

Capítulo VI: Product Implementation, Validation & Deployment

6.1. Software Configuration Management.

6.1.1. Software Development Environment Configuration.

Project Management

- Trello: Trello es una herramienta web para la gestión de proyectos que funciona en la mayoría de los navegadores. Se utilizó para organizar y registrar las tareas pendientes, en proceso y completadas, lo que permitió mantener un control claro del progreso del proyecto y las responsabilidades de cada miembro del equipo.

Product UX/UI Design

- Miro: Miro fue empleado para el desarrollo de mapas de escenarios y planificación visual, facilitando la comprensión y comunicación entre los distintos segmentos objetivo.
- Figma: Figma jugó un papel fundamental en la creación colaborativa de wireframes, maquetas y prototipos para el landing page, aplicaciones web y móviles. Su acceso es gratuito mediante una cuenta registrada, lo que facilita la colaboración remota entre los miembros del equipo.

Software Development

Landing Page:

El desarrollo de la Landing Page se realizó utilizando las tecnologías fundamentales para el diseño web moderno:

- HTML5: Para la estructura semántica del contenido.
 - CSS3: Para el diseño visual, uso de media queries y animaciones básicas.
 - JavaScript: Para la interactividad, validaciones de formularios y manipulación del DOM.
- Estas tecnologías permiten una página ligera, responsive y optimizada para dispositivos móviles, garantizando una experiencia de usuario accesible y clara.

Frontend Web:

Para el desarrollo de la aplicación web se emplearon tecnologías del entorno frontend moderno:

- **HTML5**, **CSS3** y **JavaScript** como fundamentos esenciales.
- **Angular** como framework principal, permitiendo una arquitectura basada en componentes, enrutamiento interno, servicios reutilizables y vinculación bidireccional de datos.

Angular facilitó la creación de una interfaz dinámica y escalable, especialmente útil para usuarios como padres y administradores, quienes requieren acceso en tiempo real a información del transporte escolar.

Aplicacion Movil :

La aplicación móvil fue desarrollada de forma **nativa para Android** utilizando:

- **Android Studio** como entorno de desarrollo integrado (IDE).
- **Kotlin** como lenguaje principal, por su interoperabilidad con Java, su sintaxis moderna y su orientación a la seguridad.

Esta aplicación móvil complementa la experiencia web y ofrece a los padres una herramienta confiable para el seguimiento diario de sus hijos en el transporte escolar, directamente desde su smartphone.

IDE's de desarrollo

- WebStorm : Utilizado para la creación del **Landing Page** y la **aplicación web**. Este IDE, desarrollado por JetBrains, está optimizado para tecnologías web como **HTML5**, **CSS3**, **JavaScript** y frameworks modernos como **Angular**, lo que permite una codificación más eficiente, estructurada y mantenible.
- Android Studio : Fue la herramienta principal para el desarrollo de la **aplicación móvil**. Esta se implementó en **Kotlin**, siguiendo las prácticas recomendadas para el desarrollo nativo en Android. Esto garantiza una experiencia de usuario fluida y una compatibilidad robusta con las versiones más recientes del sistema operativo.
- IntelliJ IDEA : Se empleó para el desarrollo del **backend** del sistema. Este potente IDE también desarrollado por JetBrains, permite integrar múltiples tecnologías y facilita la implementación de servicios, controladores y lógica de negocio. Además, ofrece herramientas avanzadas como **depuración**, integración con **bases de datos**, y compatibilidad con herramientas de construcción como **Gradle** o **Maven**.

Software Testing

- Para verificar el correcto funcionamiento del sistema en cada una de sus partes, se realizaron diversas pruebas utilizando herramientas específicas según el entorno. En el caso de la landing page y la aplicación web, se utilizaron navegadores modernos como Google Chrome, Microsoft Edge y Mozilla Firefox, tanto en sus versiones de escritorio como móviles, lo cual permitió inspeccionar el comportamiento visual, la interacción del usuario y la correcta adaptación del diseño responsivo. Para el backend, se empleó Postman, una herramienta que permitió validar el funcionamiento de cada API desarrollada, comprobando los métodos HTTP, las respuestas recibidas, los códigos de estado y el formato de los datos enviados y recibidos. Finalmente, para la aplicación móvil, se utilizó el emulador integrado en Android Studio, que permitió simular dispositivos con distintas resoluciones y versiones de Android, lo cual facilitó la detección de errores y la comprobación de la experiencia del usuario en diferentes escenarios sin necesidad de usar un dispositivo físico.

Software Deployment

- **Firebase** : Plataforma en la nube que proporciona un conjunto completo de herramientas para el desarrollo, despliegue y alojamiento de aplicaciones web. Facilita a los desarrolladores la creación y publicación eficiente y escalable de sitios estáticos o dinámicos.

Software Documentation

- Markdown : Lenguaje de marcado ligero que permite dar formato al texto de forma sencilla utilizando texto plano. Se utiliza para documentar información relevante sobre el contexto, organización, producción y creación del proyecto.
- Structurizr: Herramienta para la creación rápida de diagramas C4 mediante una sintaxis similar a un lenguaje de programación. Requiere una cuenta registrada para su uso en la aplicación web.
- Vertabelo: Es una aplicación web colaborativa que facilita el diseño e implementación de bases de datos en una amplia variedad de motores.

6.1.2. Source Code Management.

Organización y Gestión del Trabajo

El trabajo colaborativo fue gestionado a través de la plataforma GitHub, utilizando una organización dedicada:

- **Organización:** [Desarrollo-de-Soluciones-IOT-1ASI0572](#)
 - **Repositorio de la Landing Page:** [EduGo-landing-page](#)
 - **Repositorio del FrontEnd:** [EduGo-FrontEnd](#)
-

Flujo de Trabajo con GitFlow

Para garantizar un desarrollo ordenado, se adoptó la estrategia de ramas **GitFlow**, que permite un control riguroso del ciclo de vida del software.

Ramas Principales

- **main**
 - Contiene versiones estables listas para producción.
 - Solo recibe cambios desde ramas **release** o **hotfix**.
- **development**
 - Rama de integración para las funcionalidades en desarrollo.
 - Se parte de esta rama para crear nuevas funcionalidades (**feature**).

Ramas Secundarias

- **feature/<nombre_funcionalidad>**
 - Usadas para implementar nuevas características.
 - Se crean desde **development**.
 - Una vez completadas, se fusionan nuevamente con **development**.
 - Ejemplo: **feature/add-analytics-dashboard**

6.1.3. Source Code Style Guide & Conventions.

Para el desarrollo de futuros sprints se utilizarán las siguientes convenciones en el código:

Idioma:

- La documentación y elaboración de informes se realizarán en español.
- Los nombres de archivos, variables, clases, funciones y constantes en HTML, CSS, JavaScript, TypeScript y Java estarán en inglés.

Clases y funciones:

- Los identificadores para clases y funciones se escribirán en inglés.

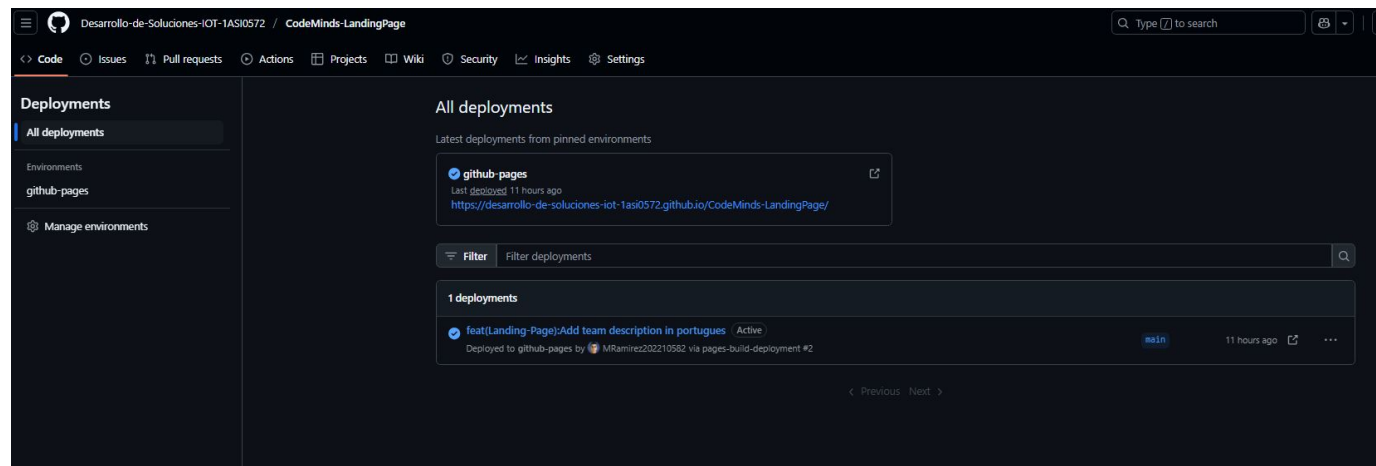
Comentarios y documentación del código:

- Se promoverá el uso de comentarios claros y breves en español, para explicar el propósito y funcionamiento de las partes relevantes del código, facilitando así la comprensión para todo el equipo de desarrollo.

6.1.4. Software Deployment Configuration.

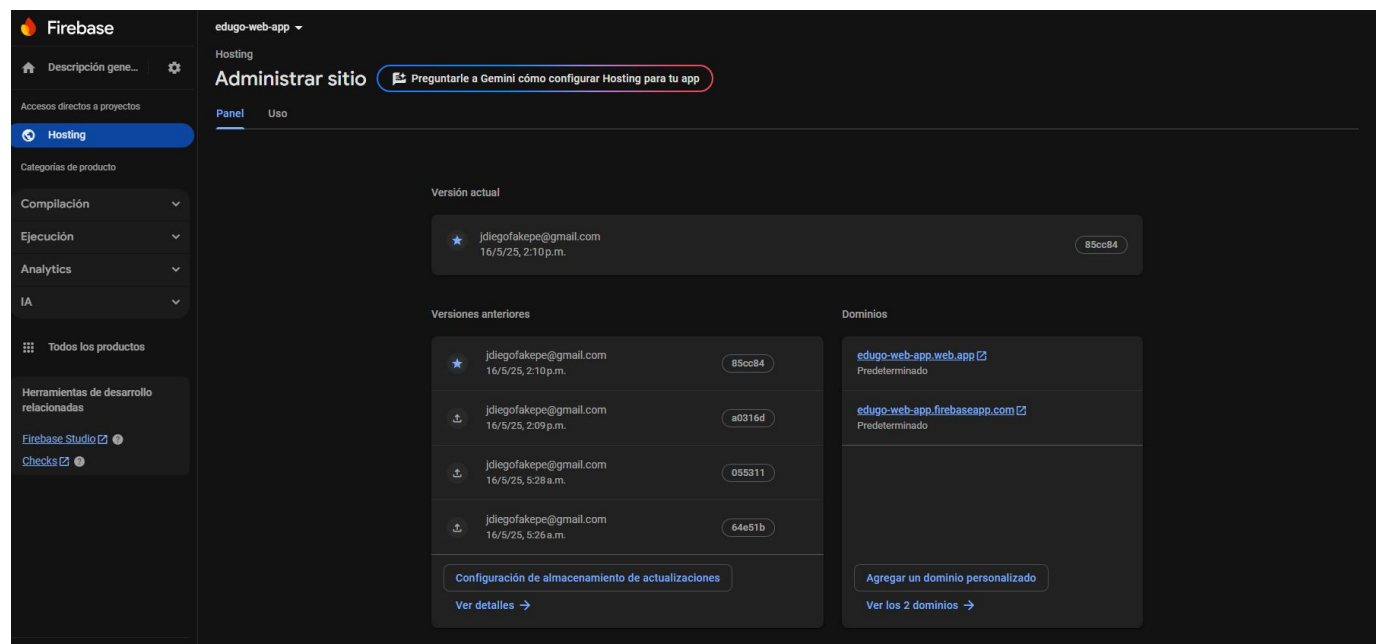
Landing Page

La landing page fue desplegada utilizando **GitHub Pages**, un servicio gratuito que permite alojar sitios web estáticos directamente desde un repositorio de GitHub. Este servicio es ideal para proyectos de código abierto y proporciona una forma sencilla de compartir el trabajo con el público.



Aplicación Web

La aplicación web fue desplegada utilizando **Firebase Hosting**, un servicio de Google que permite alojar aplicaciones web de forma rápida y segura. Firebase Hosting es ideal para aplicaciones de una sola página (SPA) y proporciona características como SSL automático, integración con otras herramientas de Firebase y un CDN global para mejorar la velocidad de carga.



6.2. Landing Page, Services & Applications Implementation.

6.2.1. Sprint 1

6.2.1.1. Sprint Planning 1.

Sprint #	Sprint 1
----------	----------

Sprint #	Sprint 1
Sprint Planning Background	
Date	2025-05-02
Time	20:00 PM
Location	Virtual meeting via Discord
Prepared By	Angel Antonio Cancho Corilla / Juan Diego Cueto Dominguez / Scott Jacobo Huachaca Advincula / Marcelo Sebastian Ramirez Ramirez / Paolo Del Carmen Martinez Villanueva / Xiao Lian Li Zegarra
Attendees (to planning meeting)	Angel Antonio Cancho Corilla / Juan Diego Cueto Dominguez / Scott Jacobo Huachaca Advincula / Marcelo Sebastian Ramirez Ramirez / Paolo Del Carmen Martinez Villanueva / Xiao Lian Li Zegarra
Pre-Sprint Review Summary	During the preparation phase, the sprint bases were defined, focusing on the development of the Landing Page and the business logic applied to the Front-End design of the web application.
Initial Sprint Retrospective Summary	The objectives set in Sprint 1 were met, however, opportunities for improvement in time and task management were identified.
Sprint Goal & User Stories	
Sprint 1 Goal	Our focus is on delivering a clear and simple landing page that communicates the main value of our service. Also an interface, that allows administrators, to visualize business related information and analytics.We believe the mentioned functionalities development will increase the value in the experience for our users.This will be confirmed when new visitors learn more about our service and users are using the implemented features
Sprint 1 Velocity	
Sum of Story Points	16 Story Points

6.2.1.2. Aspect Leaders and Collaborators.

Leadership and Collaboration Matrix (LACX)

Team Member (Last Name, First Name)	GitHub Username	Landing Page Development (L/C)	Business Logic for Front-End (L/C)	Administration Interface (L/C)	Data Visualization (L/C)	Analytics (L/C)

Team Member (Last Name, First Name)	GitHub Username	Landing Page Development (L/C)	Business			
			Logic for Front-End (L/C)	Administration Interface (L/C)	Data Visualization (L/C)	Analytics (L/C)
Cancho Corilla, Angel Antonio	angerlessdev	L	C	C	C	L
Cueto Dominguez, Juan Diego	JDu202012207	C	L	C	L	C
Huachaca Advincula, Scott Jacobo	Scott-Huachaca	C	C	L	C	C
Ramirez Ramirez, Marcelo Sebastian	MRamirez202210582	C	C	C	C	L
Del Carmen Martinez Villanueva, Paolo	paolomv02	C	L	C	C	C
Li Zegarra, Xiao Lian	XLianLZ	C	C	L	C	C

6.2.1.3. Sprint Backlog 1.

Sprint 1

User Story		Work-Item / Task					Status (To-do / In-Process / To-Review / Done)
Id	Title	Id	Title	Description	Estimation (Hours)	Assigned To	

User Story		Work-Item / Task					
HU10	Generación de reportes automáticos	TA01	Lógica para la generacion de reportes	Programacion de la logica vinculada a la generacion de reportes	5 horas	Angel Cancho-Paolo Martinez	Done
HU45	Visualización del historial de incidentes	TA02	Desarrollo del listado de incidentes	Listado de incidentes recibidos de un endpoint	2.5 horas	Juan Diego Cueto	Done
		TA03	Vista de Incidentes	Desarrollo de la vista de que muestra informacion detallada de la lista de incidentes	2.5 horas	Scott Huachaca	Done
HU52	Visualizacion de proposito y beneficios del servicio	TA04	Navegacion	Implementacion de una barra de navegacion	1 hora	Xiao Li	Done
		TA05	Contenido Principal	Insercion de informacion en la landing page	2 horas	Xiao Li	Done
HU53	Formulario de Contacto	TA06	Formulario	Desarrollo de formulario con campos de contacto(nombre, correo, mensaje, etc.)	2 horas	Marcelo Ramirez	Done

6.2.1.4. Development Evidence for Sprint Review.

Para este primer sprint se decidió trbajar en una rama diferente para después unirlo todo en develop. Cada miembro ha realizado un commit para luego hacer merge.

Repository	Branch	Commit ID	Commit Message	Committed on(date)
CodeMinds-LandingPage	main	de87e5cb957edb9c71d78843c64ff5b761060c2b	fix(landing-page): set english as default language	13/05/2025
CodeMinds-LandingPage	main	3bfbb3898b963e1e82bd4551dfaae948e3b1504f	feat(landing-page): add language configuration file	13/05/2025
CodeMinds-LandingPage	main	3b1c3c3320f227ac5f1ab5d909b52c19414b1fcd	feat(landing-page): style sheet initial commit	13/05/2025

Repository	Branch	Commit ID	Commit Message	Committed on(date)
CodeMinds-LandingPage	main	0093fdb53adc93987efdc078f1d54d1d49024dd3	feat(landing-page): script file initial commit	13/05/2025
CodeMinds-LandingPage	main	4cf928b3881c5245f13d654fb9b879f57f2c830c	feat(landing-page): html file initial commit	13/05/2025
CodeMinds-LandingPage	main	67d79de6195f29a88e98b6614db9878c3b22bbd7	feat(Landing-Page): Add team description in portugues	15/05/2025
CodeMinds-LandingPage	main	e4592aab9c36bcfd1e116e55597a408ad2ea5e08	feat(Landing-Page): add team description translation	15/05/2025
CodeMinds-LandingPage	main	790ce5d99a62f0be973d0fffb3ef3e9738df9553	feat(Landing-Page): add Team photo	15/05/2025
CodeMinds-LandingPage	main	8d945be380998242fd8dafa0e6d25f0abe678bb5	feat(Landing-Page): add team profile images	15/05/2025
CodeMinds-LandingPage	main	b93812f67ca125711b99050460703265f26fc929	chore(landing-page): manage contact and team translations	15/05/2025
CodeMinds-LandingPage	main	dfc49155d29e9517508dfa4516f63524cc51a9b2	chore(landing-page): contact and team styles	15/05/2025
CodeMinds-LandingPage	main	eb128daedcdf4758eb5710c8751e7770a11aa2cf	chore(landing-page): add contact and team references to the home	15/05/2025
CodeMinds-LandingPage	main	2b6750085efaf5a521b69be888e9d8eacf4581f6	feat(landing-page): add team html file	15/05/2025
CodeMinds-LandingPage	main	cfe4749e33cbd6eebd0e3b76ffaf629ee433f2aa	feat(landing-page): add contact html file	15/05/2025

Repository	Branch	Commit ID	Commit Message	Committed on(date)
CodeMinds-LandingPage	main	22ef41e8c68294a78d49acae7b62fbd6b90f7bdc	chore(landing-page): set english as default language	15/05/2025
edugo-web-application	develop	b8307949e7b3234859c51b4ccf784a6959713c58	feat(dashboard): add view dashboard	15/05/25
edugo-web-application	develop	35dd827d518592fdab57c292c46249fa541d1411	feat(develop): add header component	15/05/25
edugo-web-application	main	2c4ecee3371b8c08c8944ad06a791beb948494ba	docs: update readme with installation instructions	15/05/25
edugo-web-application	main	117a7e4348fd934cbfac10a7766f4f80cca3ee04	chore: initial commit	15/05/25
edugo-web-application	develop	87a4baf387429c8bd1e8309b153ababfe861d8c3	refactor: update navigation routes with semantic rules	16/05/25
edugo-web-application	develop	025e6e36dd321a23530898339d6ba9ecdca36a82	feat(profiles): add component for profile	16/05/25
edugo-web-application	develop	2b2888b8dacc6ad889d6e8c6395a66c15187602a	fix(develop): fix merge errors	16/05/25
edugo-web-application	feature/add-reports-table	fe48c3d8be5a2120fb85b1ff60610cced0431a57	feat(add-reports-table): merged in develop branch	16/05/25
edugo-web-application	feature/add-analytics-dashboard	efcd0c7f72f4eddb0aa078419bdeccc544543940	refactor(analytics): update models and services for new fake API data	16/05/25
edugo-web-application	develop	79e054aa940d0e0b7cb05354505a92699aa3cde1	refactor(iam): add logo icon to sign-in and route login to dashboard	16/05/25
edugo-web-application	develop	f2237a290593f733df798f66e851da39b04dd133	feat(iam): add sign-in page and set login as principal route	16/05/25

Repository	Branch	Commit ID	Commit Message	Committed on(date)
edugo-web-application	develop	fef1688972539792753109a14dca2df9206c4ee3	chore(update): update api url	16/05/25
edugo-web-application	feature/add-reports-table	4f3143818ddce8c519eb2ae4c9d625fd11b86fa5	feat(add-reports-table): add table filter	16/05/25
edugo-web-application	feature/add-reports-table	ce0fe99ebbdd4f3c38081342f6bcfce57be3c6d	feat(add-reports-table): add incidents model	16/05/25
edugo-web-application	feature/add-reports-table	43f8f6d6c076908c8f2bccd7541ab074417ce025	feat(add-reports-table): add reports table	16/05/25

6.2.1.5. Testing Suite Evidence for Sprint Review.

Repository	Branch	Commit ID	Commit Message	Committed on(date)
CodeMinds-LandingPage	main	67d79de6195f29a88e98b6614db9878c3b22bbd7	feat(Landing-Page):Add team description in portugues	16/05/25
edugo-web-application	main	87a4baf387429c8bd1e8309b153ababfe861d8c3	refactor: update navigation routes with semantic rules	16/05/25

6.2.1.6. Execution Evidence for Sprint Review.

Para el landing page Contact <https://desarrollo-de-soluciones-iot-1asi0572.github.io/CodeMinds-LandingPage/>

Para el FrontEnd Contact <https://edugo-web-app.web.app/sign-in>

Enlace del video de evidencia: https://upcedupe-my.sharepoint.com/:g/personal/u202010039_upc_edu_pe/EYbOPwF2r8dKos3HTZF4OBEBfZsGa5nbYHwv3INGKTWRkA?e=G3PEIV&nav=eyJyZWZlcnJhbEluZm8iOjE0OTUyMjYyZWZlcnJhbFZpZXciOiJTaGFyZURpYWxvZy1MaW5rliwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXcifX0%3D 🙌

6.2.1.7. Services Documentation Evidence for Sprint Review.

A continuación se presenta la documentación de los servicios de la aplicación web, incluyendo la descripción de los endpoints disponibles.

Endpoint	Http Verb	Accion	Descripcion
----------	-----------	--------	-------------


Endpoint	Http Verb	Accion	Descripcion
https://jdu202012207.github.io/pruebas-api/drivers.json	GET	Obtener conductores	Devuelve información de los conductores registrados, incluyendo datos personales, número de licencia, vehículo asignado y estudiantes a su cargo.
https://jdu202012207.github.io/pruebas-api/parents.json	GET	Obtener apoderados	Devuelve información de los padres de familia registrados, incluyendo nombre, DNI, número telefónico, foto y los estudiantes a su cargo
https://jdu202012207.github.io/pruebas-api/students.json	GET	Obtener estudiantes	Devuelve información de los estudiantes registrados, incluyendo su nombre, colegio, grado, código RFID, foto y relación con su apoderado y conductor asignado.
https://jdu202012207.github.io/pruebas-api/analytics-drivers.json	GET	Obtener analíticas	Devuelve analíticas de rendimiento de los conductores, incluyendo resumen de incidentes (desvíos, tardanzas, excesos de velocidad), tiempos de llegada y distancias recorridas por día.

6.2.1.8. Software Deployment Evidence for Sprint Review.

Para el sprint presentado de la landing page y FrontEnd se optó por varias herramientas para su desarrollo.

- *Git*: Se utilizó para el control de versiones del código fuente.
- *GitFlow*: Se utilizó para ver el avance de los integrantes del equipo.
- *GitHub*: Se utilizó para crear el repositorio de la landing page, donde se subió el código fuente.
- *Angular*: Se utilizó para crear el web app funcional con la ayuda de un fakeapi
- *GitHub Pages*: Es la plataforma donde se desplegó la landing page. Se eligió por ser una solución gratuita, rápida y sencilla que permite publicar directamente desde el repositorio de GitHub.

Deployamiento del landing Para el despliegue de la landing page, se configuró el servicio de GitHub Pages siguiendo los siguientes pasos:

1. Se accedió al repositorio en la plataforma de GitHub.
2. En la parte superior del repositorio, se hizo clic en la pestaña **"Settings"**.
3. En el menú lateral izquierdo, se seleccionó la opción **"Pages"**.
4. En la sección **"Source"**, se eligió la rama **main** como origen y la carpeta raíz (/) como directorio de publicación.
5. Finalmente, se hizo clic en **"Save"** para guardar los cambios y activar el despliegue. Contact

6.2.1.9. Team Collaboration Insights during Sprint.

Landing Page Contact
FrontEnd Contact