



Informe final: Desarrollo front-end de una aplicación empresarial para la gestión certificados.

Maria Isabel Rua Velez

Ingeniera de sistemas

Asesor

Luis Hernando Silva Florez

Universidad de Antioquia

Facultad de ingeniería

Ingeniería de sistemas

Medellín

2024

Cita

(Rúa Velez, 2024)

Referencia

Rúa Velez, M. I.(2024). *Desarrollo front-end de una aplicación empresarial para la gestión certificados* [Presencial]. Universidad de Antioquia, Medellín.

Estilo APA 7 (2020)



Centro de Documentación Ingeniería (CENDOI)

Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - www.udea.edu.co

Rector: Jhon Jairo Arboleda Céspedes

Decano/Director: Julio César Saldarriaga Molina

Jefe departamento: Diego José Luis Botía

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Dedicado a mis familiares y seres queridos que incondicionalmente me apoyaron en mis
decisiones dentro de mi formación profesional

Agradecimientos a

Blankfactor C.O

Johana Saldarriaga

LUIS HERNANDO SILVA FLOREZ

Tabla de contenido

Lista de figuras	5
Siglas, acrónimos y abreviaturas	6
Resumen	7
Abstract	8
Introducción	9
Objetivos	10
1.1 Objetivo general	10
1.2 Objetivos específicos	10
Marco teórico	11
Metodología	12
3.1 Análisis teórico y levantamiento de requerimientos	13
3.2 Desarrollo ágil con Scrum	14
Resultados	17
4.1 Proceso Investigativo para la Selección de Arquitectura	17
4.2 Creación del diseño del aplicativo web	18
4.3 Procesos de desarrollo	22

Lista de figuras

Figura 1. Figma Design system	12
Figura 2. Arquitectura del aplicativo	13
Figura 3. Atomic design	14
Figura 4. Visual story mapping para MVP	15
Figura 5. Ciclo de SCRUM	17
Figura 6. Flujo de autenticación para nivel empleado	21
Figura 7. Flujo de autenticación nivel administrador	22
Figura 8. Guia de estilos basado en la marca	23
Figura 9. Vistas de la aplicación desde figma	24
Figura 10. Vista de inicio de la aplicación	24
Figura 11. Vista inicial de la generación de certificados	26
Figura 12. Vista de generación de certificados con todos los campos desplegados	27
Figura 13. Vista de generación de certificados con error en uno de los campos	28
Figura 14. Vista selector de empleado para generación de certificado para administradores	29
Figura 15. Vista generación de certificados para administradores	30
Figura 16. Vista certificado generado en PDF	31
Figura 17. Vista de reportes	32
Figura 18. Cobertura de código de la aplicación	33

Siglas, acrónimos y abreviaturas

UdeA	Universidad de Antioquia
UI	User Interface (Interfaz de usuario)
UX	User Experience
HTML	HyperText Markup Language
RTK	Redux tool kit
CSS	Cascade sheet styles
DOM	Document Object Model
SPA	Single Page Application
MVP	Producto Mínimo Viable
HU	Historia de Usuario
AWS	Amazon Web Services

Resumen

En este proyecto, se gestionó la solución a un desafío en el departamento de Recursos Humanos de Blankfactor relacionado con la generación de certificados laborales. La solución consistió en una aplicación web, en la cual el frontend fue desarrollado en React con autenticación y roles, que automatiza el proceso de generación de certificados, también, integración de la autenticación de Google, así como la creación de un formulario flexible y eficiente para la generación de certificados. Para la gestión del desarrollo se adoptó una metodología ágil, utilizando el marco de trabajo SCRUM.

Abstract

In this project, the solution to a challenge in the Human Resources department of Blankfactor related to the generation of employment certificates was managed. The solution involved a web application, where the frontend was developed in React with authentication and roles, automating the process of certificate generation. It also included the integration of Google authentication, as well as the creation of a flexible and efficient form for certificate generation. Agile methodology was adopted for development, utilizing the SCRUM framework.

Introducción

En el dinámico ámbito del desarrollo de software, adoptar procesos ágiles y adaptativos se ha convertido en un requisito esencial. Este enfoque representa un factor crítico para las empresas que buscan construir productos de alta calidad.

En el departamento de Recursos Humanos de Blankfactor se identificó un problema crítico relacionado con el proceso manual de generación de certificados laborales, liderado por el encargado del área. Este procedimiento involucra la consulta a Workday para obtener información de los empleados registrados, así como la gestión de un documento de Excel almacenado en Google Drive con detalles de aquellos empleados que dejaron la compañía antes de la implementación de la plataforma Workday. La actualización de este documento se realizaba manualmente cada mes, generando retrasos significativos en el proceso.

Como respuesta a esta problemática se propuso el desarrollo de una aplicación web que automatiza el proceso de generación de certificados laborales. La visión consistió en proporcionar al departamento de Recursos Humanos una sistema amigable con un módulo de reportes para el líder que permite ver la información de los certificados emitidos, al igual que otorgar a los empleados la capacidad de gestionar de forma independiente sus procesos de certificación.

En este informe, explicaremos detalladamente el desarrollo de los componentes, desde la concepción de la idea hasta la entrega final del producto, destacando la arquitectura usada y los resultados obtenidos y reflexionando sobre posibles mejoras y futuras direcciones de trabajo.

Objetivos

1.1 Objetivo general

Desarrollar componentes de una aplicación frontend en React con autenticación y roles, para la generación de certificados laborales.

1.2 Objetivos específicos

- Diseñar un proceso automatizado de generación de certificados que utilice la información almacenada y las plantillas definidas para generar documentos.
- Diseñar un sistema de plantillas flexibles que permita al administrador personalizar y ajustar los formatos de los certificados de acuerdo a sus necesidades.
- Diseñar un módulo de reportes que permita ver la información de los certificados emitidos.

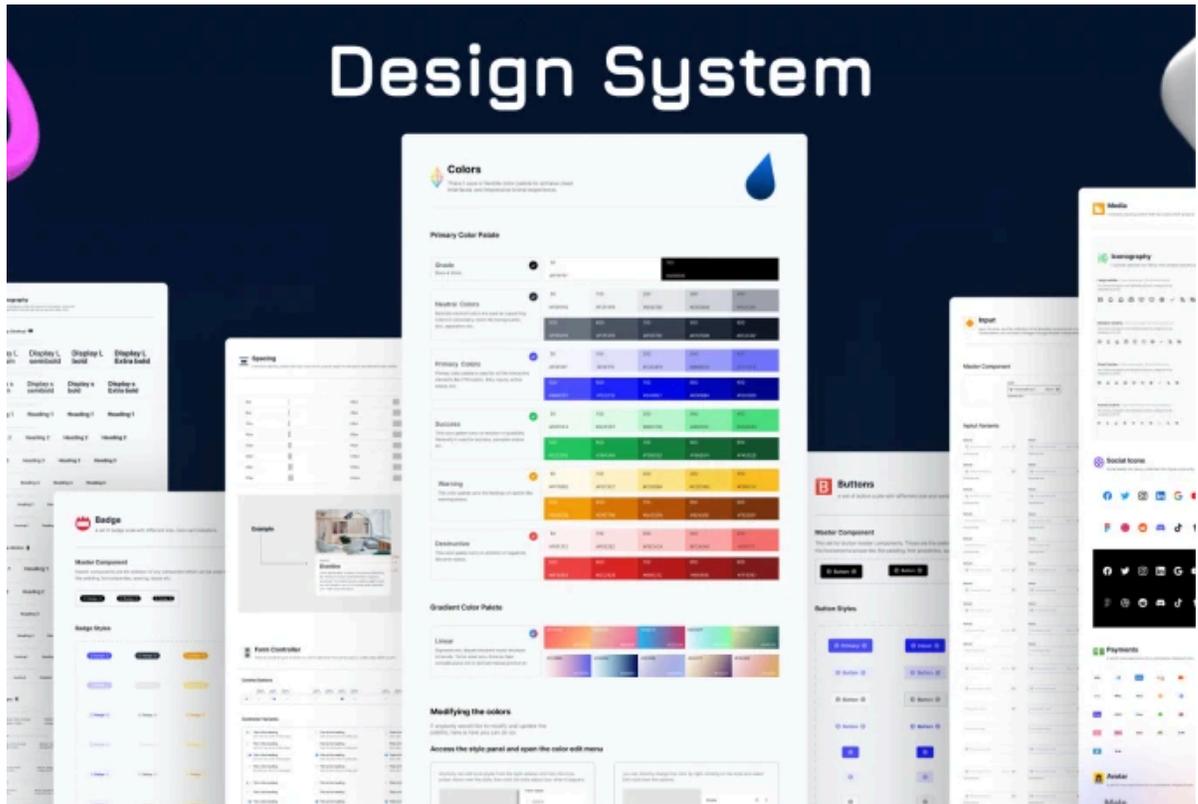
Marco teórico

En el departamento de Recursos Humanos de Blankfactor, se identificó un problema específico relacionado con la generación de certificados laborales, el cual se abordó mediante el desarrollo de una aplicación web.

La automatización de procesos en Recursos Humanos se ha vuelto fundamental para la optimización de tareas y la mejora de la productividad. En nuestro contexto, la generación de certificados laborales se convirtió en un proceso crítico que se pudo mejorar mediante tecnológicas para agilizar y automatizar su ejecución. Un certificado laboral es una carta que certifica el empleo de una persona. Generalmente, se crea con el membrete oficial de la organización para confirmar que el solicitante es un empleado actual o anterior. Este documento se suele requerir para verificar el estatus laboral, el cargo y el salario, e incluso puede contener el historial de empleo, direcciones de trabajos anteriores, crecimiento salarial y resumen del desempeño laboral.[1]

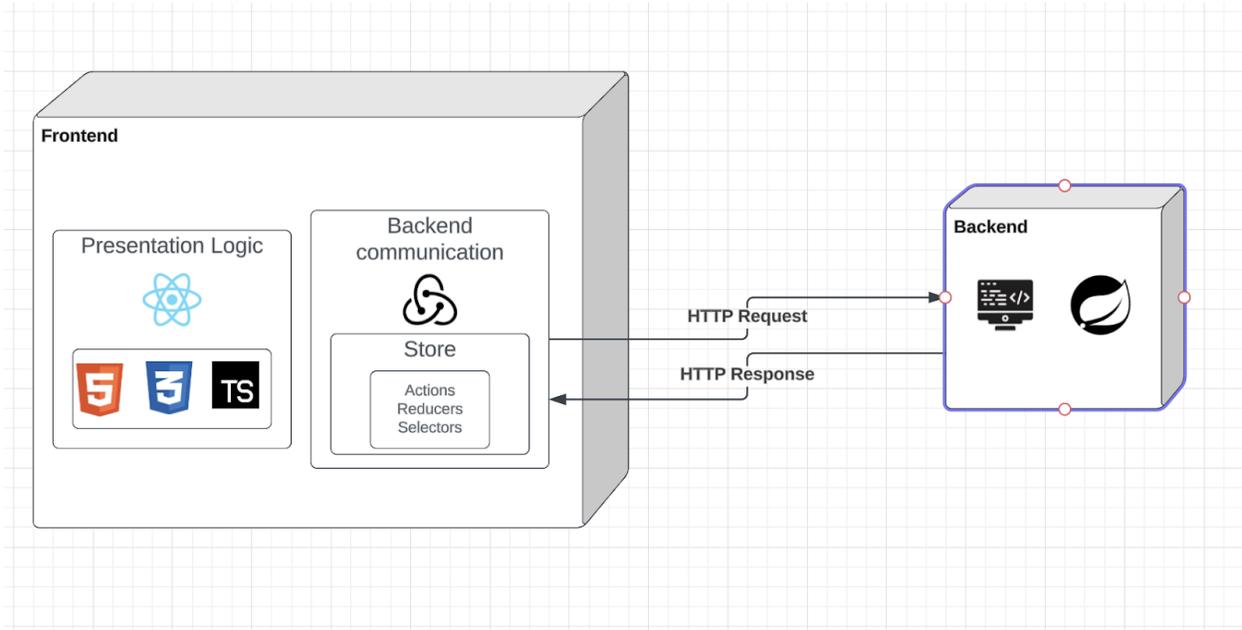
Para abordar el problema comenzamos por diseñar la interfaz de usuario de la aplicación, utilizando la herramienta Figma para este propósito. Figma Design (Ver Figura 1) es una herramienta popular que facilita la creación, compartición y prueba de diseños para sitios web, aplicaciones móviles y otras experiencias digitales. [2]

Figura 1. Figma Design system



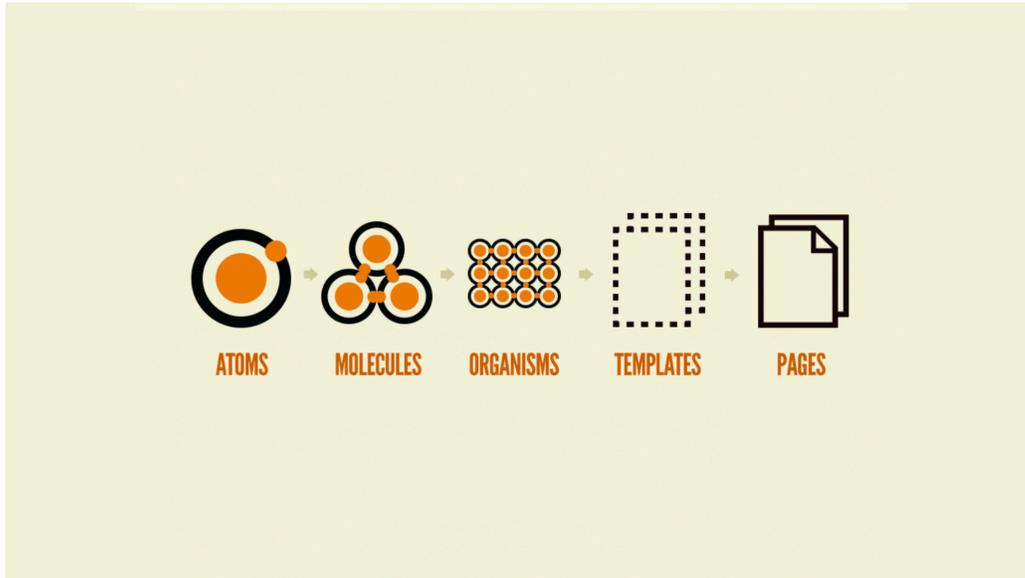
Nota: Fuente: [Figma Design](#)

El desarrollo web desempeña un papel clave en la automatización de procesos, permitiendo el acceso desde cualquier ubicación con conexión a Internet. Tecnologías como Typescript y Frameworks modernos (React, Angular, Vue) facilitan el desarrollo rápido y ágil de aplicaciones, adaptándose rápidamente a cambios en los requisitos del negocio [3] . Ver figura 2.

Figura 2. Arquitectura del aplicativo

La implementación de TypeScript mejora la legibilidad y seguridad del código, mientras que el profundo conocimiento de React es esencial para el desarrollo del FrontEnd. React JS es una biblioteca de código abierto utilizada para crear interfaces de usuario para aplicaciones de una sola página, permitiendo cambios dinámicos sin recargar la página.[4]

Durante el desarrollo, se optó por la metodología Atomic Design (Ver figura 3), que organiza los componentes en cinco elementos: Átomos, Moléculas, Organismos, Plantillas y Páginas. Esto permite crear componentes reutilizables y modulares, pensando en la consistencia y escalabilidad del diseño. Al desglosar los elementos de la interfaz en unidades más pequeñas y manejables, pudimos garantizar una mayor cohesión visual y funcional a través de toda la aplicación. Esta metodología facilita también la colaboración entre diseñadores y desarrolladores, al proporcionar un lenguaje común y una estructura clara para la construcción de la interfaz de usuario. Además, al centrarnos en componentes reutilizables, optimizamos los recursos de desarrollo, permitiendo actualizaciones y modificaciones más ágiles y eficientes en el futuro. [5]

Figura 3. Atomic design

La gestión de la información en la aplicación se llevó a cabo mediante Redux, evitando malas prácticas en el manejo de datos. Redux es un contenedor predecible del estado de aplicaciones JavaScript, ofreciendo consistencia, portabilidad y facilidad de prueba. [6]

En aplicaciones empresariales, la seguridad y privacidad de los datos son críticas. La automatización en la generación de certificados debe cumplir con estándares de seguridad, incluyendo encriptación, acceso controlado y cumplimiento de regulaciones.

Para un despliegue rápido y eficiente se utilizó la plataforma Vercel, que proporciona herramientas, flujos de trabajo y una infraestructura sin necesidad de configuración adicional. Vercel es compatible con los marcos de front-end más populares y ofrece una infraestructura escalable y segura distribuida globalmente para una óptima velocidad de servicio.

conocimiento sólido sobre los principios y las mejores prácticas asociadas al framework de trabajo.

3.2 Desarrollo ágil con Scrum

En el contexto del marco de trabajo SCRUM (Ver figura 5), los sprints representan períodos específicos durante los cuales se desarrollan funciones clave del producto. Se establecieron todas las ceremonias asociadas con este marco de trabajo, las cuales incluyen:

1. Planning: Esta reunión tiene como objetivo alinear y consensuar las tareas que se llevarán a cabo durante el sprint.

2. Daily: Cada sprint se compone de ciclos diarios que comienzan con las reuniones diarias, conocidas como "dailys", donde los miembros del equipo comparten progresos y se identifican posibles obstáculos.

3. Demo: Esta es la instancia final de cada sprint, en la cual el equipo asignado al proyecto expone lo avanzado durante el periodo.

4. Retrospectiva: Es un análisis de los problemas que pudieron haber surgido en la ejecución del proyecto, la revisión de las velocidades del equipo, así como la revisión y aclaración de objetivos internos.

Siguiendo un calendario estructurado, se implementó un seguimiento detallado de los avances a través de reuniones diarias "dailys", donde los miembros compartían progresos y se identificaban posibles obstáculos. Los sprints, con una duración de dos semanas, incluyeron una demo los martes de la segunda semana para evaluar los resultados obtenidos.

La planning de actividades se llevó a cabo los miércoles de la primera semana del sprint, definiendo con detalle las tareas a realizar. Posteriormente, durante la segunda semana, se realizó el refinamiento haciendo ajustes y optimizando según las necesidades emergentes.

Figura 5. Ciclo de SCRUM



Nota: Fuente: [Introducción a SCRUM - Rubén Bernardo Guzmán Mercado](#)

Con el objetivo de fomentar una mejora continua, se programaron reuniones retrospectivas los jueves de la segunda semana, proporcionando un espacio para reflexionar sobre el sprint recién concluido y proponer ajustes que optimizaran futuras iteraciones. Esta estructura metodológica basada en Scrum facilitó una gestión ágil y adaptativa del proyecto, promoviendo la eficiencia y calidad en el desarrollo de las prácticas propuestas.

Para potenciar la colaboración y la gestión del proyecto, se incorporaron herramientas como Slack, Jira y Confluence. Slack se empleó de manera informal para mantener al equipo conectado y facilitar la colaboración. Jira se empleó para detallar con precisión el flujo de trabajo

y hacer un seguimiento minucioso del tiempo dedicado a cada tarea. Confluence se empleó para la documentación y almacenamiento de información, proporcionando un espacio centralizado para que todo el equipo tuviese acceso a los recursos del proyecto.

Resultados

Este proyecto logró desarrollar un MVP (Producto Mínimo Viable) integral, compuesto por varios módulos. El primero que analizaremos es el proceso investigativo llevado a cabo para la creación de una arquitectura de frontend, incluyendo la selección de herramientas.

4.1 Proceso Investigativo para la Selección de Arquitectura

Tras definir el MVP para nuestro proyecto de aplicativo web, el equipo de trabajo se enfocó en la elección de herramientas para el frontend, realizó una investigación exhaustiva para determinar las más adecuadas. Esta fase culminó con la decisión de desarrollar una aplicación web de tipo Single-Page Application (SPA), elegida por su capacidad de ofrecer una experiencia de usuario fluida y eficiente, recargando únicamente los componentes necesarios.

En el proceso de selección de la librería principal para desarrollar la SPA, se optó por React, utilizando el framework de Vite.js. Esta elección se basó en su popularidad y eficiencia para el desarrollo de SPA. Complementariamente, se eligió TypeScript como lenguaje de programación por su capacidad de añadir tipado estático, lo cual es esencial para mejorar la calidad y mantenibilidad del código.

Para asegurar un código limpio y seguir las mejores prácticas, se usaron las reglas de ESLint, junto con el formateador de código Prettier para mantener un estilo consistente en todo el código fuente. Adicionalmente, se adaptó la metodología de Atomic Design, que permitió construir un sistema de diseño modular para las interfaces, optimizando la cantidad de componentes y mejorando la eficiencia del código.

En cuanto a la estilización, se seleccionó Material UI por su capacidad de crear componentes altamente personalizables y alineados con el manual de marca de la compañía. Para las pruebas de componentes, se utilizó Storybook, que facilita las pruebas aisladas de cada elemento, y JEST, una biblioteca de JavaScript para pruebas unitarias.

En la fase de estructuración modular del proyecto, se incorporó la librería react-oauth para gestionar de manera segura y eficiente la autenticación con Google. Para el módulo de formularios, se optó por React Hook Form, evitando renderizados adicionales innecesarios y asegurando un alto rendimiento.

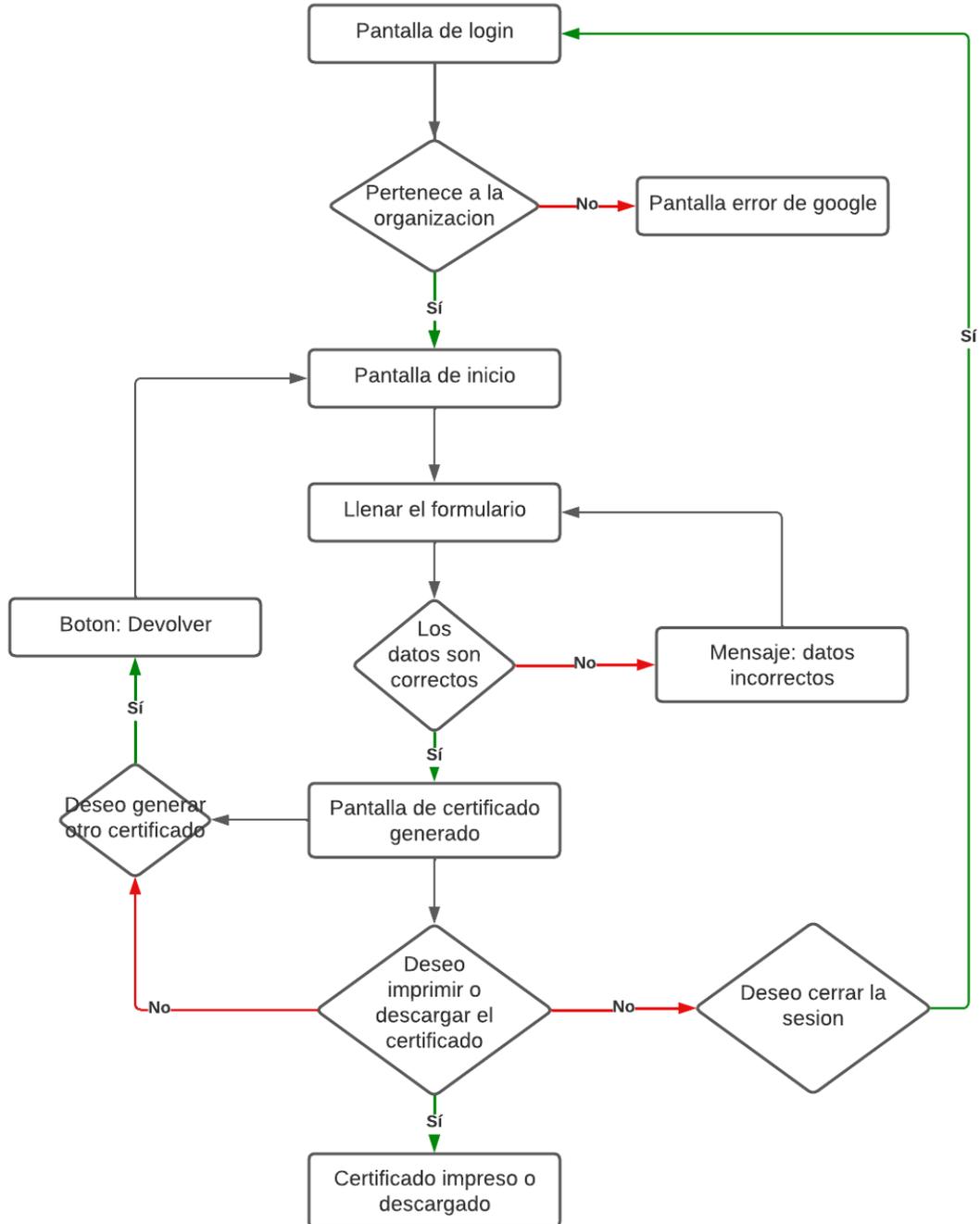
Finalmente, para el manejo de rutas en el aplicativo, se seleccionó la librería React Router, que se integra perfectamente con los componentes de React. Para el manejo de estados y las peticiones al backend, se decidió utilizar Redux, dada su eficacia y compatibilidad con nuestra arquitectura.

4.2 Creación del diseño del aplicativo web

Después de la definición de la arquitectura se realizó el proceso de diseño de las interfaces, que permitió el entendimiento del flujo del sistema. Con esto se crearon dos flujos diferentes según el tipo de usuario que ingrese a la aplicación, empleado (Ver figura 6) o administrador (Ver figura 7).

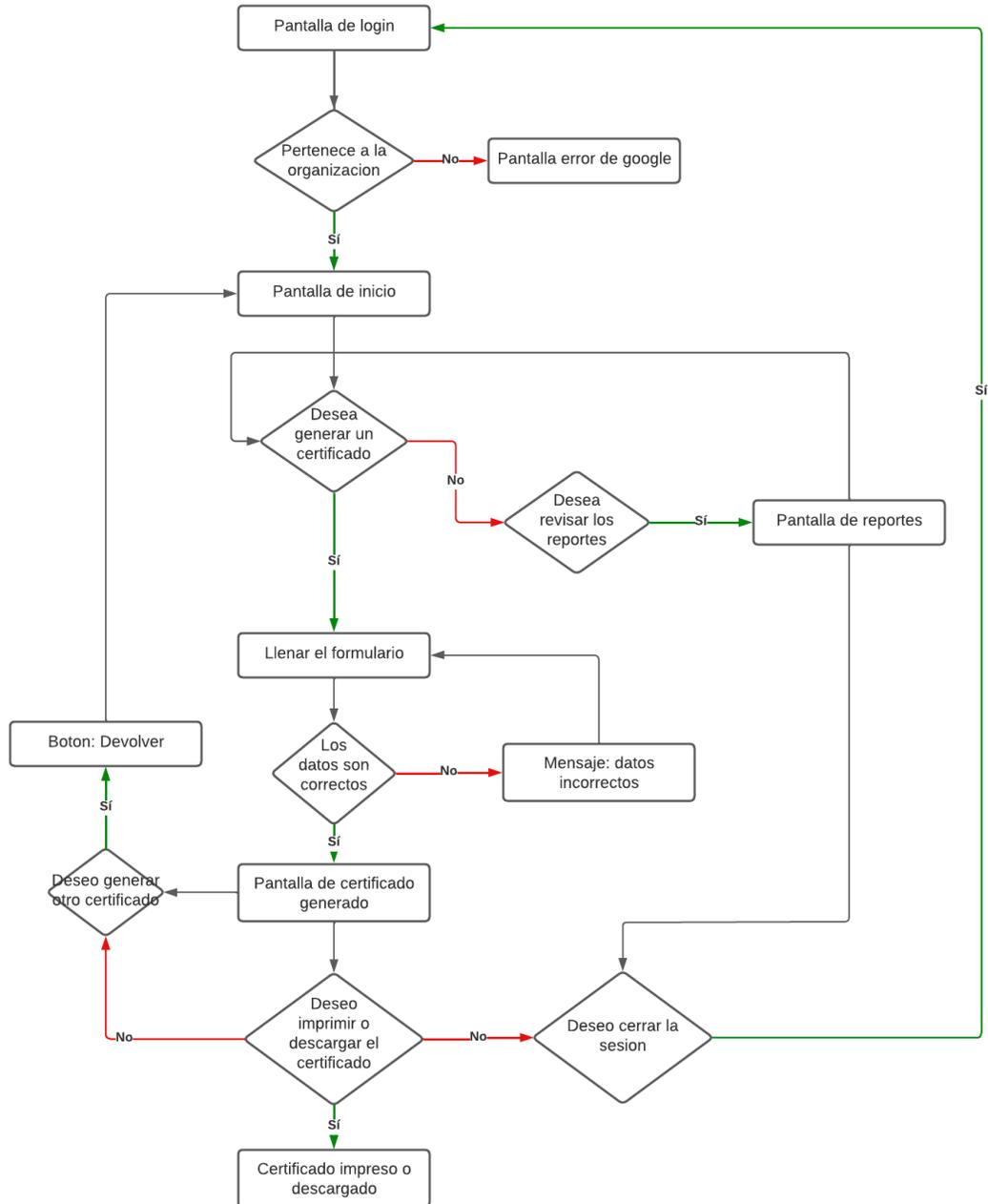
Para el empleado:

Figura 6. Flujo de autenticación para nivel empleado



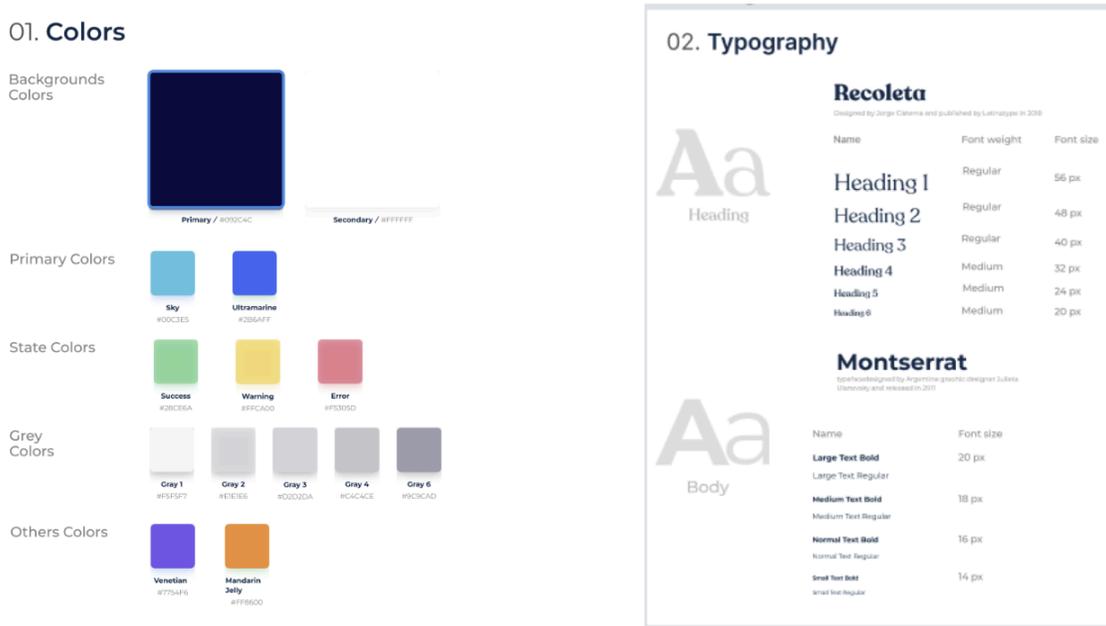
Para el administrador:

Figura 7. Flujo de autenticación nivel administrador

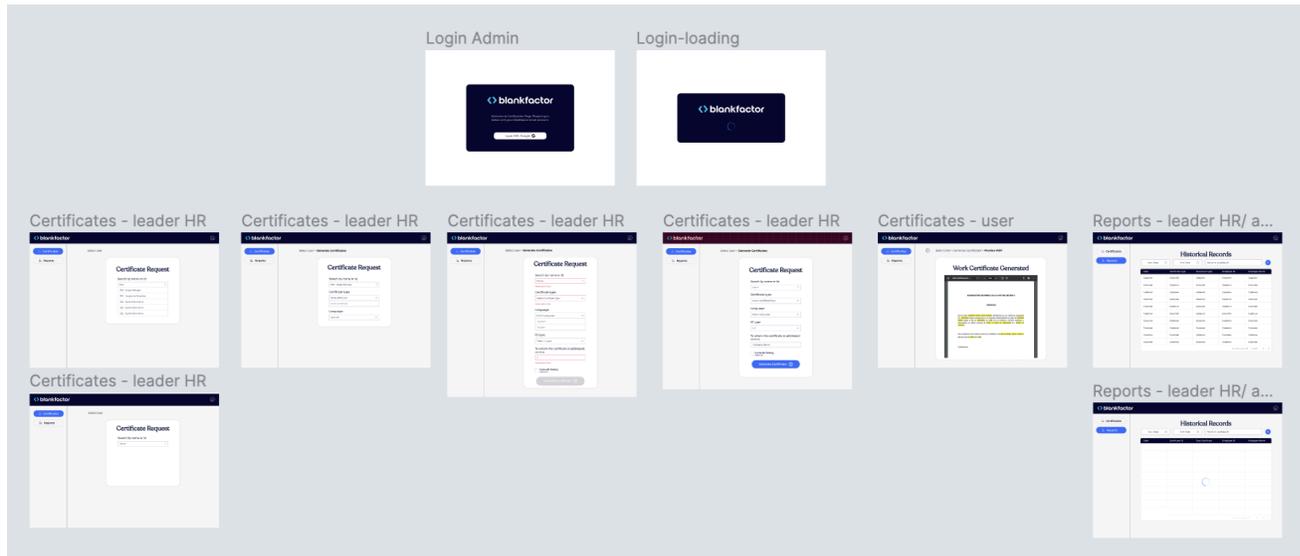


Con estos flujos creados el equipo de trabajo se puso en la tarea de crear una guía de estilos de la aplicación basados en el manual de marca de la compañía, en donde se definieron los colores primarios, la tipografía y los tamaños a usar.

Figura 8. Guia de estilos basado en la marca



Con esto definido se comenzó el desarrollo del diseño en el aplicativo Figma, con ayuda del flujo se crearon pantallas con diseños intuitivos y acordes a la compañía que permitieron tener una base clara para el diseño de los componentes que fueron construidos (Ver figura 8) .

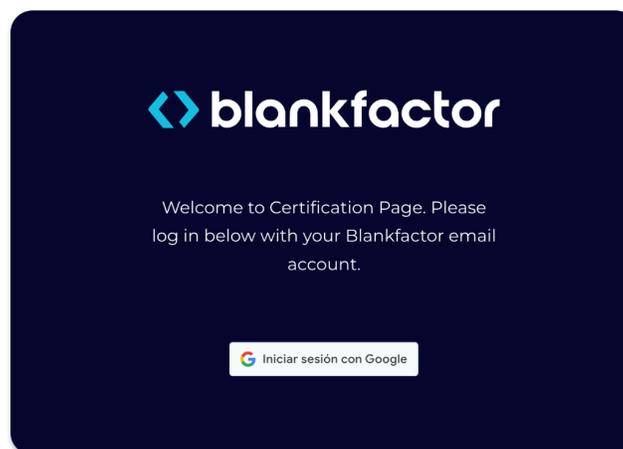
Figura 9. Vistas de la aplicación desde figma

Ya con las pantallas definidas y aprobadas por el product owner, se comenzó el proceso de desarrollo.

4.3 Procesos de desarrollo

Con los diseños listos y los flujos claros se empezaron a definir las tareas asignadas a cada integrante del equipo de trabajo en el backlog.

4.3.1 Módulo de login

Figura 10. Vista de inicio de la aplicación

Se completó la implementación del módulo de inicio de sesión, integrando la autenticación de Google. Este módulo, cuyo diseño se alineó con las especificaciones aprobadas en la plataforma de Figma por el product owner, representa un avance significativo en el proyecto (Ver figura 10).

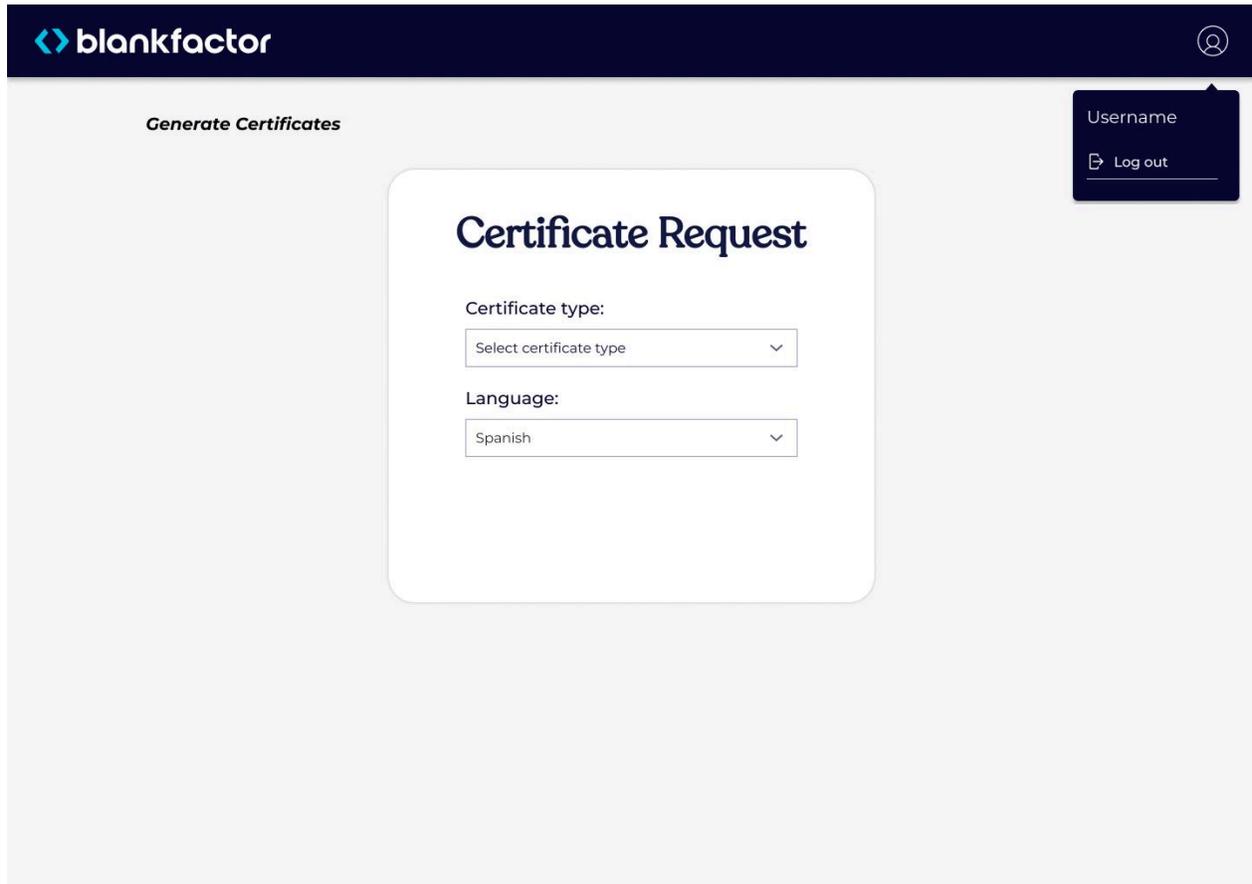
4.3.2 Módulo de certificados

La fase de desarrollo de este módulo comenzó con la creación del componente de formulario, utilizando React Hook Form. Esta herramienta resultó ser esencial, ya que permitió desarrollar un componente altamente eficiente, integrando todas las validaciones necesarias. El comportamiento del formulario se adapta en función del tipo de usuario que accede a la aplicación. (Ver figura 11)

En el caso de los empleados, el sistema interactúa con el backend para obtener los diferentes tipos de certificados disponibles. Al seleccionar un tipo de certificado, el formulario solicita al backend la especificación de los campos necesarios para ese certificado en particular, distinguiendo entre los campos opcionales y los obligatorios. Esta funcionalidad se implementó con el propósito de ofrecer una gama de plantillas de certificados que se adaptan a las diversas necesidades de cada certificado.

El flujo del formulario para el usuario se diseñó para ser claro e intuitivo, asegurando una experiencia de usuario fluida y eficiente.

Figura 11. Vista inicial de la generación de certificados



blankfactor

Generate Certificates

Certificate Request

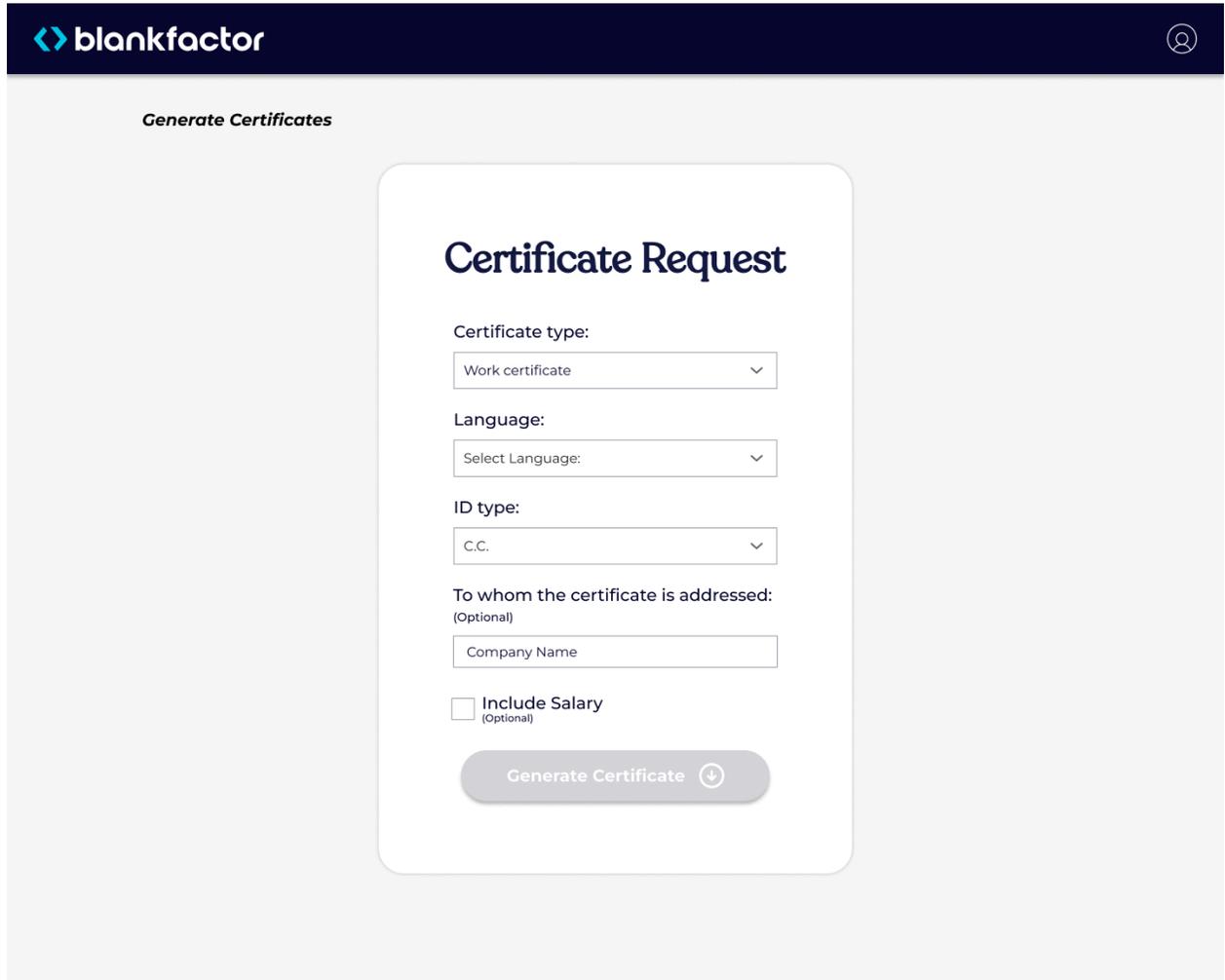
Certificate type:
Select certificate type

Language:
Spanish

Username
Log out

El primer paso para el empleado en este proceso es seleccionar el tipo de certificado deseado y el idioma en el que se emitirá. Una vez hecho esto, en función del tipo de certificado elegido, el sistema renderiza automáticamente los campos correspondientes (Ver figura 12).

Figura 12. Vista de generación de certificados con todos los campos desplegados



blankfactor

Generate Certificates

Certificate Request

Certificate type:
Work certificate

Language:
Select Language:

ID type:
C.C.

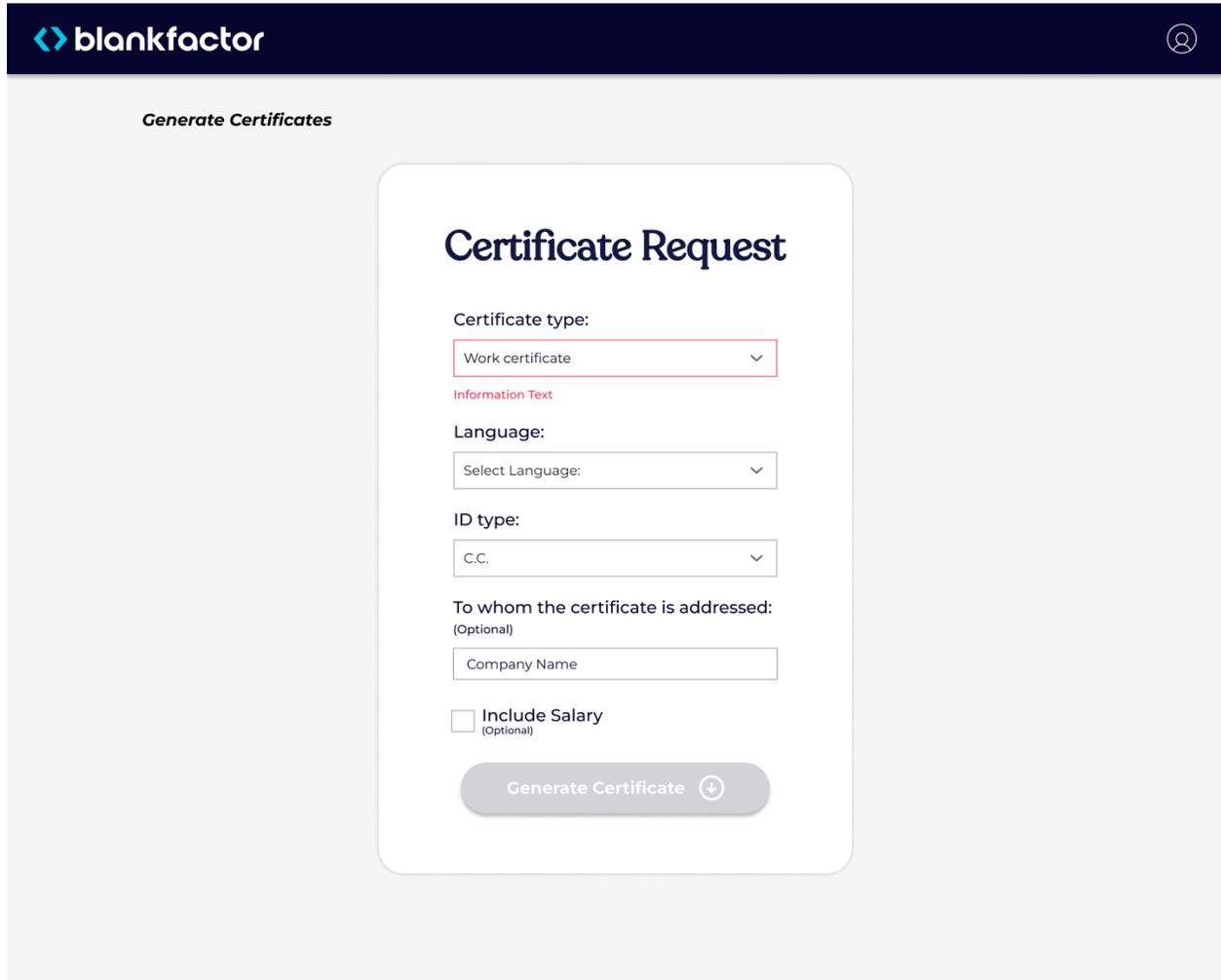
To whom the certificate is addressed:
(Optional)
Company Name

Include Salary
(Optional)

Generate Certificate

Los campos del formulario se mostrarán claramente, diferenciando entre los que son opcionales y los obligatorios. Además, se ha implementado una funcionalidad de feedback visual mediante el uso de colores en los botones: un botón gris indica que hay errores en el formulario o que algunos campos requeridos aún no se han completado. Si un campo específico está mal diligenciado se resaltará en rojo, acompañado de un mensaje de validación justo debajo, para guiar al usuario en la corrección del error (Ver figura 13).

Figura 13. Vista de generación de certificados con error en uno de los campos

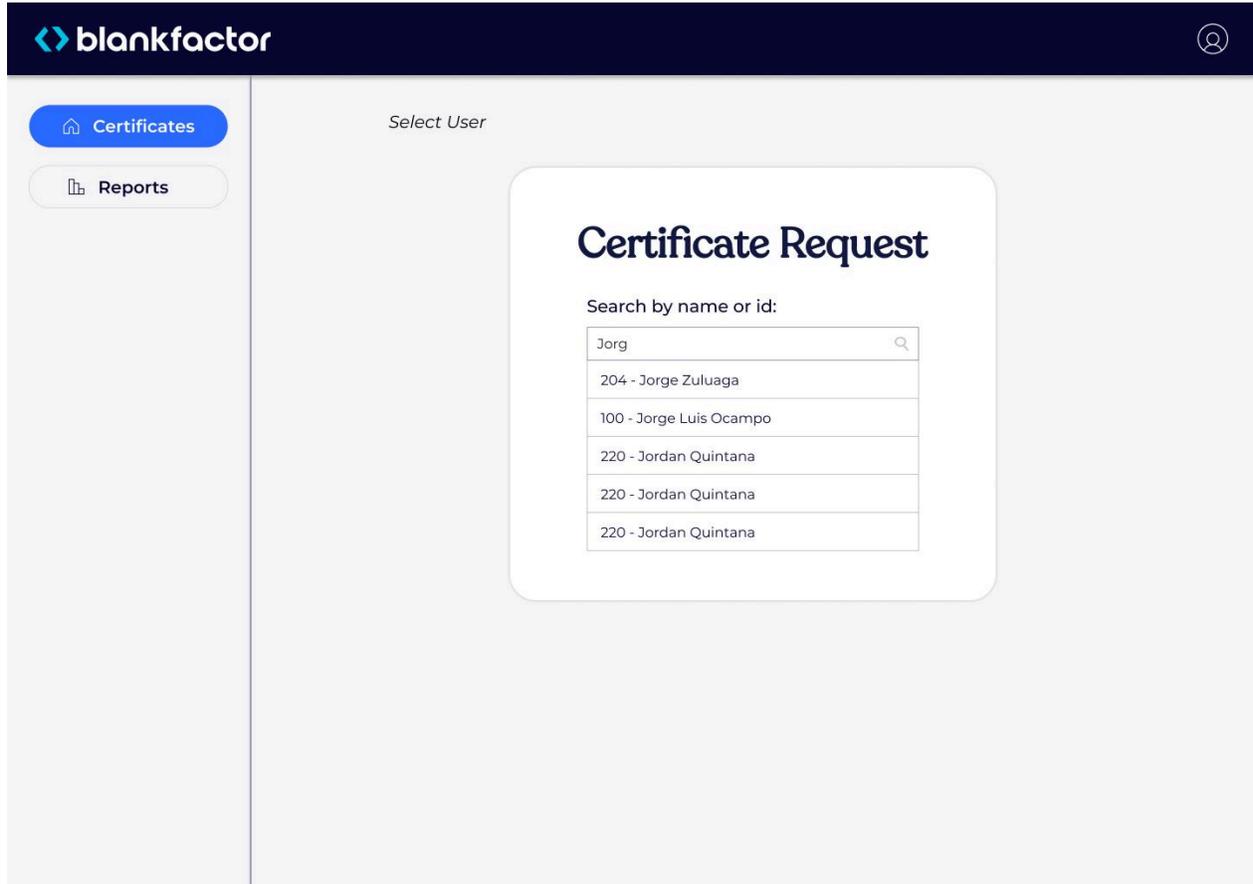


The screenshot shows a web application interface for generating certificates. At the top, there is a dark blue header with the 'blankfactor' logo on the left and a search icon on the right. Below the header, the page title 'Generate Certificates' is displayed. The main content area features a white rounded rectangle titled 'Certificate Request'. Inside this rectangle, there are several form fields: a dropdown menu for 'Certificate type' with 'Work certificate' selected and a red border around it; a dropdown menu for 'Language' with 'Select Language:' selected; a dropdown menu for 'ID type' with 'C.C.' selected; a text input field for 'To whom the certificate is addressed: (Optional)' with 'Company Name' entered; and a checkbox for 'Include Salary (Optional)' which is unchecked. At the bottom of the form is a greyed-out button labeled 'Generate Certificate' with a downward arrow icon.

Una vez que toda la información haya sido correctamente diligenciada en el formulario, el botón cambiará de color. Este cambio sirve como indicativo visual de que todos los campos han sido llenados adecuadamente y no existen errores, señalando que el formulario está listo para proceder con la generación del certificado.

Las validaciones aplicadas en el formulario son consistentes tanto para administradores como para empleados, manteniendo la misma lógica en el estado de los botones y en la señalización de errores. En cuanto al flujo específico para administradores, el primer paso consiste en seleccionar al empleado que recibirá el certificado (Ver figura 14).

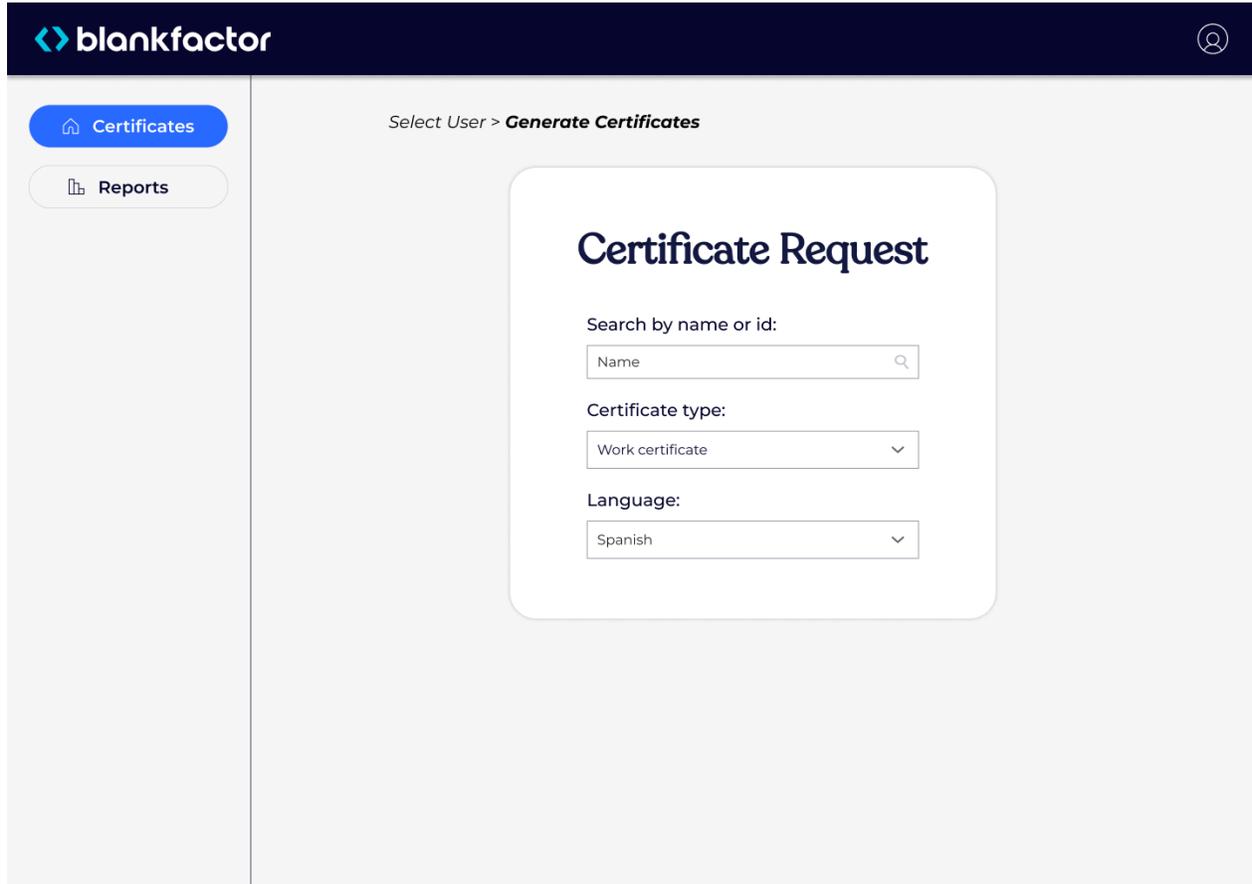
Figura 14. Vista selector de empleado para generación de certificado para administradores



Para facilitar la búsqueda de empleados por parte del administrador se desarrolló un componente de autocompletado. Este componente muestra todos los nombres de los empleados junto con su respectivo Workday ID, el identificador único de cada empleado en la compañía. Esta funcionalidad permite al administrador buscar empleados por nombre, apellido o Workday ID, ofreciendo acceso rápido y eficiente a la información relevante del empleado.

Una vez que se selecciona el empleado al cual se le va a generar el certificado, el proceso sigue un flujo similar al del usuario estándar. Se muestran en pantalla los tipos de certificados disponibles para el empleado seleccionado, así como la opción de elegir el idioma en el que se emitirá el certificado. (Ver figura 15)

Figura 15. Vista generación de certificados para administradores



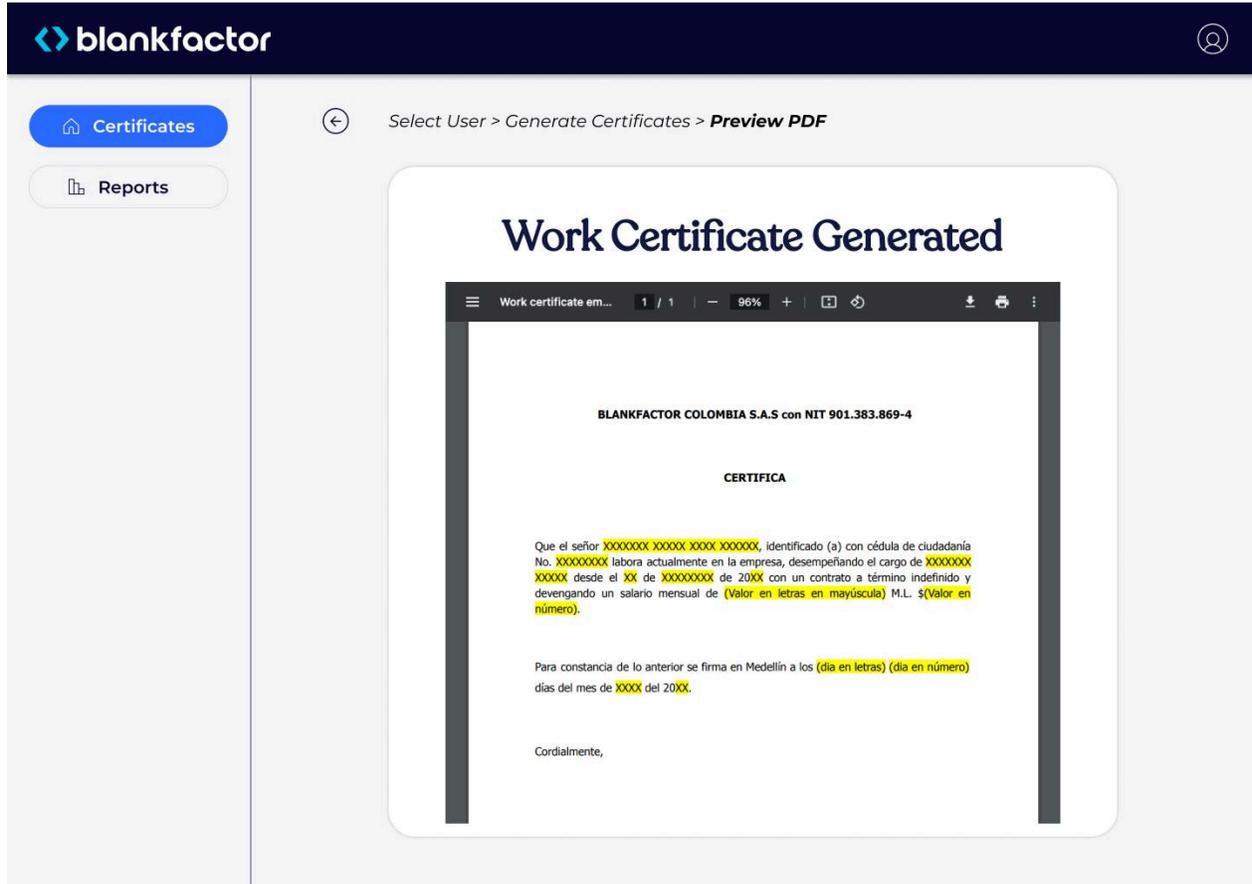
The screenshot shows the 'Certificate Request' form in the Blankfactor application. The interface includes a dark blue header with the 'blankfactor' logo and a user profile icon. A left sidebar contains 'Certificates' and 'Reports' buttons. The main content area is titled 'Select User > Generate Certificates' and features a white rounded rectangle containing the form. The form has the following fields:

- Search by name or id:** A text input field with a search icon.
- Certificate type:** A dropdown menu with 'Work certificate' selected.
- Language:** A dropdown menu with 'Spanish' selected.

A continuación, y en línea con el proceso ya descrito, una vez que se ha seleccionado el tipo de certificado, el sistema procede automáticamente a renderizar los campos específicos asociados a dicho certificado. Esta funcionalidad asegura que se presenten al usuario solo las opciones pertinentes y necesarias para el certificado elegido.

Una vez finalizado y verificado el correcto funcionamiento del formulario, se realizó la parte de renderizado de la interfaz. Esta fase implicó la integración con el backend, enviando los datos recopilados para que el backend procesara y generara el certificado correspondiente. Como resultado, se recibió el certificado en formato PDF, lo cual permitió utilizar una etiqueta nativa del navegador en el frontend para la visualización del documento. Esta implementación facilita al usuario la visualización, impresión o descarga del certificado directamente desde la interfaz de la aplicación. (Ver figura 16)

Figura 16. Vista certificado generado en PDF

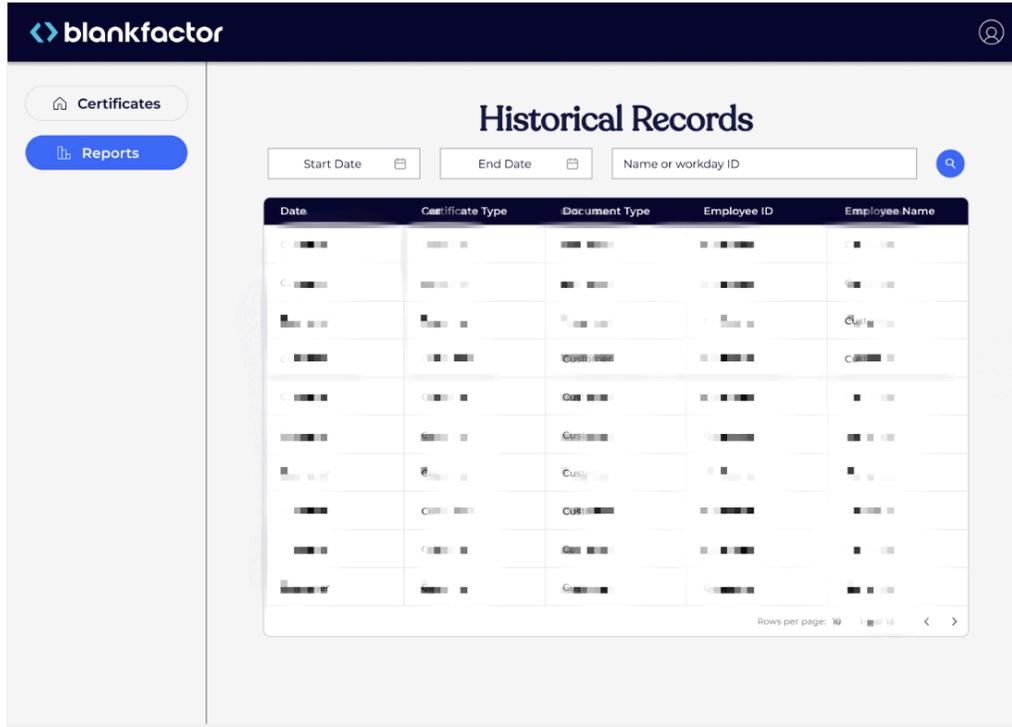


Con la finalización de estos pasos, se logró completar con éxito tanto el módulo de login como el de certificados.

4.3.3 Módulo de Certificados

El desarrollo de este módulo se realizó siguiendo fielmente el diseño previamente aprobado por el product owner. La interfaz de usuario diseñada para este módulo se presenta de la siguiente manera. (Ver figura 17)

Figura 17. Vista de reportes



The screenshot shows the 'Historical Records' page in the blankfactor application. The page has a dark blue header with the 'blankfactor' logo and a user profile icon. A sidebar on the left contains a 'Certificates' button and a highlighted 'Reports' button. The main content area is titled 'Historical Records' and features a search bar with three input fields: 'Start Date', 'End Date', and 'Name or workday ID'. Below the search bar is a table with the following columns: 'Date', 'Certificate Type', 'Document Type', 'Employee ID', and 'Employee Name'. The table contains several rows of data, with some cells containing placeholder text like 'C...' and 'E...'. At the bottom right of the table, there is a pagination control showing 'Rows per page: 10' and navigation arrows.

La tabla de reportes en el módulo de certificados incorpora un sistema de paginación, optimizando así la experiencia del usuario y la eficiencia al visualizar grandes cantidades de datos. Esta funcionalidad es especialmente útil para los administradores, quienes pueden manejar fácilmente la información relacionada con la emisión de certificados, facilitando la tarea de generar informes mensuales sobre la cantidad y tipo de certificados emitidos.

Además, el módulo incluye un formulario diseñado para aplicar filtros a los reportes. Este formulario permite a los usuarios buscar reportes específicos basándose en criterios como la fecha de inicio, la fecha final o el nombre del empleado.

En cuanto al aseguramiento de la calidad, el equipo de QA se encargó de realizar pruebas exhaustivas en todos los componentes del módulo. Su labor fue crucial para garantizar que el módulo cumpliera con las expectativas del diseño inicial, los requisitos del cliente reflejados en las historias de usuario y el correcto funcionamiento de cada funcionalidad. Paralelamente, el equipo de desarrollo se esforzó por mantener un estándar alto en las pruebas unitarias, apuntando a un porcentaje de cobertura superior al 85%. (Ver figura 18)

Figura 18. Cobertura de código de la aplicación

File	% Stmts	% Branch	% Funcs	% Lines
All files	87.12	42.36	58.82	88.23
atoms/Loader	100	100	100	100
index.tsx	100	100	100	100
style.ts	100	100	100	100
molecules/AlertSnackbar	100	100	100	100
helper.ts	100	100	100	100
index.tsx	100	100	100	100
molecules/Breadcrumb	81.25	77.77	40	83.87
index.tsx	80	77.77	40	82.75
styles.ts	100	100	100	100
molecules/DatePickerFilter	81.81	100	33.33	90
index.tsx	80	100	33.33	88.88
styles.ts	100	100	100	100
molecules/FormAutocomplete	88.23	53.84	100	88.23
index.tsx	86.66	53.84	100	86.66
styles.ts	100	100	100	100
molecules/FormSelect	81.25	40	75	86.66
index.tsx	78.57	40	75	84.61
styles.ts	100	100	100	100
molecules/UserModal	97.36	50	75	97.36
index.tsx	97.22	50	75	97.22
styles.ts	100	100	100	100
organisms/CertificateForm	81.35	11.11	25	81.35
helper.ts	100	100	100	100
index.tsx	80	11.11	25	80
styles.ts	100	100	100	100
organisms/Navbar	100	75	100	100
index.tsx	100	75	100	100
styles.ts	100	100	100	100
organisms/PreviewCertificate	81.81	36.36	66.66	84.37
index.tsx	80.64	36.36	66.66	83.33
styles.ts	100	100	100	100
organisms/RenderFormControl	80	18.18	40	80
fieldTypes.ts	100	100	100	100
index.tsx	76.19	18.18	40	76.19
styles.ts	100	100	100	100
organisms/Reports	86.76	47.82	62.5	86.76
helper.ts	100	100	100	100
index.tsx	85.93	47.82	62.5	85.93
styles.ts	100	100	100	100
organisms/Sidebar	95.12	40	83.33	95
helper.ts	100	100	100	100
index.tsx	94.28	40	83.33	94.11
styles.ts	100	100	100	100
pages/ErrorPages	92.85	100	50	92.85
index.tsx	91.66	100	50	91.66
styles.ts	100	100	100	100

Análisis

En el transcurso de nuestro proyecto de prácticas, nos enfrentamos a desafíos significativos y logros notables. A lo largo del proyecto, se implementaron con éxito módulos esenciales como el de login, certificados y reportes, demostrando una sólida comprensión y aplicación de tecnologías modernas de desarrollo web.

El módulo de login, integrando la autenticación de Google, y el módulo de reportes, con su avanzada interfaz y funcionalidades como la paginación y filtros personalizados, son ejemplos destacados de nuestra capacidad para desarrollar soluciones centradas en el usuario. El uso efectivo de herramientas como React Hook Form y la integración con el backend para la generación de certificados en formato PDF demuestran una habilidad notable para manejar aspectos complejos de desarrollo de software.

Sin embargo, un aspecto que quedó pendiente fue el desarrollo de un módulo de plantillas personalizables para los certificados. A pesar de que este módulo era una parte deseada del proyecto, limitaciones en tiempo hicieron que su implementación quedará fuera del alcance de esta fase. Es importante destacar, sin embargo, que se han establecido bases sólidas para su futura integración, en particular, la flexibilidad del formulario, diseñado para ser renderizado por el backend, prepara el terreno para una eventual inclusión de plantillas personalizables. Esto asegura que, cuando se desarrolle este módulo, la integración con los componentes existentes será fluida y coherente.

El hecho de no haber completado el módulo de plantillas personalizables en esta etapa no debe verse como un fracaso, sino como una oportunidad de mejora y una puerta abierta para el desarrollo futuro. Este aspecto resalta la importancia de la adaptabilidad y la planificación en proyectos de desarrollo de software, donde a menudo es necesario reajustar los objetivos y alcances en función de los recursos y el tiempo disponibles.

En conclusión, los resultados obtenidos en esta práctica demuestran un progreso significativo en áreas clave del desarrollo de software, al tiempo que subrayan la necesidad de flexibilidad y

adaptación continua frente a las restricciones y desafíos del mundo real. La preparación para la futura integración del módulo de plantillas personalizables ilustra una visión a largo plazo y un compromiso con la mejora continua del proyecto.

Conclusiones

Con base en el desarrollo de este proyecto de prácticas académicas podemos destacar valiosas lecciones.

Implementación Efectiva de Tecnologías Modernas: Se logró con éxito la implementación de módulos cruciales como el de login y certificados. El uso de tecnologías avanzadas, como React, React Hook Form, y la integración de autenticación de Google, demuestra la capacidad para aplicar conocimientos técnicos en un contexto práctico.

Desarrollo de Interfaces de Usuario Centradas en el Usuario: La creación de interfaces intuitivas y la implementación de funcionalidades como la paginación y filtros personalizados en el módulo de certificados destacan el enfoque en mejorar la experiencia del usuario.

Flexibilidad y Preparación para el Futuro: Aunque el módulo de plantillas personalizables no se completó, la estructura y el diseño del proyecto se han preparado para su futura integración. Esta decisión demuestra una planificación eficiente y una visión a largo plazo, destacando la importancia de la adaptabilidad en el desarrollo de software.

Reconocimiento de Limitaciones y Oportunidades de Mejora: La incapacidad de desarrollar el módulo de plantillas personalizables en esta etapa subraya la importancia de reconocer y adaptarse a las limitaciones de tiempo y recursos. Esta experiencia proporciona una base para el aprendizaje continuo y la mejora en proyectos futuros.

Colaboración y Calidad: La colaboración efectiva entre los equipos de desarrollo y QA, logrando mantener un alto estándar de calidad en el aplicativo, refleja un compromiso sólido con la calidad del software.

En conclusión, este proyecto ha cumplido con el 85% de los objetivos planteados inicialmente, proporcionando una experiencia práctica sólida en el desarrollo de software y dejando un camino abierto para futuras mejoras y desarrollos. Las lecciones aprendidas y los éxitos alcanzados son indicativos de un progreso significativo en nuestra trayectoria como futuros

ingenieros de sistemas, y destacan la importancia de la flexibilidad, la adaptabilidad y la planificación estratégica en el campo del desarrollo de software.

Referencias

- [1] Indeed. (23 de Agosto de 2023). Certificado laboral: ¿Qué es y cómo se hace uno? Orientación Profesional. <https://www.indeed.com/orientacion-profesional/cv-cartas-presentacion/certificado-laboral-que-es>
- [2] Figma. (2023). Figma Documentation. Recuperado de <https://help.figma.com/hc/en-us/articles/14563969806359-What-is-Figma->
- [3] Williams, D. E., & Miller, P. A. (2019). Agile Web Development: Accelerating Process Automation. *Journal of Software Engineering and Applications*, 12(6), 307-322.
- [4] TypeScript. (s.f.). TypeScript: JavaScript that scales. Recuperado de <https://www.typescriptlang.org/>
- [5] Aggarwal, S. (2018). Modern Web-Development using ReactJS. *International Journal of Recent Research Aspects*.
- [6] Frost, B. (2016). *Atomic design*. Pittsburgh: Brad Frost.
- [7] Nariño, Felipe. Atuesta, Jorge (2019). Modelo De Arquitectura Para Front-End (Basado En Microfrontends) Aplicado Al Producto Digital De Fuerzaespecializada De Vivienda Del Banco De Bogotá. Recuperado de <https://repository.udistrital.edu.co/handle/11349/25071?show=full>