencial para transmitir confianza, profesionalismo y facilidad de uso.

Mediante prototipos de baja y alta fidelidad se garantiza la **iteración temprana con usuarios reales** y el **alineamiento con la identidad visual del portal**.

Actividades:

- Diseño de interfaz de usuario (UI) con enfoque minimalista, limpio y responsive.
- Definición del sistema de diseño (colores, tipografía, botones, formularios, iconografía).
- Diseño de componentes reutilizables (botones, cards, modales, formularios).

Resultado:

Prototipo navegable validado con stakeholders y usuarios representativos.

4. Implementación Técnica del Frontend

Justificación:

Se selecciona **Vue.js** como framework principal por su facilidad de integración con APIs RESTful, su curva de aprendizaje accesible y su **reactividad** nativa, lo que mejora la experiencia del usuario.

Tecnologías Clave:

- **Vue.js 3 (Composition API):** para componentes modulares y mantenibles.
- **Axios / Fetch API:** para comunicación con el backend (microservicios).
- **Tailwind CSS / Vuetify:** para un diseño moderno, responsivo y coherente.
- **Socket.io-client:** para integrar mensajería y notificaciones en tiempo real.

Actividades:

- Implementación del sistema de componentes reutilizables.
- Consumo de las APIs RESTful del backend.
- Integración con OAuth (Google, GitHub) para autenticación.
- Manejo de errores y validaciones visuales.
- Implementación de pruebas unitarias con Jest o Vitest.

Resultado:

Frontend funcional, conectado con los microservicios backend, listo para pruebas de integración.

5. Pruebas de Usabilidad y Optimización

Justificación:

Las pruebas garantizan que la experiencia del usuario sea satisfactoria, detectando problemas de navegación, rendimiento o accesibilidad antes del despliegue.

Actividades:

- Pruebas de usabilidad con usuarios finales.
- Optimización de tiempos de carga (lazy loading, compresión de recursos).
- Adaptación responsiva para móviles y tablets.

Resultado:

Interfaz optimizada, accesible y validada para uso real.

6. Despliegue y Mantenimiento Continuo

Justificación:

El despliegue ágil y el mantenimiento continuo aseguran la actualización del producto frente a nuevas necesidades del mercado o retroalimentación de usuarios.

Actividades:

- Configuración del pipeline de CI/CD (GitHub).
- Despliegue en un CDN o servicio de hosting (AWS S3).
- Actualizaciones periódicas según feedback y evolución del backend.

Resultado:

Frontend productivo, mantenible y escalable en sincronía con el backend.

☐ Ventajas del enfoque propuesto

Aspecto	Beneficio
Modularidad (Vue.js + componentes reutilizables)	Facilita el mantenimiento y la escalabilidad.
UX centrado en el usuario	Aumenta la satisfacción y retención de usuarios.
Integración con APIs RESTful	Comunicación fluida y eficiente con los microservicios backend.
Diseño responsivo	Permite el uso en múltiples dispositivos y pantallas.
Arquitectura limpia y documentada	Facilita el trabajo en equipo y futuras ampliaciones.
Rendimiento optimizado	Mejora la velocidad y reduce la tasa de rebote.

□ Conclusión

El proceso de diseño del Frontend del portal de empleo se fundamenta en **principios de diseño centrado en el usuario** y en **prácticas ágiles de desarrollo iterativo**, garantizando una **interfaz moderna, accesible y funcional**.

El uso de **Vue.js** y un **sistema de diseño modular** asegura una experiencia de usuario coherente y una integración eficiente con el backend basado en microservicios, permitiendo la evolución futura del sistema sin comprometer su estabilidad ni rendimiento.