**Rúbrica de Evaluación – Proyecto “Aplicación React + Firebase”**

**1. Creación del Proyecto React**

* **Creación correcta del proyecto con Vite o Create React App.**
* **Captura de pantalla de la interfaz de Visual Studio Code mostrando la estructura inicial del proyecto.**

**2. Estructura de Archivos y Carpetas**

* **Estructura de carpetas organizada, p. ej.:**
  + **src/pages**
  + **src/components**
  + **src/assets**
  + **src/firebase.js**
* **Archivos bien nombrados y ubicados.**

**3. Configuración de Firebase**

* **Creación del proyecto en Firebase Console.**
* **Registro de la App Web en Firebase.**
* **Captura de:**
  + **Firebase Console con el proyecto creado.**
  + **Pantalla con la configuración web (SDK) de Firebase.**
* **Instalación de Firebase en el proyecto (npm install firebase).**
* **Creación y correcta configuración del archivo firebase.js.**

**4. Implementación de Autenticación con Google**

* **Configuración de método de inicio de sesión en Firebase (Google).**
* **Captura de pantalla de Firebase Console mostrando Auth habilitado.**
* **Botón visible en la app para iniciar sesión con Google.**
* **Manejo de errores en el login con alertas (p. ej. SweetAlert2).**
* **Redirección al dashboard tras login exitoso.**
* **Captura del usuario logueado en Firebase Auth console.**

**5. Configuración de Rutas con React Router**

* **Instalación y configuración de React Router DOM.**
* **Creación de rutas públicas:**
  + **Login**
  + **Register**
  + **Forgot Password**
* **Creación de rutas privadas:**
  + **Dashboard**
  + **Otras páginas protegidas**
* **Implementación de <ProtectedRoute /> con Firebase Auth.**
* **Captura del archivo App.js o App.jsx con las rutas configuradas.**

**6. Validación de Formularios**

* **Validación de campos de login (email, password) con mensajes de error claros.**
* **Uso de librerías como SweetAlert2 para mostrar mensajes.**
* **Código limpio y legible.**

**7. Implementación de Hooks**

* **Uso de hooks básicos de React:**
  + **useState**
  + **useEffect**
* **Implementación de un hook extra opcional (ej.: useRef, useContext, useAuthState de react-firebase-hooks).**
* **Capturas o fragmentos de código de cada hook usado.**

**8. Diseño Responsivo y Visual**

* **Uso de Bootstrap (CDN o instalación) o librerías UI modernas (Material UI, Tailwind, etc.).**
* **Diseño responsivo, interfaz limpia.**
* **Capturas en diferentes tamaños de pantalla (desktop, mobile).**

**9. Spinner o Loading**

* **Implementación de un spinner visual durante:**
  + **carga de auth**
  + **redirecciones**
* **Integración opcional con SweetAlert para loaders.**

**10. Página 404 (Not Found)**

* **Creación de página de error para rutas inexistentes.**
* **Diseño acorde a la identidad visual de la app.**
* **Captura de la página 404.**

**11. Uso de Git y GitHub**

* **Proyecto subido a GitHub.**
* **Commits claros, con mensajes descriptivos.**
* **README con:**
  + **Capturas de interfaz.**
  + **Instrucciones de instalación.**
  + **Tecnologías utilizadas.**
* **Captura del archivo README.md en GitHub.**

**12. Despliegue en Firebase Hosting**

* **Build de la app (e.g. npm run build).**
* **Configuración de Firebase Hosting.**
* **Ejecución de:**
  + **firebase login**
  + **firebase init**
  + **firebase deploy**
* **Captura del despliegue exitoso.**
* **Link funcional del proyecto en Firebase Hosting.**

**13. Entrega Completa**

* **Entrega de:**
  + **Link del repositorio GitHub.**
  + **Link del despliegue en Firebase Hosting.**
  + **Capturas o PDF con evidencias de cada punto.**