



**Análisis, diseño e implementación de aplicación Front-end para Sistema de matrículas de
Ingeni@**

Juan Fernando Ríos Franco

Trabajo de grado presentado para optar el título de Ingeniero de Sistemas

Asesores

Sandra Patricia Zabala Orrego, Especialista (Esp) en gerencia de proyectos

Clara Lucía Monsalve Ríos, Magíster (MSc) en Educación

Universidad de Antioquia

Facultad de Ingeniería

Ingeniería de Sistemas

Medellín, Colombia

2023

Cita	Ríos Franco [1]
Referencia Estilo IEEE (2020)	[1] J. F. Ríos Franco, “Análisis, diseño e implementación de aplicación Front-end para Sistema de matrículas de Ingeni@”, Trabajo de grado profesional, Ingeniería de sistemas, Universidad de Antioquia, Medellín, 2023.



Centro de Documentación Ingeniería (CENDOI).

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

Universidad de Antioquia - www.udea.edu.co

Rector: John Jairo Arboleda Céspedes.

Decano / Director: Julio César Saldarriaga Molina

Jefe departamento: Diego José Luis Botía Valderrama.

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

TABLA DE CONTENIDO

RESUMEN.....	6
ABSTRACT.....	7
I. INTRODUCCIÓN.....	8
II. PLANTEAMIENTO DEL PROBLEMA.....	9
III. OBJETIVOS.....	10
A. Objetivo general	10
B. Objetivos específicos	10
IV. MARCO TEÓRICO	11
V. METODOLOGÍA.....	13
VI. RESULTADOS.....	15
A. Módulo administrativo	15
B. Módulo estudiante	21
VII. CONCLUSIONES.....	25
REFERENCIAS	26

LISTA DE FIGURAS

Fig. 1. Diagrama de componentes de la aplicación web.....	15
Fig. 2. Pantalla de autenticación de usuarios administradores.....	16
Fig. 3. Pantalla de lista de proyectos.....	16
Fig. 4. Pantalla de lista de cohortes de un proyecto.....	17
Fig. 5. Pantalla de lista de cursos de un proyecto	17
Fig. 6. Pantalla de configuración de parámetros de un proyecto	18
Fig. 7. Formulario de creación/edición de campos de estudiantes de un proyecto.....	18
Fig. 8. Pantalla de configuración de estilos, políticas e imágenes de proyecto	19
Fig. 9. Pantalla de información y configuración de cohorte	19
Fig. 10. Pantalla de lista de estudiantes de una cohorte.....	20
Fig. 11. Pantalla de detalle de estudiante de una cohorte	20
Fig. 12. Pantalla de grupos de un curso en una cohorte.....	21
Fig. 13. Ejemplo de inicio de sesión para estudiantes de un proyecto.....	21
Fig. 14. Ejemplo de política de privacidad de un proyecto	22
Fig. 15. Ejemplo de cohortes activas para un estudiante de un proyecto	22
Fig. 16. Ejemplo de calendario académico de una cohorte.....	23
Fig. 17. Ejemplo de oferta de cursos para un estudiante	23
Fig. 18. Ejemplo de constancia de matrícula de un estudiante	24

SIGLAS, ACRÓNIMOS Y ABREVIATURAS

DRAI	Departamento de Recursos de Apoyo e Informática
SIMAT	Sistema integrado de matrícula
MinEducación	Ministerio de educación
E2E	End to End (Pruebas de punta a punta)
UdeA	Universidad de Antioquia

RESUMEN

Ingeni@, una unidad del Departamento de Recursos de Apoyo e Informática de la Facultad de Ingeniería de la Universidad de Antioquia es la encargada de gestionar las matrículas de estudiantes de diferentes proyectos tanto internos como lo es los nivelatorios de la facultad o externos como Misión-TIC o Ubicua, es por esto que la práctica académica se enfocó en el levantamiento de requisitos, creación de mockups, selección de tecnologías y desarrollo de un aplicativo web que permite por medio de un rol administrativo, la creación de proyectos y la gestión de cohortes en cada proyecto, carga de estudiantes, cursos y grupos ofertados, para que, por medio de un enlace único para cada proyecto y que es enviado a los estudiantes que se vinculan a una cohorte, estos puedan ingresar de acuerdo al calendario estipulado, a realizar la matrícula a los grupos disponibles al momento del ingreso y que finalmente se puedan extraer reportes de estudiantes matriculados e índices de ocupación de los grupos.

Palabras clave — matrículas, desarrollo web, scrum, ReactJs, Redux Toolkit.

ABSTRACT

Ingeni@, a unit of the Department of Support Resources and Informatics of the Faculty of Engineering of the Universidad de Antioquia is responsible for managing the enrollment of students in different projects both internal and external as is the leveling of the faculty or external as Misión-TIC or Ubicua, which is why the academic practice focused on the collection of requirements, creation of mockups, selection of technologies and development of a web application that allows through an administrative role, the creation of projects and the management of cohorts in each project, loading of students, courses and groups offered, so that, through a unique link for each project and that is sent to students who are linked to a cohort, they can enter according to the stipulated calendar, to enroll in the groups available at the time of entry and that finally reports of enrolled students and occupancy rates of the groups can be extracted.

Keywords — **enrollments, web development, scrum, ReactJs, Redux Toolkit.**

I. INTRODUCCIÓN

El Departamento de Recursos de Apoyo e Informática DRAI de la Universidad de Antioquia es una unidad administrativa de la Facultad de Ingeniería y presta servicio a todos los departamentos académicos y administrativos de la Facultad y de la Universidad [1]. La virtualidad, uno de sus frentes de trabajo a cargo de la unidad Ingeni@, se encarga de apoyar, acompañar y capacitar a los docentes de la Facultad de Ingeniería en el diseño, desarrollo y montaje de recursos educativos que dinamicen los procesos de enseñanza y aprendizaje en los programas de pregrado y posgrado.

Ahora bien, dicha unidad se encarga desde hace algún tiempo de gestionar las matrículas de estudiantes de diferentes proyectos tanto internos como los nivelatorios o externos como Misión-TIC o Ubicua, es por lo anterior que surgió la posibilidad de realizar la práctica académica en el Departamento de Recursos de Apoyo e Informática participando en el levantamiento de requerimientos, diseño y construcción de una aplicación web que permitiera la gestión de matrículas a proyectos, cohortes y cursos creados de forma dinámica.

Finalmente, las funciones que se desempeñaron durante la práctica académica estuvieron principalmente relacionadas con el análisis, diseño e implementación del Front-end de la aplicación.

II. PLANTEAMIENTO DEL PROBLEMA

Ingeni@, una de las unidades del Departamento de Recursos de Apoyo e Informática, es la encargada de gestionar el proceso de inscripción de estudiantes a diferentes cursos que se imparten en la universidad pertenecientes a proyectos alternos a la programación habitual, con el paso del tiempo y con la asignación de proyectos como Misión-TIC y Ubicua, el proceso de inscripción y matrículas se volvía tedioso y demandante, es por ello que se identificó la necesidad de un sistema informático que permitiera gestionar la oferta y matrícula de estudiantes a cursos para los diferentes proyectos que se han manejado y que se puedan manejar en el futuro desde Ingeni@.

III. OBJETIVOS

A. Objetivo general

Implementar un aplicativo web de matrículas que permita de forma dinámica y flexible, la creación y configuración de proyectos, sus cohortes y cursos, así como los horarios, docentes, matrículas de estudiantes y reportes.

B. Objetivos específicos

- Diseñar la visualización y organización del contenido del sistema de matrículas en la Web.
- Caracterizar y seleccionar tecnologías a utilizar en el desarrollo.
- Gestionar los requisitos funcionales y no funcionales y definir la arquitectura del software.
- Desarrollar e implementar la arquitectura y diseño del aplicativo cliente (Front-end).
- Asegurar la calidad de código durante el desarrollo del sistema mediante pruebas unitarias, de integración y e2e.

IV. MARCO TEÓRICO

Toda persona, compañía y en general toda organización, continuamente está captando una colección de datos, gran parte de los cuales no tienen significado alguno para ella, pero también existen otros datos que le sirven para conocer mejor el entorno que le rodea y también para conocerse mejor y plantear índices de desempeño. Estos datos, que constituyen la llamada información, le van a permitir tomar decisiones más acertadas. Por ello, la información a tiempo y en la cantidad precisa es un factor clave para toda organización. Para el caso de las instituciones educativas, es de vital importancia tener información a tiempo y precisa de sus estudiantes, cursos, grupos y horarios, dicha información, por ejemplo, se obtiene durante el proceso de matrícula que es el conjunto de políticas, procedimientos y actividades, que permiten organizar la continuidad de los alumnos antiguos y el ingreso de alumnos nuevos a una institución [2], abordando generalmente las siguientes etapas:

- Determinación de la oferta educativa.
- Planeación de la continuidad de los alumnos.
- Pre-matrícula y reserva de cupo de los alumnos matriculados.
- Asignación o cupos de traslado.
- Inscripción de alumnos nuevos.
- Ajuste de los alumnos matriculados por repotencia.
- Matrícula y liberación de cupos.

El sistema integrado de matrícula SIMAT es una herramienta del Ministerio de Educación Nacional, que permite organizar y controlar el proceso de matrícula en todas sus etapas, así como tener una fuente de información confiable y disponible para la toma de decisiones, la cual está diseñada para el uso de instituciones oficiales y facilita la inscripción de alumnos nuevos [3], así como SIMAT permite gestionar el proceso matrícula para escuelas y colegios, las universidades también disponen de sistemas que les facilitan este proceso, pero son sistemas aislados y diseñados específicamente para satisfacer las necesidades de cada institución, en función de sus políticas y reglamento institucional, es decir, cada establecimiento tiene potencialmente diferentes sistemas para la gestión del proceso de matrículas, en donde ciertos componentes pueden tener similitudes, pero que realmente desde la construcción son diferentes ya que cada uno cumple requerimientos y necesidades específicas dependiendo el contexto.

Ingeni@, una de las unidades del Departamento de Recursos de Apoyo e Informática, es la encargada de gestionar el proceso de inscripción de estudiantes a diferentes cursos que se imparten en la universidad pertenecientes a proyectos alternos a la programación habitual, con el paso del tiempo y con la asignación de proyectos como Misión-TIC y Ubicua, el proceso de inscripción y matrículas se vuelve tedioso y demandante, es por ello que se identifica la necesidad de un sistema informático que permita gestionar la oferta y matrícula de cursos para los diferentes proyectos que se manejan y que se puedan manejar en el futuro desde Ingeni@.

V. METODOLOGÍA

Se empleó una metodología híbrida y por fases en las cuales se tuvieron unas actividades como entregables los cuales ayudaron al cumplimiento de los objetivos. La primera fase adoptó la metodología Cascada, también llamada modelo en cascada y es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior [4]. Dicha metodología fue apropiada para las actividades de elicitación de requisitos, selección de tecnologías, diseño visual de cómo se debe ver la aplicación (mockups) y diseño o modelamiento del contexto de dominio.

Por otra parte, la fase de desarrollo estuvo gestionada con el marco de trabajo Scrum, el cual es una de las metodologías de desarrollo ágil de Software más reconocidas a nivel mundial, presentando múltiples ventajas, reduciendo riesgos involucrando al cliente en el proceso de desarrollo, usando eventos claramente identificados y con alto grado de inspección, seguimiento, adaptación al cambio y trabajo en equipo [5].

El product backlog que contenía las historias de usuario que satisfacen los requerimientos funcionales se gestionaron desde la plataforma Teams de Microsoft por medio de un board que permitía la planificación, organización y visualización de todo el flujo de trabajo (backlog, to do, doing, testing, production, closed).

Las ceremonias del marco de trabajo Scrum se implementaron de la siguiente manera:

- Sprints: estos ciclos o iteraciones durante el desarrollo tuvieron una duración de 2 semanas (15 días).
- Dailys: estos se llevaron a cabo todos los días laborales con una duración máxima de 15 minutos.
- Planning: se llevaron a cabo los días martes de la primera semana de cada ciclo de los sprints, estimando por puntos de historia y priorizando por las historias de usuario definidas en el backlog.
- Review: Se llevaron a cabo los días viernes de la segunda semana de cada ciclo de los sprints.

Finalmente, el proyecto se abordó en fases y cada fase con diferentes actividades, descritas de la siguiente forma:

1. Refinamiento de requisitos.

Se refinaron los requisitos funcionales y no funcionales del sistema, con el apoyo de todos los involucrados y teniendo en cuenta el modelo de dominio del problema.

2. Creación de mockups iniciales

Con base a los requerimientos identificados y con los posibles cambios hechos al diagrama de dominio, se realizaron los mockups y diseños iniciales de lo que debería ser el aplicativo de cara al usuario final, con el fin de validar y dar mayor entendimiento al problema y a la solución.

3. Selección de tecnologías.

Búsqueda y elección de tecnologías candidatas para la construcción del sistema (tipo de base de datos relacional o no, motor de base de datos, lenguajes de programación para Front-end y Back-end, Frameworks o librerías a utilizar de las disponibles según el lenguaje, etc.)

4. Implementación del sistema.

- a. Configuración inicial de los proyectos (Back-end, Front-end) y de la base de datos. Es acá donde se define la estructura del árbol a seguir, reglas, convenciones, estándares, etc.
- b. Inicio de desarrollo del sistema como tal donde se trabajaron en los sprints las diferentes historias de usuario que se acordaron.
- c. Refactorización y validación constante de la arquitectura y el código (las reglas y convenciones acordadas).
- d. Validación constante de las funcionalidades y solución de bugs encontrados.

VI. RESULTADOS

Al finalizar la práctica académica, se obtuvo un producto web desarrollado en React, con pruebas unitarias y e2e, dicho producto consta de dos módulos principales uno para la administración de proyectos, cohortes, cursos, grupos y usuarios y otro para la gestión de matrícula de un estudiante perteneciente a una cohorte de un proyecto, esto se puede observar en la Figura 1 que representa el diagrama de componentes del aplicativo web y el uso de redux toolkit para manejar el estado de la aplicación y mantener la consistencia de la información entre componentes.

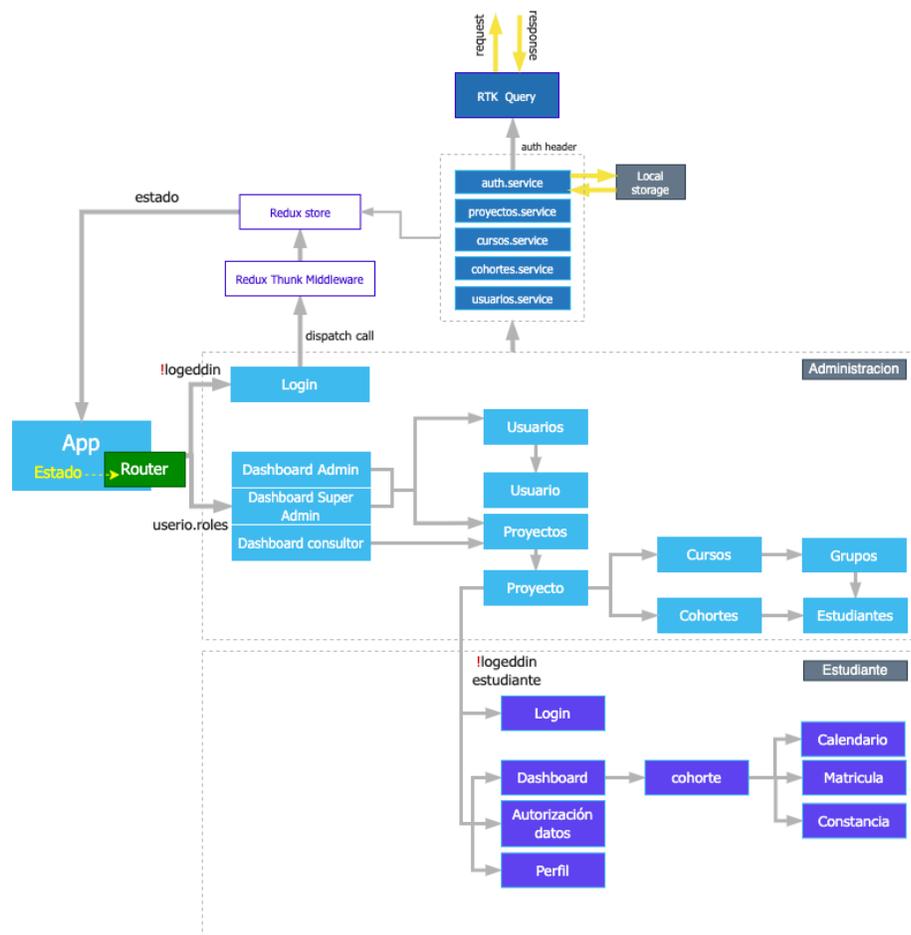


Fig. 1. Diagrama de componentes de la aplicación web

A. Módulo administrativo

El módulo administrativo consta de un login que permite ingresar con correo electrónico o número de identificación y contraseña como se puede apreciar en la figura 2.

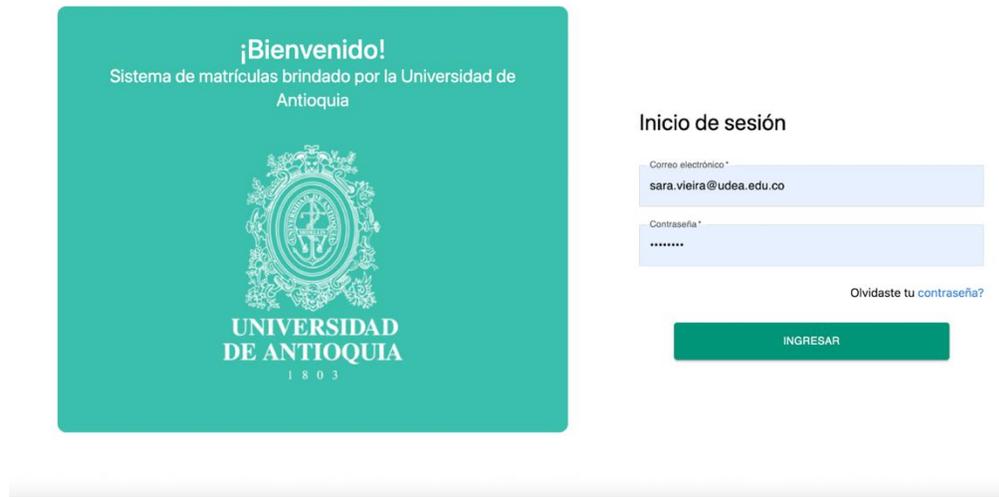


Fig. 2. Pantalla de autenticación de usuarios administradores

Una vez se hace una autenticación válida, lo primero que se muestra es la página de inicio que se puede apreciar en la figura 2, donde se listan los proyectos que se han creado en el sistema.

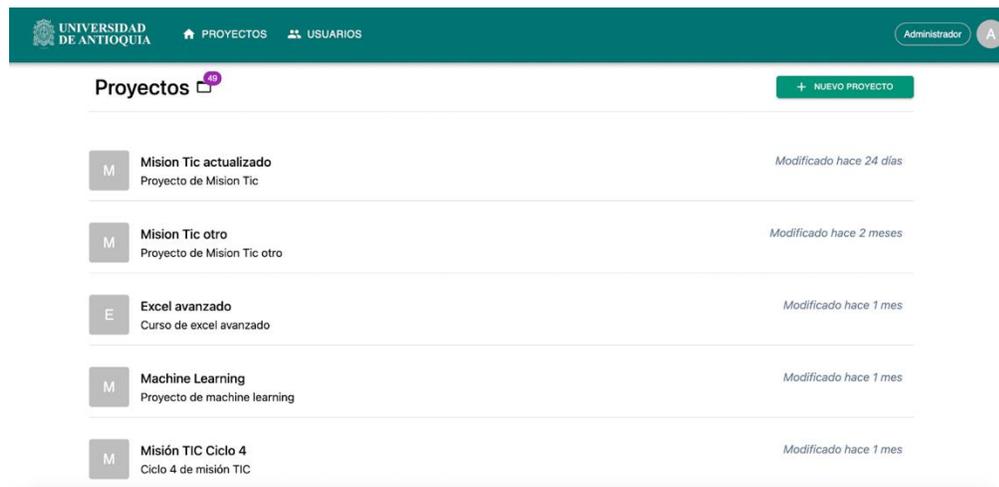


Fig. 3. Pantalla de lista de proyectos

Al elegir un proyecto se navega al detalle de proyecto en donde se podrán gestionar sus cohortes como lo muestra la figura 4, administrar el banco de cursos disponibles para las cohortes como se observa en la figura 5, o lo evidenciado en la figura 6 referente a configurar los parámetros del proyecto, incluyendo imágenes, colores, información necesaria de los estudiantes, usuarios administradores con acceso y políticas de tratamiento de datos.

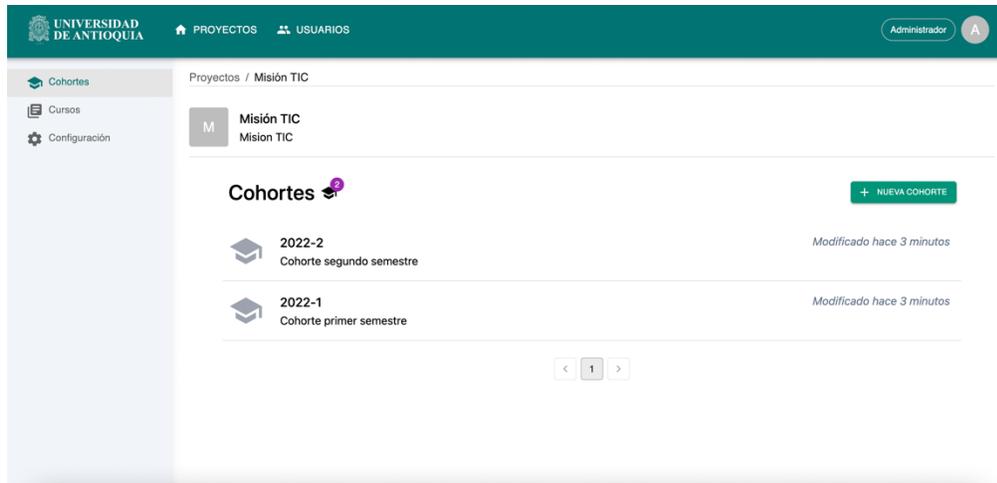


Fig. 4. Pantalla de lista de cohortes de un proyecto

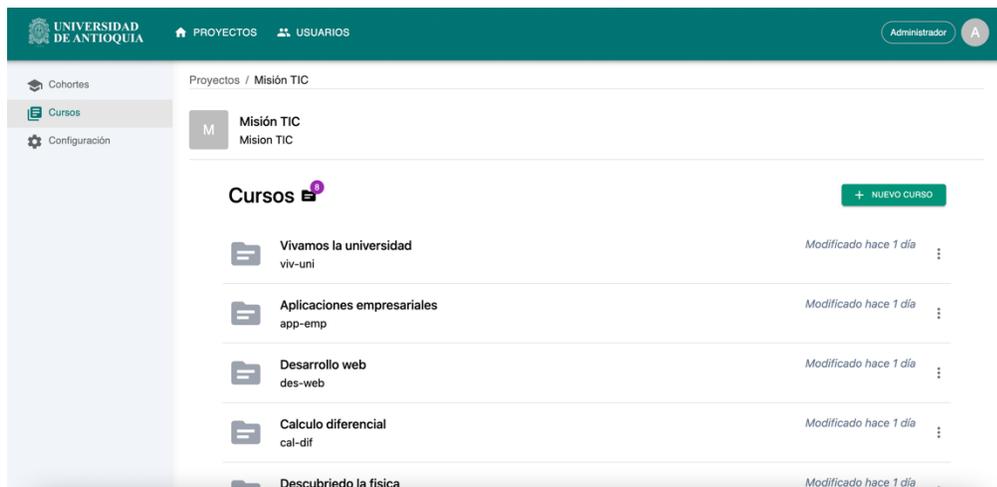


Fig. 5. Pantalla de lista de cursos de un proyecto

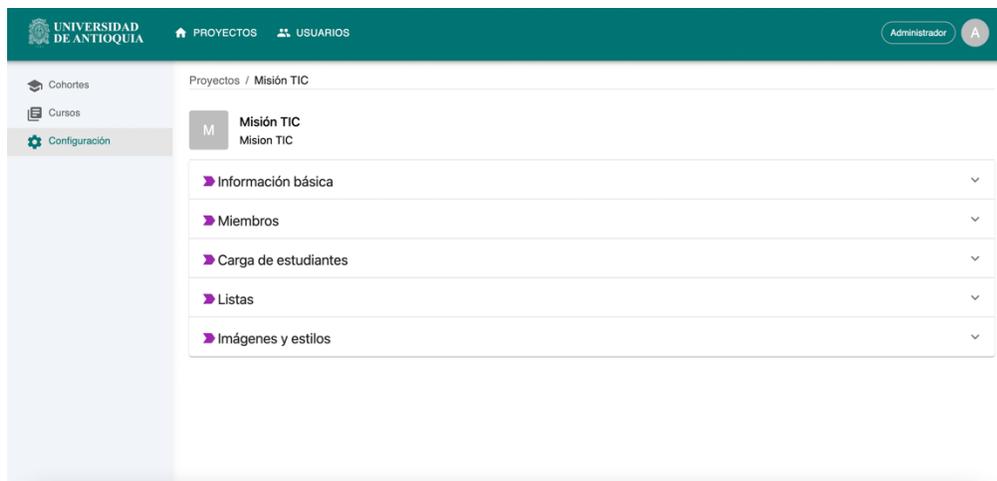


Fig. 6. Pantalla de configuración de parámetros de un proyecto

Algo que es necesario destacar, es la flexibilidad que brinda el sistema a la hora de configurar los proyectos, pues es posible indicar que información se quiere recopilar de los estudiantes mediante la opción ‘Carga de estudiantes’ y crear campos de forma dinámica mediante un formulario que se puede apreciar en la figura 7, donde se indica la propiedad a la cual hará referencia, la etiqueta asociada, el tipo de campo (lista, texto, fecha, numero), habilitación de edición y su obligatoriedad.

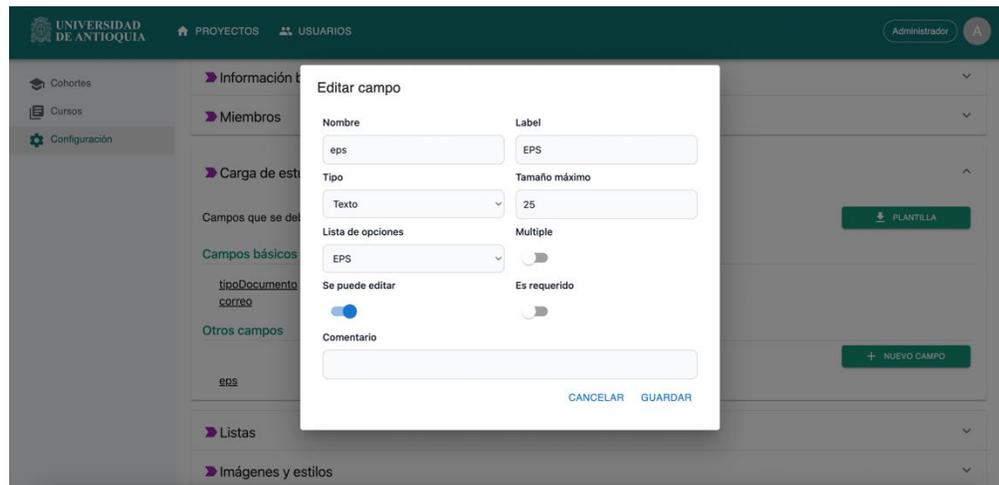


Fig. 7. Formulario de creación/edición de campos de estudiantes de un proyecto

Seguidamente, la figura 8 exhibe cómo es posible configurar la paleta de colores con la que se le mostrará el portal de matrículas a los estudiantes, además de una imagen personalizada para el login, el logo, un banner para dispositivos móviles, otra para dispositivos grandes, la política de tratamiento de datos que manejará el proyecto y un correo electrónico de soporte.

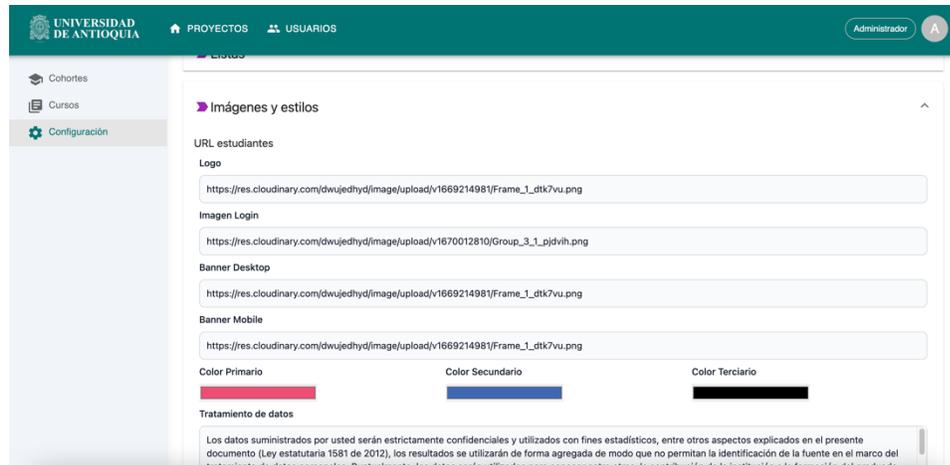


Fig. 8. Pantalla de configuración de estilos, políticas e imágenes de proyecto

Por otra parte, la figura 9 exhibe que cuando se elige una cohorte de la pantalla de lista de cohortes mostrada en la figura 4, se navega a la pantalla de detalle de cohorte en la opción ‘Información’ en donde se puede configurar información básica, cursos que se dictarán de acuerdo a los disponibles en el banco de cursos del proyecto y las reglas de matrícula que se validarán a la hora de que los estudiantes envíen la matrícula.

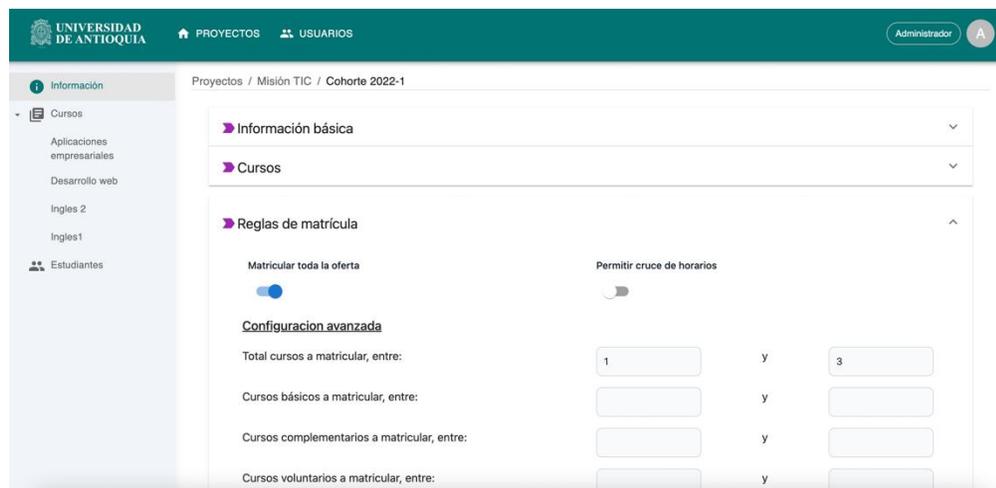


Fig. 9. Pantalla de información y configuración de cohorte

La opción ‘Estudiantes’ en el detalle de cohorte permite gestionar a los estudiantes de la cohorte, precargarlos en el sistema y habilitarlos para la posterior matrícula, en la figura 10, se muestra la pantalla que permite visualizar el total de estudiantes de la cohorte, así como crear o

cargar nuevos estudiantes o ir al detalle de estudiante exhibido en la figura 11, en donde se podrá visualizar su información completa, editarla en caso de que el campo este habilitado para edición o gestionar su matrícula.

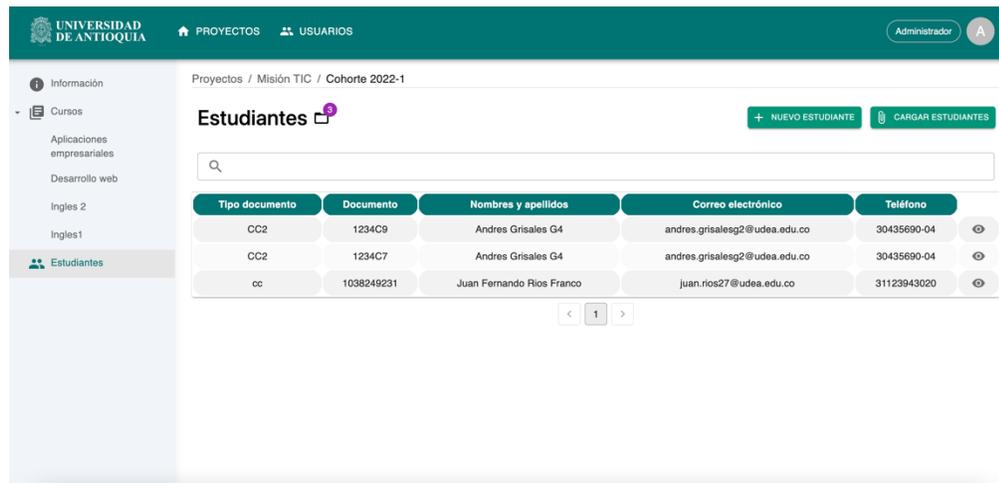


Fig. 10. Pantalla de lista de estudiantes de una cohorte

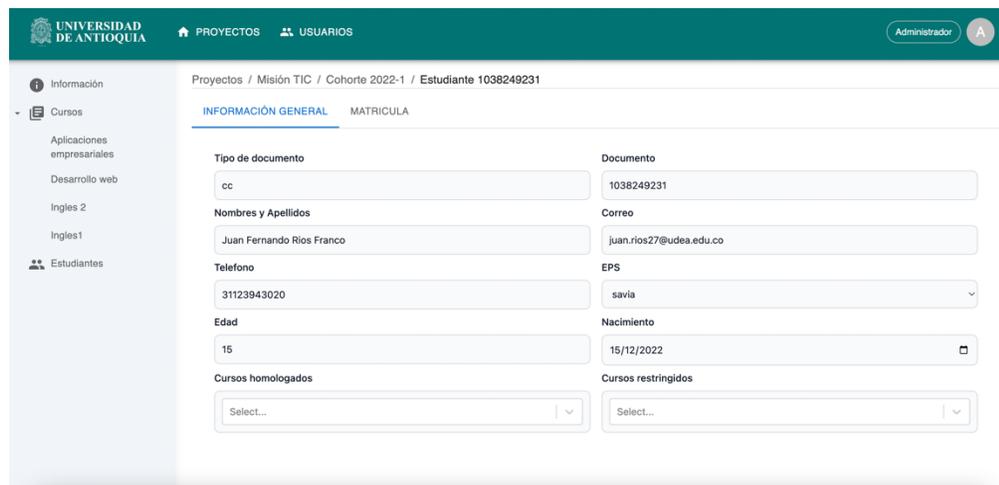


Fig. 11. Pantalla de detalle de estudiante de una cohorte

Adicional a lo anterior, cada uno de los cursos que se imparten en una cohorte aparecen como una opción del menú en donde se puede ir directamente a uno de ellos, visualizar los grupos que hay habilitados y agregar o modificar su información como lo muestra la figura 12.

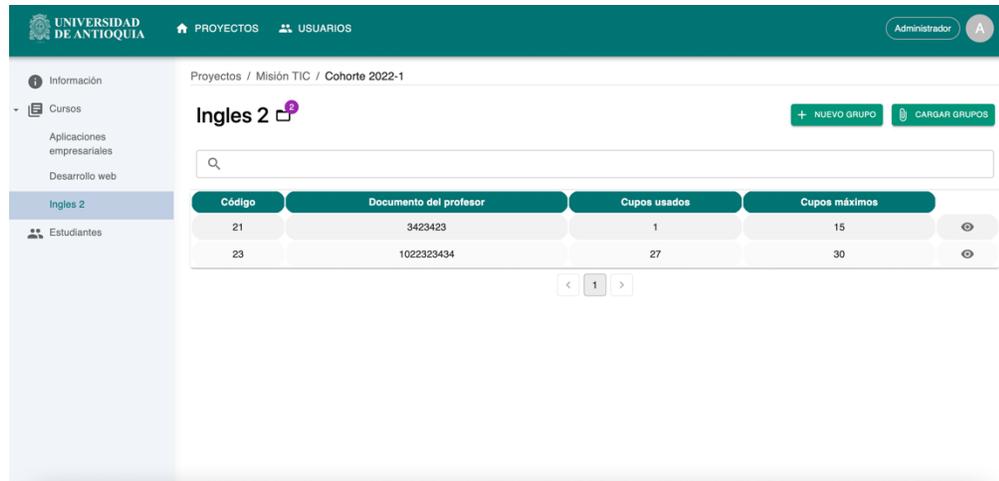


Fig. 12. Pantalla de grupos de un curso en una cohorte

B. Módulo estudiante

Una vez se hacen las configuraciones necesarias para una cohorte por parte del administrador y se cargan los estudiantes, se envía un correo electrónico a estos indicándoles el enlace por el que debe ingresar para la matrícula. Tomando como ejemplo el proyecto Misión-TIC, el login es el que se visualiza en la figura 13, en donde se visualiza una imagen y la paleta de colores que son configuradas por el administrador.



Fig. 13. Ejemplo de inicio de sesión para estudiantes de un proyecto

Una vez un estudiante ingresa sus credenciales, navega hacia la política de privacidad y autorización de tratamiento de datos la cuál es configurada por cada uno de los proyectos que se trabajen en el sistema, en la figura 14, se puede ver un ejemplo de política de privacidad para un proyecto en donde el estudiante debe aceptar para poder continuar.



Fig. 14. Ejemplo de política de privacidad de un proyecto

Luego de la aceptación de términos el sistema le muestra al estudiante las cohortes del proyecto que están activas como se puede observar en la figura 15, de ahí tiene la posibilidad de elegir una y navegar hacia la gestión de matrícula en donde podrá visualizar el calendario académico de la cohorte (figura 16), hacer la matrícula de cursos de acuerdo a la oferta y al calendario académico como lo muestra la figura 17 y finalmente la figura 18 da un ejemplo de la visualización de la constancia de la matrícula realizada.

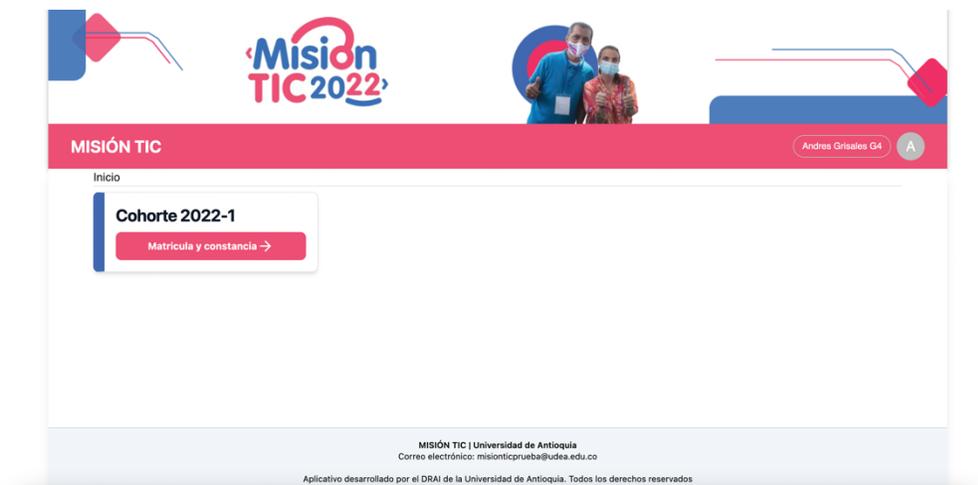


Fig. 15. Ejemplo de cohortes activas para un estudiante de un proyecto



Fig. 16. Ejemplo de calendario académico de una cohorte



Fig. 17. Ejemplo de oferta de cursos para un estudiante



Fig. 18. Ejemplo de constancia de matrícula de un estudiante

VII. CONCLUSIONES

Establecer canales de comunicación constantes en el equipo de desarrollo afecta positivamente la ejecución de un proyecto, permitiendo tener entregables en los tiempos estipulados y que cumplen con los criterios de aceptación establecidos.

La implementación de metodologías ágiles como SCRUM, no solo ha permitido la entrega temprana de funcionalidades, sino también la identificación factores críticos a los que se les debe dar manejo y que no se hubieran identificado de no ser por la interacción constante con el cliente que supone la metodología usada.

El levantamiento de requerimientos, la definición del alcance y la elaboración de diseños iniciales, marcan la diferencia a la hora de realizar entregas tempranas de un producto de software.

La implementación de pruebas automatizadas en un proyecto de desarrollo, garantizan la calidad esperada y dan tranquilidad a todos los involucrados.

La formación académica impartida por el departamento de ingeniería de sistemas, especialmente la referente a los cursos de la línea de ingeniería del software y administración de la información, impactó de manera positiva en el desempeño, la calidad y experticia con la que se abordó el problema, se propuso soluciones y se entregó un producto funcional en un tiempo razonable.

REFERENCIAS

- [1] Universidad de Antioquia. (n.d.). DRAI. Recuperado de Universidad de Antioquia [En línea]. Disponible en: <https://bit.ly/3GxK4iX>
- [2] Bello, Y. C. (2011). Organización, planeación y administración educativa. Perspectivas teóricas en la escuela. *Revista Logos ciencia & tecnología*, 2(2), 88-103.
- [3] Simat—. ::Ministerio de educación nacional de Colombia:.. (n.d.). [En línea]. Disponible en: <https://bit.ly/3k96T4G>
- [4] Sommerville, I. (2005). *Ingeniería del software*. Pearson educación.
- [5] Rodríguez, C., & Dorado, R. (2015). ¿Por qué implementar Scrum? *Revista Untare*, 3(1), 125–144. <https://doi.org/10.21158/23823399.v3.n1.2015.1253>