***12 Software***

***13 Componentes de Software***

***14 Configuración de Software***

***ISO/IEC 29110-4-1:2011***

**12.- Software**

* + **Elementos de Software (Código fuente y código ejecutable):**El código fuente se encuentra organizado dentro de la carpeta 'src/', donde residen todos los componentes de React, archivos de estilo (como 'App.css'), recursos estáticos (imágenes, tipografías, scripts) y estructuras propias del proyecto. Este código es mantenido en un sistema de control de versiones (Git), permitiendo su trazabilidad, colaboración y respaldo.  
      
    El código ejecutable se genera al correr el comando 'npm run build', lo que crea una versión optimizada y empaquetada del proyecto en la carpeta 'dist/'. Durante la auditoría, es esta versión la que se revisa, evaluando su funcionamiento en el entorno real de implementación (por ejemplo, en el servidor institucional del sitio Clavijero). Es fundamental que esta versión sea consistente con el código fuente versionado y que la ejecución sea estable y funcional en el entorno de destino.

Gestión de Artefactos en Proyectos Web (React + Vite + Tailwind)

1. Estructura del Proyecto y Artefactos

src/: Contiene el código fuente principal. Aquí están los componentes React, estilos, imágenes, scripts propios y recursos tipográficos.

- App.jsx, main.jsx, Unidad2.jsx, etc.: Componentes React que representan partes de la interfaz.

- App.css, index.css: Estilos base o globales, posiblemente gestionados por Tailwind CSS.

- assets/, css/, img/, tipografia/, visores/: Carpetas con recursos estáticos como imágenes, fuentes y estilos personalizados.

- programa\_de\_estudio.pdf: Documento PDF usado como referencia o recurso descargable.

2. Gestión de Artefactos

Módulos y Librerías

- node\_modules/: Carpeta generada automáticamente por npm con todas las dependencias externas (React, Tailwind, etc.).

- package.json: Lista módulos y versiones, scripts y metadatos del proyecto.

- package-lock.json: Bloquea versiones específicas para asegurar entornos consistentes.

3. Versionado

Git (Control de versiones)

- .gitignore: Archivos que no se deben incluir en el repositorio (como node\_modules o dist).

- Uso de GitHub, GitLab, etc., para respaldo, colaboración y control de cambios.

- Buenas prácticas: commits claros y frecuentes, uso de ramas, integración mediante pull requests.

4. Almacenamiento

- Local: Archivos del proyecto en la computadora del desarrollador.

- Nube: Repositorio Git (GitHub, GitLab) y almacenamiento de recursos estáticos (imágenes, fuentes) en public/ o assets/.

5. Scripts

En package.json:

- dev: vite → Ejecuta servidor local de desarrollo.

- build: vite build → Genera la carpeta dist/ con archivos optimizados.

- preview: vite preview → Visualiza el proyecto como si estuviera en producción.

6. Despliegue

- Se ejecuta npm run build para crear la carpeta dist.

- Se sube dist/ a servidores como Netlify, Vercel o institucionales (como el del sitio Clavijero).

- Opcional: configuración de dominios personalizados.

7. Enfoques por área

|  |  |  |
| --- | --- | --- |
| **Área** | **Carpeta / Archivo** | **Acción** |
| Código fuente | src/ | Organización modular |
| Dependencias | package.json | Mantener actualizado |
| Versionado | Git, .gitignore | Uso de ramas, commits |
| Recursos estáticos | img/, public/ | Optimización y nombres claros |
| Estilos | css/, tipografia/ | Bien separados y referenciados |
| Despliegue | dist/, vite.config.js | Verificación de rutas, build |
| Configuración | tailwind.config.js | Personalización del framework |

**13.- Componentes de Software**

* https://drive.google.com/drive/folders/1YkewlvEqLqkv6rKHYIdEiWxj96uVV\_uu

**14.- Configuración de Software:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Componente** | **Ruta en Git** | **Versión Inicial** | **Descripción breve** |
| CS01 | Código fuente Frontend (React+Vite) | /src/ | V1.0.0 | Todos los componentes .jsx, CSS, assets |
| CS02 | Archivos estáticos (public) | /public/ | V1.0.0 | Index.html, favicon, logos, PDFs embebidos |
| CS03 | Dependencias | /package.json + /package-lock.json | ------------ | Listado y versionesde librerías npm |
| CS04 | Build output | /dist/ | V1.0.0 | HTML, JS, CSS minificados para despliegue |
| CS05 | Configuración del proyecto | /vite.config.js, tailwind.config.js, .eslintrc.js | V1.0.0 | Ajustes de bunding, estilos y linting |
| CS06 | Assets de proyecto | /src/assents/, /src/img/, /src/tipografia/, /src/visores/, /src/css/ | V1.0.0 | Imágenes, fuentes, estilos globales |
| CS07 | Páginas y componentes clave | /src/main.jsx, /src/App.jsx, /src/NuevaPagina.jsx, /src/TableProgram.jsx, /src/DinamicaTrabajo.jsx, /src/Unidad2.jsx, /src/Unidad3.jsx | V1.0.0 | Páginas y vistas principales |
| CS08 | Documentación interna | /README.nd, /src/programa\_de\_estucios (1).pdf | V1.0.0 | Guía rápida, plan de estudios |

Juan Pérez María Gómez

.

Jefe de Departamento Diseñador Informático 1

Luis Rodríguez Ana Castillo

.

Diseñador 2 Diseñadora Grafica

Sofia Ramírez

.

Diseñadora Institucional