



Proyecto de Grado

Prototipo de un sistema para el seguimiento a la evaluación por resultados de aprendizaje en la Universidad del Valle.

Bryan Stiven Martinez Bríñez
bryan.brinez@correounivalle.edu.co
201968181-3743

Kevin Steven Victoria Ospina
victoria.kevin@correounivalle.edu.co
201968140-3743

Director:
Mauricio López Benítez
mauricio.lopez@correounivalle.edu.co

Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación
Ingeniería de Sistemas

Universidad del Valle
Sede Tuluá

22 de enero de 2025

Dedicatoria

Dedicamos este proyecto, con amor y gratitud, a nuestras familias, quienes han sido el pilar fundamental en nuestro camino. A nuestros padres, por creer en nosotros, por su apoyo inquebrantable y por los sacrificios realizados para que pudiéramos alcanzar esta meta. Este logro es tanto nuestro como suyo.

A nuestros amigos y seres queridos que, con sus palabras de ánimo y compañía, nos ayudaron a superar los momentos difíciles.

Y, especialmente, dedicamos este esfuerzo a la comunidad educativa de la Universidad del Valle, en la esperanza de que este trabajo contribuya a la mejora continua de los procesos académicos y al fortalecimiento de la educación en nuestra institución.

Agradecimientos

Expresamos nuestro más profundo agradecimiento a nuestras familias, por su apoyo incondicional, paciencia y motivación constante durante esta etapa de nuestra formación.

Agradecemos también a nuestro director de proyecto, el profesor Mauricio López Benítez, por su guía, experiencia y compromiso en cada etapa del desarrollo. Su orientación fue fundamental para la realización de este trabajo. De igual manera, extendemos nuestra gratitud a los profesores, compañeros y coordinadores de la Universidad del Valle que contribuyeron con sus conocimientos y aportes a la consolidación de este prototipo.

Finalmente, valoramos el esfuerzo de todos los involucrados en este proceso, quienes, directa o indirectamente, hicieron posible alcanzar los objetivos propuestos. Este proyecto es el resultado de un trabajo colaborativo y del apoyo constante de una comunidad educativa comprometida con la excelencia.

Resumen

Este proyecto de grado presenta el diseño y desarrollo de un prototipo para la gestión de resultados de aprendizaje en la Universidad del Valle, en respuesta a las exigencias del Decreto 1330 del Ministerio de Educación Nacional. Este decreto promueve un modelo de evaluación educativa centrado en resultados medibles, lo que plantea nuevos retos para las instituciones académicas en términos de registro, seguimiento y análisis de resultados de aprendizaje.

El prototipo desarrollado busca mejorar la trazabilidad y precisión en el manejo de las evaluaciones, proporcionando una herramienta tecnológica que centraliza la gestión de actividades, indicadores y calificaciones asociadas a los resultados de aprendizaje. El sistema incluye funcionalidades como la asignación de resultados a actividades específicas, la generación de informes visuales de cumplimiento, y la automatización de procesos clave para facilitar el trabajo de docentes y coordinadores.

Presentación

Mauricio Lopez Benitez
Docente Facultad de Ingeniería, Universidad del Valle

Kevin Steven Victoria Ospina

Bryan Stiven Martinez Briñez

Estudiantes del programa Ingeniería de Sistemas,
Universidad del Valle

Índice

Capítulo 1, planteamiento del problema y objetivos	9
1.1. Introducción	9
1.2. Planteamiento del problema	9
1.2.1. Descripción del problema	9
1.2.2. Formulación del Problema	10
1.3. Objetivos	10
1.3.1. Objetivo general	10
1.4. Objetivos específicos	10
1.4.1. Resultado obtenido por cada objetivo	12
1.5. Alcance de la propuesta	12
1.6. Justificación	13
Capitulo 2, marco referencial	14
2.1. Marco de antecedentes	14
Moodle	14
Watermark (Taskstream)	14
Blackboard Learn	14
SENA Sofia Plus	14
2.2. Marco teórico	16
Teoría de la Evaluación Educativa:	16
Propósito de la Evaluación Educativa:	17
Medición del Aprendizaje:	17
Modelo de Evaluación como Análisis de Sistemas:	17
Modelo de Evaluación como logro de objetivos:	18
LMS	18
2.3. Desarrollo web	18
2.4. Glosario	19
	19
Capitulo 3, metodologías	20
3.1. Metodología de Investigación	20
3.2. Metodología de desarrollo de software	22
3.2.1. Análisis crítico de la selección de XP frente a otras metodologías . .	23
3.2.2. Limitaciones encontradas durante el desarrollo y cómo se abordaron	24
Capitulo 4, ingeniería del proyecto	25
4.1. Requisitos de software	25
4.1.1. Requisitos funcionales	25
4.1.2. Requisitos no funcionales	26
4.2. Base de datos	27
4.3. Arquitectura	29
4.4. HTTP	31
4.5. Herramientas utilizadas	32
4.5.1. Next.js	32
4.5.2. Firebase	32

4.5.3.	Nodemailer	33
4.5.4.	NextAuth	33
4.5.5.	Autenticación de Firebase	34
4.5.6.	Zod	34
4.5.7.	React	35
4.5.8.	Axios	35
4.5.9.	React Icons	35
4.5.10.	Tailwind CSS	36
4.5.11.	ESLint	36
4.5.12.	Scheduler	36
4.6.	Control de versiones	37
4.7.	Mockups	38
4.8.	Reportes	40
4.8.1.	Reporte por grupo	40
4.8.2.	Reporte por estudiante	40
4.9.	Prototipo	41
4.10.	Resultado de Pruebas	49
4.10.1.	Pruebas Funcionales	49
4.10.2.	Pruebas no funcionales	53
5.	Conclusiones	55
5.1.	Reflexión sobre el impacto y limitaciones del prototipo	56
5.2.	Futuras Líneas de Trabajo	57
6.	Anexos	61

Índice de cuadros

1.	Resultados obtenidos de cada objetivo específico.	12
2.	Comparativa entre Moodle, Watermark (Taskstream) y Blackboard Learn .	15
3.	Tabla Usuario 1	27
4.	Tabla Curso	27
5.	Tabla Resultado de aprendizaje	27
6.	Tabla Indicador de logro	27
7.	Tabla Programa	28
8.	Tabla Grupo	28
9.	Pruebas para la tabla Cursos	49
10.	Pruebas para la tabla Facultad	49
11.	Pruebas para la tabla Grupo	49
12.	Pruebas para la tabla Indicador Logro	50
13.	Pruebas para la tabla Nota	50
14.	Pruebas para la tabla Programa	51
15.	Pruebas para la tabla Resultado Aprendizaje	51
16.	Pruebas para la tabla Sede	52
17.	Pruebas para la tabla Usuario	52
18.	Prueba de carga de página (Rendimiento)	53
19.	Pruebas de tiempo de respuesta (Rendimiento)	53
20.	Prueba de disponibilidad de funcionalidades en el menú (Usabilidad) . . .	53
21.	Prueba de respuesta a errores de base de datos (Fiabilidad)	54
22.	Prueba de escalabilidad (Rendimiento)	54
23.	Pruebas de tasa de intentos de acceso no autorizado bloqueados (Seguridad)	54

Índice de figuras

1.	Proceso XP.	23
2.	Modelo entidad relación con tablas mas detalladas	28
3.	Diagrama Arquitectura utilizada	30
4.	Proceso Github ramas	37
5.	Mockups Inicio de Sesión	38
6.	Mockups Gestión Usuarios	39
7.	Mockups Gestión Resultados AP	39
8.	Modulo gestión usuario	41
9.	Creación de usuario	42
10.	Usuario con sus cursos asignados	42
11.	Usuario con sus certificados de estudio	43
12.	Modulo gestión facultades	43
13.	Modulo gestión de programas	44
14.	Modulo gestión cursos	44
15.	Cursos y sus grupos	45
16.	Gestión de un curso, apartado notas	45
17.	Gestión de un curso, apartado evidencias	46
18.	Gestión de un curso, apartado informes	46
19.	Gestión de un curso, apartado informes	47
20.	Gestión de los resultados de aprendizaje e indicadores de logro	47
21.	Creación de un indicador de logro	48

Capítulo 1, planteamiento del problema y objetivos

1.1. Introducción

En un entorno educativo en constante evolución, marcado por la innovación tecnológica y los cambios normativos, las instituciones de educación superior enfrentan desafíos significativos para adaptar sus métodos de enseñanza y evaluación. La Universidad del Valle no es ajena a esta realidad, especialmente tras la implementación del Decreto 1330 del Ministerio de Educación, que establece lineamientos para la evaluación por resultados de aprendizaje.

Este nuevo enfoque de evaluación busca no solo medir el conocimiento adquirido por los estudiantes, sino también evaluar su desempeño en relación con los objetivos de aprendizaje establecidos. Sin embargo, este cambio ha traído consigo desafíos operativos y metodológicos significativos, principalmente debido a la falta de un sistema que permita gestionar este tipo de evaluación.

Frente a esta situación, surgió la necesidad de desarrollar un sistema que facilite la gestión del registro y seguimiento de la evaluación por resultados de aprendizaje, mejorando así la calidad de la educación y la experiencia tanto de estudiantes como de profesores. Este trabajo de grado propone el diseño y desarrollo de un prototipo para dicho sistema, enfocándose en la usabilidad en el manejo de los procesos evaluativos.

El proyecto se alineó con los objetivos estratégicos de la Universidad del Valle y respondió a una necesidad imperante de adaptación y mejora continua en el ámbito educativo. A través de este prototipo, se busca proporcionar una herramienta práctica que permita a la institución cumplir con los requerimientos del Decreto 1330, al tiempo que se mejora la calidad educativa y se adaptó a las nuevas demandas del entorno educativo moderno.

1.2. Planteamiento del problema

1.2.1. Descripción del problema

El Decreto 1330 del Ministerio de Educación estableció los lineamientos para el proceso de evaluación por Resultado de aprendizaje en las instituciones de educación superior, este decreto introdujo cambios significativos en la forma de evaluar el aprendizaje de los estudiantes. [1]

Anteriormente, la evaluación se centraba principalmente en medir el conocimiento adquirido por los estudiantes a través de pruebas y exámenes, sin embargo, con la implementación de este decreto, se buscó evaluar no solo el conocimiento, sino también el desempeño de los estudiantes en relación con los Indicador de logro asociados a los resultados de aprendizaje.

Los cambios incorporados en el citado decreto tuvieron como objetivo principal mejorar la calidad de la educación y asegurar que los estudiantes adquirieran las competencias necesarias para su desarrollo académico y profesional, además, se buscó fomentar un enfoque más centrado en el estudiante, que promoviera la reflexión, la aplicación práctica del conocimiento y el desarrollo de habilidades.

De acuerdo con lo anterior, la Universidad del Valle se enfrenta a un desafío significativo en la implementación de este nuevo enfoque de evaluación; la falta de un sistema que permita gestionar de manera eficiente esta forma de evaluación ha generado complicaciones notables en el proceso de evaluar y hacer seguimiento a los resultados de aprendizaje de cada estudiante, es por esto que nació la necesidad de indagar a los coordinadores de programa, docentes y directivas académicas para establecer el panorama de como se esta llevando esta gestión, la falta de seguimiento se traduce en un aumento en la tasa de reprobación, insatisfacción de los estudiantes y, en última instancia, una disminución en la calidad de la educación ofrecida.

Por tanto, el propósito de este proyecto de grado es abordar estas dificultades y proponer soluciones efectivas que permitan a la Universidad del Valle adaptarse de manera efectiva al nuevo enfoque de evaluación por resultados de aprendizaje. Mejorar la evaluación de los resultados de aprendizaje no solo beneficiará a los estudiantes, al asegurar su desarrollo académico y profesional, sino que también fortalecerá la capacidad de la universidad para cumplir con los requisitos del decreto y mantener su prestigio en el ámbito educativo.

1.2.2. Formulación del Problema

¿Cómo se implementó un sistema practico para el registro y seguimiento de calificaciones por resultados de aprendizaje en la Universidad del Valle, conforme a los lineamientos del Decreto 1330 del Ministerio de Educación?

Sistematización del problema

- ¿Cuáles fueron las necesidades específicas que se abordaron en el diseño de un sistema que permita el registro, control y seguimiento de la evaluación por resultados de aprendizaje en la Universidad del Valle?
- ¿Cuáles fueron los aspectos clave que se consideraron al desarrollar el sistema de gestión de evaluación por resultados de aprendizaje?
- ¿Existieron desafíos o limitaciones potenciales que podrían afectar la viabilidad de la implementación exitosa de este sistema en la universidad?

1.3. Objetivos

1.3.1. Objetivo general

Desarrollar el prototipo de un sistema de gestión integral que permita la gestión de la evaluación por resultados de aprendizaje de los estudiantes en Universidad del Valle.

1.4. Objetivos específicos

- Analizar los requisitos para el desarrollo de una plataforma web, enfocándose en la elaboración de un sistema accesible y los procesos de retroalimentación para profesores y estudiantes en la Universidad del Valle.

- Diseñar módulos para la plataforma, enfocándose en el desarrollo de una interfaz de usuario que siga principios de usabilidad establecidos y en la implementación de mecanismos de retroalimentación automatizada basados en criterios de rendimiento medibles.
- Implementar la plataforma con los módulos establecidos, asegurando una integración efectiva de las funcionalidades para el registro y gestión de evaluaciones.
- Evaluar la funcionalidad y la usabilidad de la plataforma desarrollada para el registro y seguimiento de calificaciones por resultados de aprendizaje en la Universidad del Valle, a través de pruebas piloto y análisis de su diseño en relación con los lineamientos del Decreto 1330 del Ministerio de Educación.

1.4.1. Resultado obtenido por cada objetivo

N°	Objetivo específico	Resultado obtenido
1	Analizar los requisitos para el desarrollo de una plataforma web, enfocándose en la elaboración de un sistema accesible y los procesos de retroalimentación para profesores y estudiantes en la Universidad del Valle.	Documento de requisitos completado que detalla especificaciones funcionales y de diseño para un sistema accesible y procesos de retroalimentación, basado en el análisis de necesidades de profesores y estudiantes.
2	Diseñar módulos para la plataforma, enfocándose en el desarrollo de una interfaz de usuario que siga principios de usabilidad establecidos y en la implementación de mecanismos de retroalimentación automatizada basados en criterios de rendimiento medibles.	Conjunto de módulos funcionales con interfaz de usuario siguiendo principios de usabilidad y mecanismos de retroalimentación, acompañados de documentación de diseño y pruebas de concepto.
3	Implementar la plataforma con los módulos establecidos, asegurando una integración efectiva de las funcionalidades para el registro y gestión de evaluaciones.	Se desarrolló un prototipo de la plataforma, donde los profesores, puedan probarla en situaciones reales, registrando y gestionando evaluaciones sin problemas.
4	Evaluar la funcionalidad y la usabilidad de la plataforma desarrollada para el registro y seguimiento de calificaciones por resultados de aprendizaje en la Universidad del Valle.	Se realizó informe de evaluación que incluyó los resultados de las pruebas de usabilidad y funcionalidad; este resalta aspectos como la facilidad de uso, la eficiencia del flujo de trabajo, la precisión en el registro y seguimiento de calificaciones

Cuadro 1: Resultados obtenidos de cada objetivo específico.

1.5. Alcance de la propuesta

La propuesta que se presentó aborda los desafíos enfrentados por la Universidad del Valle en relación con el registro y seguimiento de calificaciones de evaluaciones basadas en resultados de aprendizaje. Este proyecto se centró en el diseño y la implementación de una solución que facilite un registro de datos más ágil, contribuyendo de este modo a la mejora de la calidad educativa. Un componente esencial de este proyecto es el desarrollo de una interfaz de usuario diseñada para facilitar la interacción

El proyecto se enfocó en mejorar el proceso de registro de evaluaciones, facilitando la carga de evidencias y la generación de informes necesarios para tomar decisiones basadas en datos en la Universidad del Valle. Se diseñará una solución que proporcione a los educadores y administradores acceso a información relevante, permitiéndoles tomar decisiones informadas. Además, se realizará una evaluación de la efectividad de la plataforma para asegurar una mejora continua en los procesos educativos y en la formulación de cursos.

1.6. Justificación

La adaptación de la Universidad del Valle al Decreto 1330 del Ministerio de Educación llevó a la actualización de su método de evaluación, sin embargo, la institución enfrenta la limitación de no contar con un sistema especializado para la gestión de esta nueva metodología. Actualmente, se recurre a herramientas no diseñadas específicamente para la evaluación por resultados de aprendizaje, evidenciando la necesidad de desarrollar un sistema adecuado que facilite un proceso organizado y sistemático para el registro y manejo de calificaciones. Este contexto subraya la importancia de crear un prototipo de sistema que no solo responda a las exigencias normativas, sino que también mejore la operatividad y precisión en la evaluación académica

Capítulo 2, marco referencial

2.1. Marco de antecedentes

Moodle Moodle es un sistema de aprendizaje que ofrece a docentes, gestores y alumnos una solución integrada, firme y segura, destinada a la creación de entornos educativos personalizables. [2]

Watermark (Taskstream) Watermark es el mayor proveedor de sistemas de inteligencia educativa para instituciones de educación superior en todo el mundo. Watermark pone mejores datos en manos de administradores, educadores y estudiantes de todo el mundo para capacitarlos para conectar información y obtener conocimientos sobre el aprendizaje que impulsarán mejoras significativas. A través de su innovadora plataforma de inteligencia educativa, Watermark apoya a las instituciones en el desarrollo de un enfoque intencional para el aprendizaje y el desarrollo basado en datos en los que pueden confiar. [3]

Blackboard Learn A medida que cambian los estilos de vida de los estudiantes, necesita un LMS inclusivo, flexible y revelador para brindar experiencias educativas atractivas y satisfacer las necesidades de los estudiantes de todos los orígenes. Blackboard® Learn, enfocado estratégicamente en crear eficiencia para los instructores, tiene la tasa de desarrollo de productos más rápida disponible en el mercado y es verdaderamente el LMS para la enseñanza y el aprendizaje modernos. [4]

SENA Sofia Plus SENA Sofia Plus es un sistema educativo integral que brinda acceso a formación profesional en Colombia, centrado en el desarrollo de competencias laborales. A través de este sistema, los estudiantes pueden acceder a una amplia gama de recursos didácticos y herramientas de aprendizaje, permitiendo un seguimiento efectivo de su progreso educativo. Su enfoque práctico y aplicado está diseñado para preparar a los estudiantes para enfrentar los desafíos del mercado laboral actual, adaptando su oferta formativa a las necesidades cambiantes de la industria y promoviendo la empleabilidad de los egresados. [5]

Características	Moodle	Watermark (Taskstream)	Blackboard Learn
Objetivo principal	Plataforma de aprendizaje	Herramienta de evaluación y ePortfolio	Plataforma de aprendizaje
Registro de evaluaciones	✓	✓	✓
Seguimiento de evaluaciones	✓	✓	✓
Registro de Resultados	✓	✓	✓
Evaluación por Resultados	✓ (con Plugins específicos)	✓	✓
Analítica del Aprendizaje	✓ (con Plugins específicos)	✓	✓
Idiomas disponibles	Multilingüe (100+ idiomas)	Principalmente inglés	Multilingüe
Dispositivos disponibles	Web, móvil	Web	Web, móvil
Costo	Libre (Código abierto) + costos de hosting y personalización	Varía según las necesidades institucionales	Varía según el paquete y la institución
Soporte y Formación	Comunidad + empresas de soporte	Soporte directo de la empresa	Soporte directo de la empresa
Actualizaciones y Mejoras	Regularmente (comunidad activa)	Regularmente	Regularmente
Uso en instituciones educativas	Ampliamente adoptado	Adoptado principalmente para evaluación y acreditación	Ampliamente adoptado

Cuadro 2: Comparativa entre Moodle, Watermark (Taskstream) y Blackboard Learn

Comparación crítica

El prototipo desarrollado para la gestión de la evaluación por resultados de aprendizaje se diferencia significativamente de herramientas existentes como Moodle, Blackboard y Watermark (Taskstream) en varios aspectos clave:

- **Enfoque específico:** Moodle y Blackboard son sistemas amplios que abarcan una variedad de funciones, como la gestión de cursos, foros y contenidos. Por su parte, el prototipo está diseñado exclusivamente para cumplir con los lineamientos del Decreto 1330 del Ministerio de Educación Nacional, lo que lo hace más enfocado en las necesidades específicas de la Universidad del Valle.
- **Costo y personalización:** Moodle es de código abierto, pero requiere costos adicionales de personalización, hosting y mantenimiento. Blackboard y Watermark son soluciones comerciales con modelos de pago variables. En cambio, el prototipo desarrollado ofrece una solución interna que aprovecha la infraestructura existente de la universidad, reduciendo los costos de implementación y operación a largo plazo.
- **Analítica de aprendizaje:** Moodle y Blackboard integran analíticas avanzadas mediante plugins o soluciones externas. Sin embargo, el prototipo proporciona informes personalizados diseñados específicamente para los indicadores de logro establecidos por la universidad, facilitando decisiones basadas en datos.
- **Flexibilidad:** Mientras que Moodle y Blackboard ofrecen personalizaciones limitadas o costosas, el prototipo permite adaptar módulos específicos, como la asignación de indicadores de logro, para alinearse completamente con las necesidades de la institución.
- **Soporte técnico:** Blackboard y Watermark dependen de soporte externo, mientras que el prototipo facilita el control y la actualización directa por parte del equipo interno de desarrollo, garantizando respuestas rápidas a problemas o nuevas necesidades.

2.2. Marco teórico

Teoría de la Evaluación Educativa: La evaluación educativa es una parte importante de la medición y mejora de la calidad de la educación. Este enfoque se centra no sólo en el aprendizaje de los estudiantes sino también en el funcionamiento de todo el sistema educativo, incluida la eficacia de las instituciones escolares y el desempeño de los docentes. La evaluación se ha convertido en una disciplina que requiere estrategias diversas y reflexivas para captar la complejidad de la educación y sus resultados. Durante muchos años se ha enfatizado la necesidad de una evaluación integral. . [6]

La evolución de la evaluación educativa ha estado marcada por importantes contribuciones, como la de Ralph W. Tyler, quien enfatizó la importancia de evaluar si se están logrando los objetivos educativos. Aunque su modelo ha sido criticado por ser limitante, proporciona una base para analizar y mejorar la práctica de la evaluación. [7]

Propósito de la Evaluación Educativa: El propósito de la evaluación educativa abarca una variedad de objetivos importantes más allá de simplemente calificar a los estudiantes. Su propósito incluye no sólo mejorar el aprendizaje y proporcionar retroalimentación valiosa a estudiantes y profesores, sino también influir en las decisiones estratégicas en la planificación de cursos. Además, la evaluación educativa es fundamental para la rendición de cuentas del sector educativo. Facilitar el seguimiento del progreso y la eficacia de los métodos de enseñanza y ayudar a identificar áreas para el desarrollo y mejora continua de los programas educativos.

Medición del Aprendizaje: Las evaluaciones están diseñadas para medir objetivamente el aprendizaje de los estudiantes y utilizan una variedad de programas, rúbricas y evaluaciones de varios tipos. Estas herramientas son esenciales para la evaluación integral del conocimiento, la práctica y el seguimiento de los estudiantes. Además, permiten la autoevaluación para adaptarse a diferentes estilos de aprendizaje. Al proporcionar mediciones más precisas y representativas del desempeño y conocimiento real de los estudiantes, promovemos un mejor aprendizaje.

Desde este enfoque, la evaluación se entiende como un proceso sumativo enfocado en productos finales que generan resultados mensurables y cuantificables. Los criterios para evaluar se basan en objetivos previamente definidos, conocidos como específicos instruccionales, entre otros, los cuales se expresan en términos de comportamientos observables, medibles y cuantificables. [8]

Desde esta perspectiva, al centrarse en mediciones estandarizadas, las especificidades y diferencias únicas de cada individuo tienden a pasarse por alto porque todos son evaluados utilizando los mismos criterios o criterios. Esto puede resultar en evaluaciones que no reflejen completamente las habilidades o antecedentes de cada estudiante. Además, puede limitar la capacidad de identificar y valorar habilidades únicas que no se ajustan a los parámetros estándar, reduciendo así las oportunidades de desarrollar un aprendizaje diverso y personalizado que abarque el potencial de todos los estudiantes.

Modelo de Evaluación como Análisis de Sistemas: Este enfoque pone énfasis en los resultados logrados tras el paso de ciertos insumos a través de procesos interconectados. Por lo tanto, la evaluación se centra en la información cuantitativa recogida al término de diversas etapas o fases; evaluar desde esta perspectiva implica explorar las relaciones de causa y efecto a través de variables cuantificables. [9]

Este enfoque de evaluación basado en resultados establece un vínculo directo entre los estándares académicos y las deficiencias identificadas a través de indicadores y resultados de pruebas. Se caracteriza por el uso de relaciones de rendición de cuentas, un esfuerzo por lograr objetividad y coherencia en el proceso de evaluación y un énfasis en la eficacia y rentabilidad de las estrategias educativas. La pregunta clave con este enfoque es si se logra el efecto deseado y si existen formas efectivas y rentables de lograrlo. Este enfoque se basa en una era dinámica y narrativa en la que las evaluaciones no sólo miden el desempeño sino que también cuentan historias sobre éxitos y desafíos académicos.

Modelo de Evaluación como logro de objetivos: Este modelo ha sido influyente en la mayoría de los prototipos desarrollados posteriormente, siendo Tyler su principal impulsor. Se basa en la definición previa de objetivos específicos a alcanzar. Por ello, el objetivo principal de la evaluación es determinar hasta qué punto se logran estos objetivos. Pone más énfasis en los resultados finales que en los procesos en sí, y no toma en cuenta directamente las acciones realizadas durante el proceso ni la influencia de otros factores que puedan afectarlo. Se centra principalmente en verificar, comprobar o medir los objetivos establecidos. [9]

El modelo de evaluación educativa desarrollado por Tayler tuvo una profunda influencia en muchos de los prototipos desarrollados posteriormente. El modelo se basa en definir objetivos de aprendizaje de antemano y, esencialmente, determinar en qué medida se han logrado estos objetivos. Se centra más en el resultado final del proceso de aprendizaje que en el proceso en sí, sin considerar directamente los resultados de las actividades realizadas durante el proceso u otros factores externos. Su enfoque principal es validar, verificar o determinar el logro de los objetivos establecidos.

LMS Un Sistema de Gestión de Aprendizaje (LMS, por sus siglas en inglés) es una herramienta de software desarrollada específicamente para gestionar, documentar, monitorear, reportar y automatizar cursos educativos, programas de capacitación y materiales de aprendizaje y desarrollo. Originados del e-Learning, los LMS representan el segmento más amplio en el mercado de sistemas educativos. Fueron introducidos por primera vez a finales de la década de 1990 y han experimentado un crecimiento masivo en su uso, especialmente impulsado por la necesidad de aprendizaje remoto durante la pandemia de COVID-19. Además de centrarse en la entrega de aprendizaje en línea, los LMS están diseñados para identificar brechas en la formación y el aprendizaje, utilizando datos analíticos e informes. Pueden actuar como plataformas para contenido en línea, incluyendo cursos basados en modalidades asíncronas y síncronas, y en el ámbito de la educación superior, pueden ofrecer gestión de aulas para capacitación dirigida por instructores o para modelos de aulas invertidas. Los LMS modernos incluyen algoritmos inteligentes para hacer recomendaciones automatizadas de cursos basadas en el perfil de habilidades del usuario, así como para extraer metadatos de materiales de aprendizaje y hacer estas recomendaciones aún más precisas. [10]

2.3. Desarrollo web

En el desarrollo web es muy importante hablar sobre dos términos esenciales para el buen funcionamiento de un sitio web, los cuales son el Backend y el Frontend. Estos no solo garantizan el correcto funcionamiento sino que también brindan seguridad y accesibilidad, usabilidad y optimización. Para definirlos de una mejor manera, a continuación:

- **Backend**

El desarrollo frontend, también conocido como "desarrollo del lado del cliente", implica la creación de HTML, CSS y JavaScript. Estas tres tecnologías son responsables de dar estructura y diseño a la interfaz de un sitio web o aplicación, incluyendo aspectos como fondos, colores, texto, animaciones y otros efectos visuales. [11]

- **Frotend**

El backend gestiona y procesa la información que nutre al frontend. Está formado por frameworks, bases de datos y códigos. Para el funcionamiento eficiente de un sitio web o aplicación, se necesita una gran cantidad de datos e información que se guardan en el sistema informático, pero en una sección no visible para el usuario. A diferencia del frontend, esta parte no es accesible ni visible para el usuario final. [11]

2.4. Glosario

Código abierto El código abierto se refiere a un modelo de producción descentralizado que habilita a cualquiera a modificar y distribuir tecnología, dado que su diseño está disponible públicamente. [12]. 15

ePortfolio Un eportfolio, o portafolio electrónico, es un portafolio creado utilizando herramientas tecnológicas y generalmente es también digital, lo que significa que los documentos incluidos en él son accesibles mediante un ordenador [13]. 15

Indicador de logro Los indicadores de logro son medidas cuantitativas o cualitativas utilizadas para evaluar el grado de éxito en el cumplimiento de los resultados de aprendizaje o metas educativas. Estos indicadores suelen ser específicos, medibles y relevantes para los objetivos educativos establecidos, y sirven como una herramienta para monitorear y evaluar el progreso y la eficacia de los procesos de enseñanza y aprendizaje.. 9

Multilingüe Que habla varias lenguas [14]. 15

Plugins Los plugins son complementos que añaden funcionalidades extra o mejoras a los programas. Es decir, son miniprogramas que suman alguna característica que no venía por defecto en el programa original. Funcionan como añadidos pero no por sí mismos [15]. 15

Producto mínimo viable Es un producto (de software para fines de este artículo) que cuenta con las características, requerimientos y requisitos mínimos para que un software pueda ser lanzado al mercado y poder percibir una remuneración económica igual o superior a su inversión inicial. [13]. 20

Resultado de aprendizaje Un resultado de aprendizaje se refiere a los conocimientos, habilidades y competencias adquiridas y demostradas por un estudiante al finalizar un proceso de aprendizaje. Estos resultados pueden ser medidos y evaluados contra estándares predefinidos, proporcionando una indicación clara del progreso y logro educativo del estudiante.. 9

Capítulo 3, metodologías

3.1. Metodología de Investigación

La investigación tecnológica es un proceso sistemático que implica la aplicación de conocimientos científicos, habilidades técnicas y recursos materiales para desarrollar soluciones innovadoras a problemas prácticos utilizando nuevas tecnologías. Incluye especialistas de campos como la ingeniería, la informática y la biotecnología y tiene un impacto significativo en sectores como la industria, la educación y la medicina. El objetivo es mejorar los procesos existentes o crear nuevas soluciones para problemas complejos. [16] Algunas de las formas en las que la investigación tecnológica ayuda al progreso:

- **Mejora del rendimiento:** Descubre maneras más eficientes de hacer las cosas, aumentando la productividad y el crecimiento económico. [16]
- **Mayor conocimiento sobre el mundo:** Nos ayuda a entender mejor el funcionamiento de nuestra economía y sociedad. [16]
- **Desarrollo de nuevas soluciones:** Encuentra soluciones creativas a problemas complejos, abriendo nuevas oportunidades en áreas comerciales y sociales. [16]
- **Mejora del bienestar:** Desarrolla soluciones para mejorar nuestro bienestar general, como inventos útiles o software de entretenimiento. [16]

Ahora, para enfocar la investigación tecnológica en la ingeniería de software, particularmente en el desarrollo web que es en lo que se va desarrollar el prototipo, se centra en varios aspectos importantes para mejorar la calidad y eficiencia de las aplicaciones web. Algunos de los enfoques y metodologías que en este campo incluyen:

- **Metodología de desarrollo:** Debido a la naturaleza única de las aplicaciones web, incluidos los equipos pequeños, los recursos y las limitaciones de tiempo, la adopción de una metodología de desarrollo, especialmente una metodología ágil, es muy importante en el contexto del desarrollo web. Esto incluye la definición de una metodología adecuada para los productos mínimos viables Producto mínimo viable. [16]
- **Levantamiento de Requerimientos:** definir los requisitos mediante propuestas de valor e historias de usuario es un elemento clave del desarrollo web, ya que permite presentar los hechos de forma sucinta y precisa. [16]
- **Análisis y diseño:** se recomiendan la modelización y los diagramas de secuencia para mejorar la comprensión del sistema por parte del usuario y facilitar la retroalimentación. Es importante la elección de tecnologías y marcos de trabajo front-end y back-end, preferiblemente de código abierto. [16]
- **Mejores prácticas y estándares de codificación:** Deben establecerse mejores prácticas y estándares como la notación PSR, la metodología DRY (Don't Repeat Yourself) y los estándares MVC (Model View Controller) para garantizar que los componentes desarrollados sean eficaces, eficientes y mantenibles. [16]

Por lo tanto, la metodología de investigación aplicada en este proyecto incluiría los siguientes pasos:

- **Investigación de Tecnologías Existentes:** Se examinan las plataformas y herramientas actuales (como Moodle, Watermark y Blackboard Learn) para entender sus capacidades y limitaciones en la gestión de evaluaciones.
- **Identificación de Necesidades y Requisitos:** Se realiza un análisis detallado de las necesidades específicas de la Universidad del Valle en términos de gestión de evaluaciones por resultados de aprendizaje.
- **Desarrollo del Prototipo:** Se diseña y desarrolla un prototipo que aborde las necesidades identificadas, integrando tecnologías adecuadas y enfoques innovadores.
- **Pruebas y Retroalimentación:** El prototipo es probado para asegurar que cumple con los requisitos y necesidades identificados. La retroalimentación de los usuarios finales (profesores y estudiantes) es crucial para refinar el sistema.

3.2. Metodología de desarrollo de software

Al investigar sobre distintas metodologías de desarrollo de software, se muestra que cada una tienen sus fortalezas según el proyecto que se va a realizar. Tras un análisis en conjunto, se ha elegido Extreme Programming (XP) como la metodología que mas se adecua a las necesidades del proyecto de desarrollo que se va a realizar. XP tiene algunas razones claves para ser elegida.

En primer lugar, XP facilita la adaptación al cambio. Esto es muy importante en entornos dinámicos donde los requisitos pueden cambiar con el tiempo. Esta metodología valora los comentarios de los clientes y los integra continuamente, permitiendo ajustes periódicos para garantizar que el producto final satisfaga las necesidades reales de los usuarios.

Otra razón de peso para elegir XP es su enfoque en la calidad y la eficiencia en el proceso de desarrollo. A través de prácticas como la programación en pares y la integración continua, XP promueve código de alta calidad y minimiza el riesgo de errores. Las pruebas frecuentes también pueden ayudarle a identificar y corregir errores tempranamente, evitando sorpresas desagradables más adelante en el proyecto.

Extreme Programming, se basa en una serie de valores fundamentales que sirven como la base para todas las actividades realizadas bajo esta metodología. Estos valores no son solo conceptos abstractos, sino que actúan como verdaderos impulsores para las distintas actividades, acciones y tareas específicas que se llevan a cabo en el marco de XP.

El enfoque de Extreme Programming se centra en la orientación a objetos como su estilo de desarrollo preferido. Este enfoque incluye varias prácticas y reglas específicas a lo largo de las cuatro fases principales del proceso de desarrollo. Como se muestra en la Figura 3 con todo su proceso.

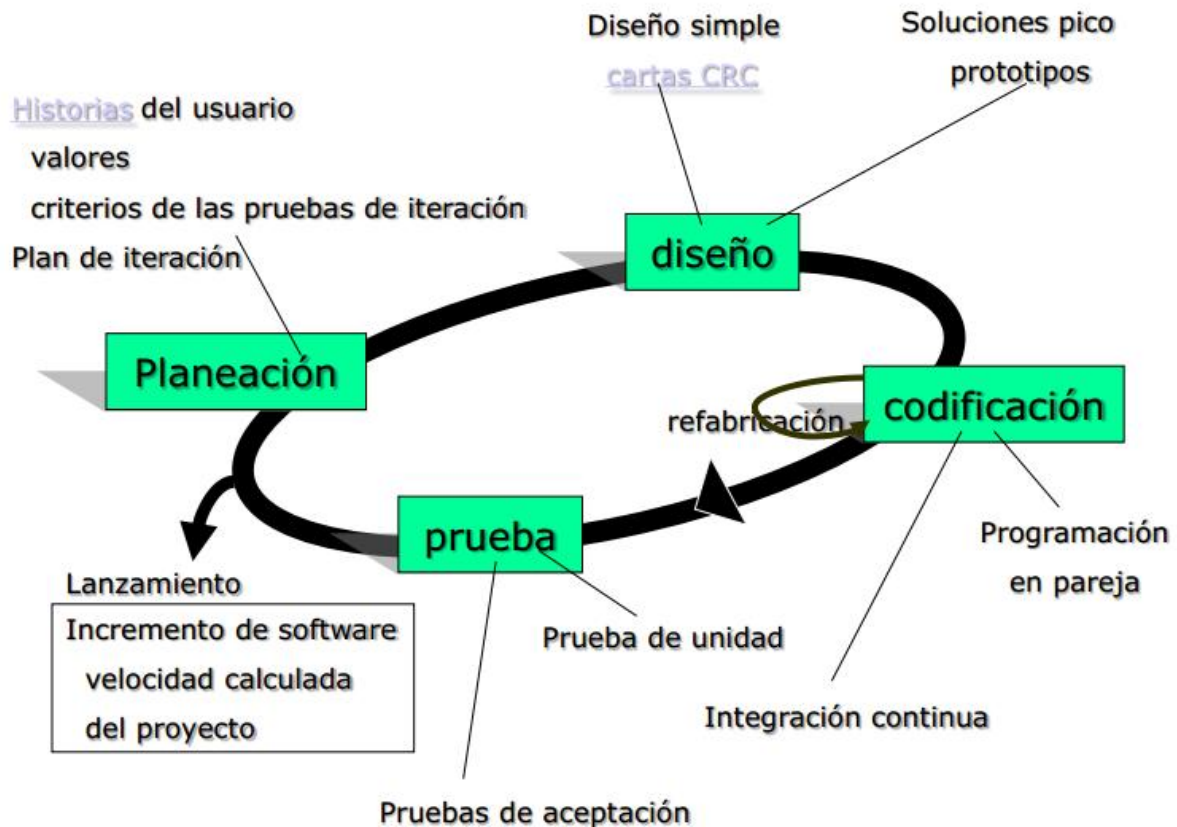


Figura 1: Proceso XP.
Fuente: Karla Cevallos [17].

3.2.1. Análisis crítico de la selección de XP frente a otras metodologías

A la hora de seleccionar la metodología de desarrollo más adecuada, se evaluaron varias alternativas ágiles, incluyendo Scrum y Kanban, junto con Extreme Programming (XP). Aunque todas estas metodologías comparten principios básicos como la colaboración del equipo, la entrega incremental y la respuesta rápida al cambio, XP sobresalió por varias razones.

Scrum, por ejemplo, es una metodología ampliamente utilizada que estructura el trabajo en sprints, lo que permite entregas periódicas y planificación detallada. Sin embargo, en proyectos con requisitos altamente cambiantes y necesidad de retroalimentación continua del cliente, esta estructura fija de los sprints puede ser menos flexible que XP, donde las iteraciones son más cortas y la adaptación al cambio es más fluida. Además, Scrum no enfatiza tanto algunas prácticas técnicas clave como la programación en pares y las pruebas continuas, que son esenciales para mantener una alta calidad de código.

Kanban, por otro lado, ofrece un enfoque visual y flexible para la gestión del flujo de trabajo, lo que facilita la identificación de cuellos de botella y la mejora continua del proceso. Sin embargo, Kanban carece de una estructura definida de roles y ceremonias que fomenten una colaboración estrecha y constante del equipo, algo que XP garantiza a través de sus prácticas colaborativas y sus valores fundamentales.

En resumen, XP fue elegida porque combina lo mejor de la adaptabilidad al cambio

con un enfoque riguroso en la calidad del software. La programación en pares, las pruebas continuas y la integración frecuente permiten reducir errores y mejorar la eficiencia general del desarrollo. Además, su énfasis en la comunicación constante con el cliente asegura que el producto final cumpla con las expectativas reales. Frente a metodologías como Scrum y Kanban, XP sobresale en proyectos donde la calidad técnica y la adaptación continua son críticas.

3.2.2. Limitaciones encontradas durante el desarrollo y cómo se abordaron

Durante el desarrollo del proyecto utilizando Extreme Programming, se encontraron algunas limitaciones que requirieron soluciones adaptativas para garantizar el éxito del proceso.

Una de las principales limitaciones fue la curva de aprendizaje inicial para los miembros del equipo que no estaban familiarizados con prácticas específicas de XP, como la programación en pares y la integración continua. Para abordar este problema, se organizaron sesiones de capacitación y talleres prácticos donde los desarrolladores pudieron familiarizarse con estas prácticas y mejorar sus habilidades.

Otra dificultad significativa fue mantener la disciplina en la realización de pruebas frecuentes y la integración continua. A medida que el proyecto creció en complejidad, el volumen de pruebas aumentó considerablemente, lo que provocó tiempos de ejecución más largos. Para mitigar este problema, se implementaron herramientas de automatización de pruebas y se optimizaron los entornos de integración para reducir los tiempos de retroalimentación.

Finalmente, debido a la naturaleza dinámica del proyecto y la necesidad de adaptarse continuamente a los cambios en los requisitos, surgieron problemas de comunicación entre el equipo de desarrollo y los stakeholders. Para resolver esto, se incrementó la frecuencia de las reuniones de retroalimentación y se establecieron canales de comunicación claros y abiertos que facilitaron la colaboración efectiva y la comprensión de los cambios necesarios.

En general, estas limitaciones fueron superadas mediante un enfoque proactivo, el uso de herramientas adecuadas y el fortalecimiento de la comunicación y la capacitación del equipo. Esto permitió que el proceso de desarrollo se mantuviera alineado con los objetivos del proyecto, garantizando un producto final de alta calidad.

Capítulo 4, ingeniería del proyecto

4.1. Requisitos de software

Los requisitos funcionales representan el núcleo de las habilidades y características que el sistema debe poseer para cumplir con las demandas operativas de la evaluación de resultados de aprendizaje en la Universidad del Valle. Estos requisitos proporcionan una guía esencial para los desarrolladores, delineando las funcionalidades y acciones que el sistema debe ejecutar para optimizar la gestión de la evaluación, seguimiento de indicadores de logro, almacenamiento de evidencias y demás aspectos claves para el cumplimiento del Decreto 1330 del Ministerio de Educación. A continuación, se especificarán los requisitos funcionales de este prototipo. El detalle de cada uno de ellos se encuentra en el Anexo A.

4.1.1. Requisitos funcionales

Gestión de usuarios

- Creación de usuarios.
- Actualizar datos de perfil.
- Gestión de roles de usuario.
- Restablecimiento de contraseña.
- Cierre de sesión.
- Inicio de sesión.
- Visualizar perfil de usuario.
- Suministro de información.
- Desactivación de usuarios.

Gestión de programas universitarios

- Creación de un programa.
- Asignación de coordinador.

Gestión de cursos

- Gestión de cursos y grupos.
- Retroalimentación del profesor.
- Asociación de profesores a cursos y grupos.
- Gestión de resultados de aprendizaje e indicadores de logro.

Gestión de resultados de aprendizaje e indicadores de logro

- Asignación de indicadores de logro.

Registro de calificaciones

- Registro de calificaciones.

Registro de Evidencias

- Gestión de evidencias.

Visualización del progreso

- Visualización del progreso del curso y grupo.
- Visualización del progreso con gráficos y tablas.

Generación de informes

- Generación de informes de rendimiento académico.
- Generación de informes de rendimiento académico individual.

4.1.2. Requisitos no funcionales

Los requisitos no funcionales definen las características y restricciones bajo las cuales el sistema debe operar. En este sentido, estos requisitos representan criterios esenciales que determinan la usabilidad, seguridad y eficiencia del sistema. Por tanto, proporcionan una guía crucial para los desarrolladores, esbozando las condiciones y estándares que el sistema debe cumplir para garantizar un rendimiento óptimo, proteger la información y ofrecer una experiencia de usuario intuitiva y eficiente. Estos requisitos son fundamentales para el funcionamiento eficaz del sistema, y su cumplimiento asegura que el sistema pueda satisfacer las demandas operativas de la Universidad del Valle y sus usuarios de manera segura y eficiente. A continuación, se especificarán los requisitos no funcionales de este prototipo. El detalle de cada uno de ellos se encuentra en el Anexo A.

- Seguridad.
- Usabilidad.
- Desempeño.
- Mantenibilidad.
- Disponibilidad.

4.2. Base de datos

En el proceso de planificación de la estructura de datos para la nueva plataforma educativa, se llevó a cabo un análisis detallado de las necesidades de almacenamiento y gestión de información. Desde las primeras etapas del proyecto, se realizaron diversas sesiones de trabajo para definir la organización óptima de la base de datos. Durante este proceso, se consideraron las mejores prácticas en diseño de sistemas de información, así como experiencias previas relevantes. Además, se consultaron diversas fuentes de referencia y se evaluaron modelos utilizados en plataformas educativas similares, asegurando así la eficiencia y la capacidad de escalamiento del sistema. La estrecha colaboración con los responsables del proyecto, incluidos líderes académicos y técnicos, permitió alinear de manera precisa los requisitos funcionales con el diseño final del modelo de datos. Se presenta un resumen del modelo resultante, el detalle se encuentra en el Anexo D

Usuario
+ ID + nombre + apellido + email + programa asignado + tipo identificacion + numero identificacion + codigo + rol + sede + tel

Cuadro 3: Tabla Usuario 1

Curso
+ codigo + nombre curso + Status + Profesor + ID programa + credits + intensidad horaria + habilitable + validable + prerrequisitos

Cuadro 4: Tabla Curso

Resultado de aprendizaje
+ codigo + nombre resultado + descripcion + ID curso

Cuadro 5: Tabla Resultado de aprendizaje

Indicador de logro
+ codigo + nombre indicador + descripcion + ID resultado

Cuadro 6: Tabla Indicador de logro

Programa
+ ID + ID coordinador + nombre programa + facultad + duracion + periodicidad de admisiones + modalidad + jornada + creditos + registro ICFES + registro SNIES + resolucion MEN + resolucion del PENSUM + fecha de creacion + sede + email

Cuadro 7: Tabla Programa

Grupo
+ codigo + numero grupo + ID curso

Cuadro 8: Tabla Grupo

En la Figura 2 se presenta el modelo entidad-relación resultante, donde se ilustran todas las relaciones existentes entre las tablas.

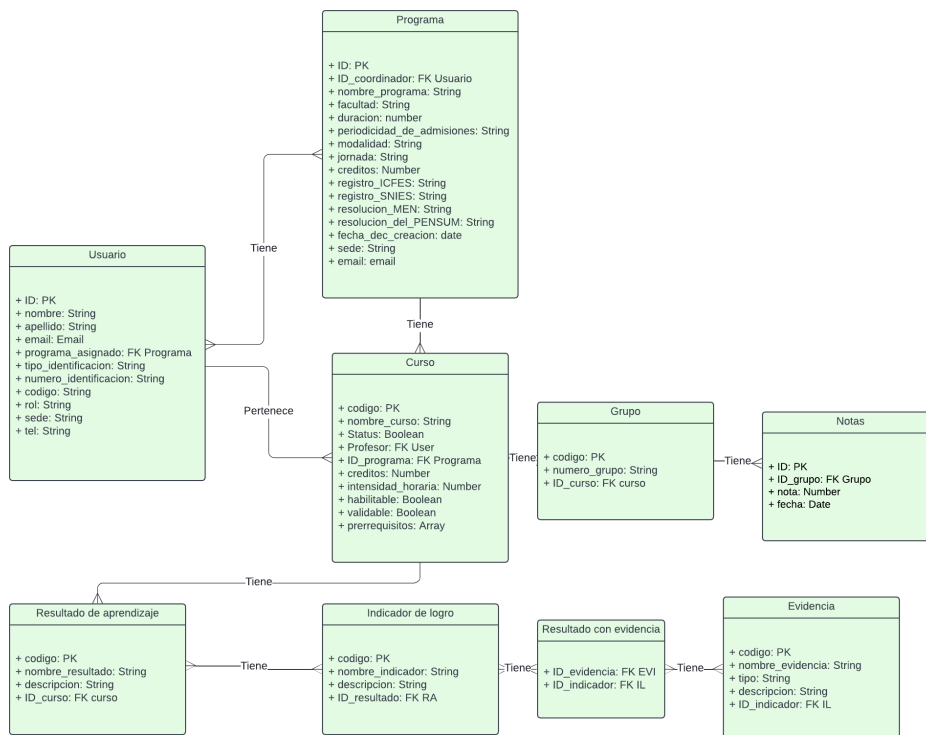


Figura 2: Modelo entidad relación con tablas mas detalladas

4.3. Arquitectura

La arquitectura de este proyecto fusiona tecnologías de vanguardia con prácticas avanzadas de desarrollo para garantizar un rendimiento óptimo, seguridad robusta y escalabilidad. En el frontend, se opta por Next.js en conjunto con React por su integración fluida y capacidades de renderizado del lado del servidor (SSR), proporcionando una navegación rápida y eficiente entre secciones de la aplicación.

En el backend, se adopta Node.js como base, aprovechando Firebase como plataforma principal. Firebase destaca por su manejo eficiente de Firestore, una base de datos NoSQL que permite almacenar datos dinámicos y mantener sincronización en tiempo real. Además, Firebase es elegido específicamente por su capacidad para gestionar archivos como CSV, PDF y otros formatos, facilitando la gestión de contenido multimedia y documentos dentro de la aplicación.

Se implementarán prácticas avanzadas de seguridad, como la autenticación robusta a través de Firebase Authentication y el uso de estrictas reglas de seguridad en Firebase Security Rules. Estas medidas garantizan la protección contra vulnerabilidades, asegurando la integridad y confidencialidad de los datos sensibles de los usuarios.

Además, la arquitectura cliente-servidor permitirá una adaptación fluida tanto para la aplicación web como para futuras expansiones hacia dispositivos móviles, asegurando una experiencia de usuario consistente y optimizada en todos los dispositivos.

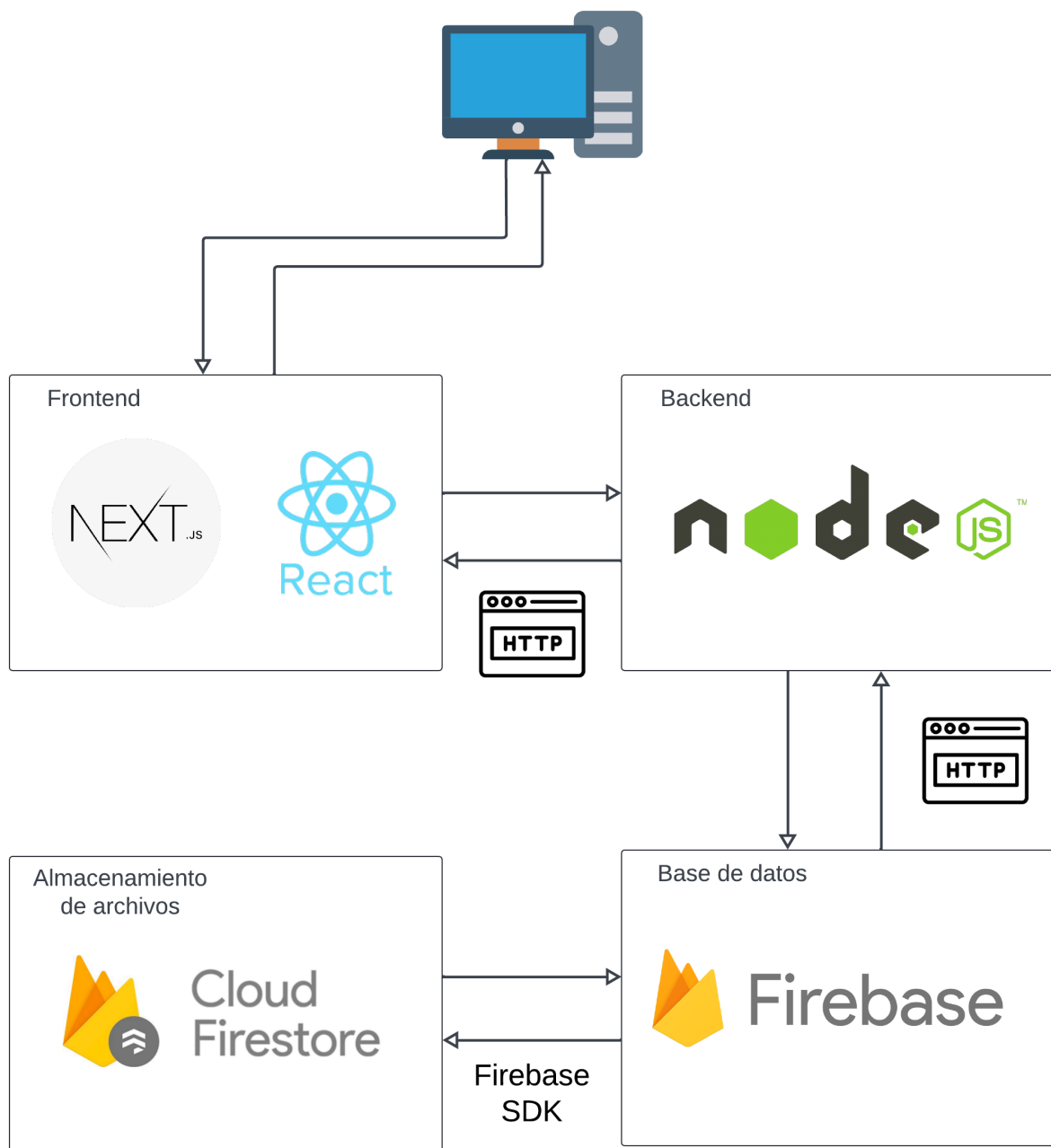


Figura 3: Diagrama Arquitectura utilizada

4.4. HTTP

HTTP (Hypertext Transfer Protocol) es un protocolo de comunicación que permite la transferencia de datos en la web entre un cliente (como un navegador) y un servidor. Es la base de las aplicaciones web y define cómo se estructuran y envían las solicitudes y respuestas, permitiendo a los usuarios acceder a recursos como páginas web, imágenes y archivos.

Métodos

- **GET:** Solicita un recurso específico del servidor sin modificarlo. Es el método más común y se usa para obtener datos, como al cargar una página web.
- **POST:** Envía datos al servidor para crear un nuevo recurso o para realizar alguna acción. Por ejemplo, cuando se envía un formulario de registro, se utiliza POST para transmitir los datos.
- **PUT:** Actualiza un recurso existente en el servidor con nuevos datos. Es común en aplicaciones donde se permite modificar la información almacenada, como actualizar un perfil de usuario.
- **DELETE:** Elimina un recurso específico en el servidor. Se usa en aplicaciones donde los usuarios pueden borrar contenido, como comentarios o archivos.
- **HEAD:** Similar a GET, pero solo solicita las cabeceras de la respuesta sin el cuerpo. Se usa para verificar si un recurso está disponible o para obtener información sobre él sin descargarlo.
- **OPTIONS:** Consulta al servidor sobre los métodos HTTP permitidos para un recurso específico. Es útil para conocer las operaciones permitidas antes de realizar solicitudes.

HTTP se utiliza para manejar la comunicación entre el cliente y el servidor, facilitando la transferencia de datos y el acceso a los recursos de la aplicación. Esta integración permite que los usuarios interactúen de manera fluida con el sistema, ya sea enviando o recibiendo información, y asegura que la experiencia sea rápida y eficiente en cada solicitud realizada.

4.5. Herramientas utilizadas

4.5.1. Next.js

Next.js fue la base fundamental para estructurar el frontend de la plataforma, aprovechando su poderosa funcionalidad de renderizado en el servidor (SSR) para mejorar significativamente la velocidad de carga inicial y optimizar el rendimiento en general. Gracias al SSR, las páginas web se generan en el servidor y se entregan al usuario como HTML ya renderizado, lo que reduce el tiempo hasta el primer byte y mejora la indexación en motores de búsqueda. Además, la posibilidad de utilizar la renderización estática (SSG) para secciones de la plataforma que no cambian frecuentemente, como las páginas de contenido general o las de resultados históricos, permitió mejorar aún más el tiempo de carga y la accesibilidad. El sistema de enrutamiento dinámico de Next.js facilitó la creación de rutas personalizadas y la administración de módulos de gestión como los de usuarios, cursos y resultados de aprendizaje, lo que no solo simplificó la navegación, sino que también permitió escalar la plataforma de manera eficiente.

La optimización automática de la carga de recursos, como imágenes y fuentes, así como el soporte para dividir el código en fragmentos más pequeños, permitió a Next.js ofrecer una experiencia de usuario más fluida, especialmente en dispositivos con conexión de red más lenta. Además, el soporte nativo para la internacionalización (i18n) permitió que la plataforma pudiera ser adaptada fácilmente a diferentes idiomas, ampliando su accesibilidad a nivel global. En términos de desarrollo, Next.js simplifica la creación de aplicaciones React al ofrecer una estructura organizada que facilita el mantenimiento del código a largo plazo. Gracias a su robusto ecosistema, Next.js también facilitó la integración de herramientas de terceros como analíticas, gestión de contenidos y APIs personalizadas.

4.5.2. Firebase

Firebase fue seleccionado como la solución backend principal de la plataforma debido a su arquitectura flexible y altamente escalable. Su sistema Firestore, una base de datos NoSQL en tiempo real, permitió almacenar y sincronizar datos de manera eficiente, incluyendo información crítica como el perfil de usuarios, calificaciones, evidencias de aprendizaje y resultados de pruebas. Firestore no solo proporciona alta disponibilidad y rendimiento, sino que su capacidad para gestionar datos en tiempo real permite a los usuarios recibir actualizaciones instantáneas sin necesidad de recargar la página, lo que resultó en una experiencia mucho más dinámica e interactiva.

Además, Firebase Storage fue utilizado para almacenar y gestionar archivos y documentos adjuntos, como tareas entregadas, materiales de curso y evidencias de aprendizaje, utilizando una infraestructura en la nube que garantiza una alta disponibilidad y baja latencia. Firebase también facilitó la configuración de reglas de seguridad detalladas, que protegieron los datos sensibles de los usuarios, asegurando que solo personas autorizadas pudieran acceder o modificar ciertos recursos. Las funciones de seguridad avanzadas basadas en reglas de Firestore también se combinaron con la autenticación de Firebase, lo que permitió un acceso seguro y controlado a diferentes partes de la plataforma según los roles de los usuarios.

Una de las ventajas clave de Firebase fue su facilidad de integración con otros servicios de Google, como Firebase Analytics, lo que permitió obtener valiosos datos sobre el comportamiento del usuario en tiempo real, mejorando la toma de decisiones basadas en datos. La sincronización en tiempo real proporcionada por Firebase permitió a los usuarios ver las actualizaciones de las calificaciones, comentarios y resultados inmediatamente después de su publicación, lo que mejoró la interacción de los usuarios y permitió una retroalimentación constante. En términos de escalabilidad, Firebase ofreció una solución completamente gestionada que permitió al equipo de desarrollo centrarse en las funcionalidades de la plataforma sin preocuparse por la administración de la infraestructura.

4.5.3. Nodemailer

Nodemailer fue utilizado para facilitar el envío de correos electrónicos automatizados desde la plataforma, un aspecto esencial para mejorar la comunicación entre el sistema y sus usuarios. Esta herramienta permitió implementar una serie de notificaciones automáticas, como los correos electrónicos de bienvenida para nuevos usuarios, restablecimiento de contraseñas, confirmaciones de registro, alertas de cambios en el estado de las evaluaciones y recordatorios de tareas pendientes. El uso de Nodemailer facilitó la automatización de estas comunicaciones, reduciendo la necesidad de intervención manual y asegurando que los usuarios recibieran información crítica en tiempo y forma.

Nodemailer soporta múltiples transportadores de correo, lo que permitió al equipo de desarrollo configurar fácilmente integraciones con diferentes servicios de correo electrónico, como Gmail, SendGrid y otros, optimizando la entrega y fiabilidad de los mensajes. Además, su compatibilidad con autenticación segura mediante OAuth2 garantizó que los correos electrónicos se enviaran de forma segura, sin comprometer la privacidad de los datos del usuario. La integración de plantillas personalizadas de correo electrónico también permitió mejorar la profesionalidad y la coherencia visual de las notificaciones enviadas, brindando a los usuarios una experiencia más cohesiva y agradable. Gracias a su flexibilidad, Nodemailer permitió ajustar los envíos de correos electrónicos a las necesidades específicas del proyecto, adaptando los mensajes según el tipo de usuario (administrador, docente, estudiante) y las acciones que estos realizaran en la plataforma.

4.5.4. NextAuth

NextAuth fue fundamental en la gestión de la autenticación de usuarios en la plataforma, proporcionando un sistema flexible y seguro para el inicio de sesión y la gestión de sesiones. Implementamos flujos de inicio de sesión persistente, lo que significaba que los usuarios podían permanecer conectados incluso después de cerrar el navegador, mejorando la experiencia del usuario al eliminar la necesidad de iniciar sesión repetidamente. Además, NextAuth permitió implementar la autenticación mediante múltiples proveedores, incluyendo credenciales tradicionales como correo electrónico y contraseña, así como opciones de inicio de sesión social, como Google y GitHub. Esta flexibilidad no solo facilitó la autenticación de los usuarios, sino que también abrió la puerta para futuras integraciones con otros proveedores.

La integración de NextAuth con el sistema de roles y permisos también fue un aspecto clave. Los roles, como administrador, docente y estudiante, fueron definidos y gestiona-

dos dentro del sistema, lo que permitió controlar el acceso a diferentes secciones de la plataforma de manera granular. Los administradores tenían acceso a todas las funcionalidades de gestión, mientras que los docentes solo podían acceder a los módulos relevantes para su labor educativa, y los estudiantes solo podían ver los cursos y resultados que les correspondían. La gestión de sesiones segura de NextAuth, combinada con la protección de rutas, aseguró que solo los usuarios autenticados y autorizados pudieran acceder a secciones específicas del sistema.

4.5.5. Autenticación de Firebase

La integración de Firebase Authentication proporcionó una solución de autenticación altamente segura, escalable y fácil de implementar, esencial para la gestión de usuarios en la plataforma. Los métodos de autenticación, como el inicio de sesión mediante correo electrónico y contraseña, fueron configurados para ofrecer una experiencia de usuario fluida. Además, Firebase Authentication soportó múltiples métodos adicionales, como la autenticación mediante redes sociales (por ejemplo, Google), lo que facilitó el acceso a la plataforma sin la necesidad de crear nuevas credenciales.

Firebase Authentication también facilitó la implementación de flujos de recuperación de contraseñas y verificación de correos electrónicos, lo que aumentó la seguridad y redujo el riesgo de accesos no autorizados. La gestión centralizada de los usuarios en Firebase también simplificó la sincronización de los datos de los usuarios con otros servicios de Firebase, como Firestore y Firebase Storage, lo que permitió mantener un control completo sobre los datos de los usuarios, mejorando la eficiencia en el desarrollo y la gestión del sistema.

4.5.6. Zod

Zod fue una herramienta esencial para garantizar que los datos enviados a través de los formularios de la plataforma cumpliera con las estructuras y formatos adecuados. Su uso permitió una validación de datos robusta y confiable, asegurando que solo los datos válidos llegaran al backend, reduciendo el riesgo de errores y aumentando la seguridad general del sistema. Además, Zod es compatible con TypeScript, lo que permitió aprovechar los beneficios de tipado estático, mejorando la experiencia de desarrollo y evitando errores comunes relacionados con tipos de datos incorrectos.

Zod facilitó la creación de esquemas de validación complejos para formularios y entradas de usuario, como la validación de correos electrónicos, contraseñas seguras, fechas y otros campos específicos de los formularios. La capacidad de Zod para validar datos asíncronos también permitió implementar procesos como la verificación de disponibilidad de correos electrónicos de forma eficiente, lo que mejoró la experiencia del usuario al prevenir errores en tiempo real. En conjunto con TypeScript, Zod permitió mantener un código limpio, claro y seguro, lo que resultó en una plataforma más robusta y menos propensa a fallos.

4.5.7. React

React fue la piedra angular en la creación de la interfaz de usuario, proporcionando una estructura altamente dinámica y eficiente para la construcción de componentes reutilizables. Cada sección de la plataforma, desde los formularios hasta los paneles de administración, fue construida como un conjunto de componentes modulares, lo que facilitó la gestión y actualización del código a medida que la plataforma crecía. React permitió a la plataforma actualizar la interfaz de usuario en tiempo real sin necesidad de recargar la página, mejorando la fluidez y la experiencia general del usuario.

Además, el uso de React Context API y Redux permitió gestionar el estado de la aplicación de manera centralizada, lo que facilitó la comunicación entre componentes y permitió un manejo eficiente de los datos. Esta capacidad de manejar el estado de forma efectiva fue crucial para el seguimiento de los avances de los estudiantes, la actualización de resultados en tiempo real y la visualización de estadísticas y gráficos.

4.5.8. Axios

Axios fue la herramienta elegida para la gestión de solicitudes HTTP entre el frontend y el backend, facilitando la interacción con la base de datos Firestore y otros servicios en la nube. Axios permitió realizar operaciones como la carga, actualización y eliminación de datos en Firestore, garantizando que la plataforma estuviera siempre sincronizada con los cambios realizados por los usuarios. La capacidad de Axios para interceptar solicitudes y respuestas permitió una implementación centralizada de la autenticación, gestión de errores y procesamiento de respuestas, lo que mejoró la eficiencia y la seguridad de la aplicación.

Además, Axios proporcionó una forma sencilla de manejar las peticiones de red, lo que facilitó la integración de APIs externas y servicios de terceros, como los servicios de correo electrónico de Nodemailer. Esta flexibilidad en el manejo de solicitudes HTTP hizo que la plataforma fuera más fácil de mantener y expandir, permitiendo al equipo de desarrollo agregar nuevas funcionalidades rápidamente.

4.5.9. React Icons

React Icons se utilizó para mejorar la experiencia visual de la plataforma. Al integrar una amplia variedad de iconos dentro de los botones, menús y formularios, React Icons contribuyó a hacer la plataforma más intuitiva y fácil de usar. La capacidad de integrar estos iconos directamente en los componentes de React permitió mantener un diseño limpio y moderno sin necesidad de agregar imágenes externas, lo que también mejoró el rendimiento general de la plataforma.

La biblioteca de React Icons es altamente flexible, permitiendo personalizar el tamaño, color y estilo de los iconos según las necesidades de la interfaz. Esta personalización también permitió que los iconos fueran consistentes con la identidad visual de la plataforma, garantizando que la apariencia fuera profesional y coherente en todas las pantallas.

4.5.10. Tailwind CSS

Tailwind CSS fue la herramienta clave para crear un diseño moderno, flexible y atractivo para la plataforma. Utilizando un enfoque basado en clases utilitarias, Tailwind permitió crear una interfaz de usuario completamente responsiva y adaptativa a diferentes dispositivos y tamaños de pantalla sin necesidad de escribir código CSS complejo. Este enfoque aceleró el proceso de desarrollo y facilitó la creación de componentes reutilizables, lo que resultó en una base de código más limpia y fácil de mantener.

Tailwind CSS se integró perfectamente con React, permitiendo la creación rápida de componentes con estilos personalizados sin necesidad de escribir extensas hojas de estilo. Además, su capacidad para crear interfaces completamente responsivas permitió que la plataforma tuviera un excelente rendimiento en dispositivos móviles y de escritorio, asegurando que los usuarios pudieran acceder a la plataforma sin importar el dispositivo que estuvieran utilizando.

4.5.11. ESLint

ESLint fue una herramienta esencial en el proceso de desarrollo del proyecto, ya que permitió mantener un código limpio, consistente y libre de errores. Su integración con el entorno de desarrollo, especialmente con editores de texto como Visual Studio Code, permitió la detección en tiempo real de errores de sintaxis, convenciones de estilo y posibles problemas en el código, lo que facilitó la corrección inmediata y evitó que los errores llegaran a las etapas de producción. Al aplicar reglas predefinidas y configuradas específicamente para el equipo de desarrollo, ESLint garantizó que todos los miembros siguieran las mismas pautas de codificación, lo que resultó en un código más legible, coherente y fácil de mantener.

Una de las características clave de ESLint es su capacidad para detectar errores potenciales antes de que el código sea ejecutado. Esto incluyó la identificación de variables no utilizadas, funciones sin retorno o condiciones redundantes, lo que ayudó a mejorar la calidad y el rendimiento del código desde las primeras etapas del desarrollo. Además, su integración con herramientas de CI/CD permitió la validación automática del código en cada commit, asegurando que los cambios realizados por los desarrolladores no introdujeran nuevos errores y cumplieran con los estándares del proyecto.

El uso de ESLint también fue clave para reducir el tiempo de revisión de código durante las fases de control de calidad y pruebas. Los desarrolladores pudieron centrarse en la lógica del negocio y las nuevas funcionalidades, mientras que ESLint se encargaba de asegurar que el código se ajustara a los estándares establecidos. Esto no solo aumentó la productividad, sino que también contribuyó a una base de código más estable y predecible, lo que facilitó la implementación de nuevas características sin generar conflictos o problemas de integración.

4.5.12. Scheduler

Scheduler se utilizó para programar tareas automatizadas en la plataforma, como la generación de informes periódicos y el envío de notificaciones. Esto optimizó la gestión

de procesos recurrentes y mejoró la eficiencia en la administración de las tareas dentro del sistema. Su capacidad para manejar tareas programadas de manera precisa permitió implementar funcionalidades clave sin afectar el rendimiento general de la plataforma.

4.6. Control de versiones

En este proyecto se utilizó control de versiones con GitHub, lo que permitió gestionar de manera eficiente el desarrollo y seguimiento de los cambios en el código. Se emplearon dos ramas principales para organizar el flujo de trabajo:

Rama Develop: En esta rama se subían los cambios que estaban en desarrollo. Todos los nuevos avances, correcciones y características se implementaban aquí, y se realizaban las pruebas necesarias para asegurarse de que todo funcionara correctamente antes de integrarlos a la versión estable. Era el espacio donde se desarrollaba y validaba el código antes de su despliegue.

Rama Main: Esta es la rama principal y estable, que representa la versión final del producto. Una vez que los cambios en la rama develop eran aprobados y probados, se fusionaban con la rama main. Esta rama main estaba desplegada en el servicio gratuito de Vercel, lo que permitía tener siempre disponible una versión actualizada y funcional del proyecto en producción para su acceso público o pruebas finales.

Este flujo de trabajo con dos ramas permitió mantener un control claro sobre las etapas de desarrollo y producción, asegurando que la versión que estaba en main siempre fuera estable y lista para su despliegue, mientras que la rama develop servía como un entorno para el desarrollo continuo y las pruebas. En la imagen a continuación se puede ver un ejemplo Figura 3

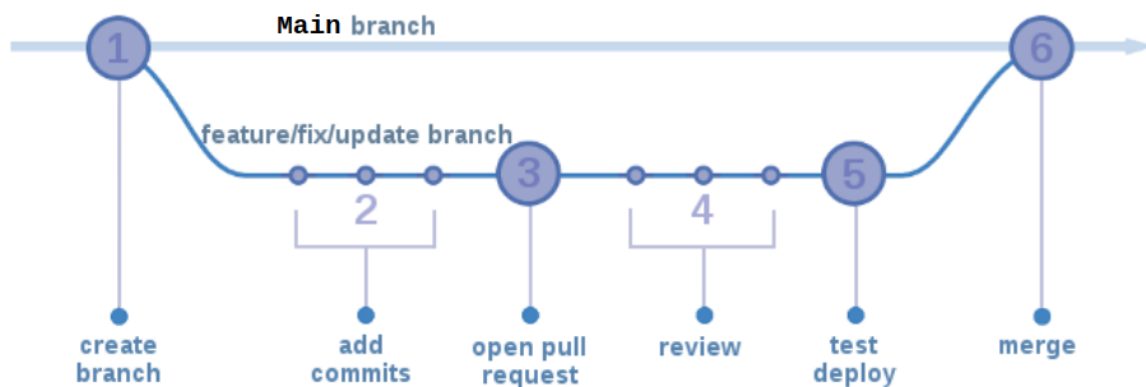


Figura 4: Proceso Github ramas
Fuente: Flavio Antunes [18].

4.7. Mockups

Un mockup es una representación visual estática que muestra cómo lucirá una página web o aplicación. Incluye colores, tipografías, imágenes y la distribución de los elementos. Es una herramienta fundamental para planificar el diseño y asegurar que cumpla con las expectativas antes de pasar al desarrollo.

En este caso, los mockups presentados están hechos en Figma, una herramienta muy utilizada en diseño de interfaces por su capacidad para colaborar en tiempo real y crear diseños detallados. Sin embargo, también existen otras herramientas como Sketch, Adobe XD o incluso soluciones más simples como Canva, que permiten desarrollar diseños según las necesidades y el nivel de detalle requerido. Estas herramientas ayudan a crear vistas de páginas, formularios, menús y otros elementos clave para garantizar un diseño funcional y estético.

A continuación se presentarán algunos mockups como muestra: Figura 4, Figura 4, Figura 5



Universidad del Valle

Correo
correo@correounivalle.edu.co

Contraseña

Iniciar Sesión

☐ Recordar contraseña [¿Has olvidado tu contraseña?](#)



Figura 5: Mockups Inicio de Sesión

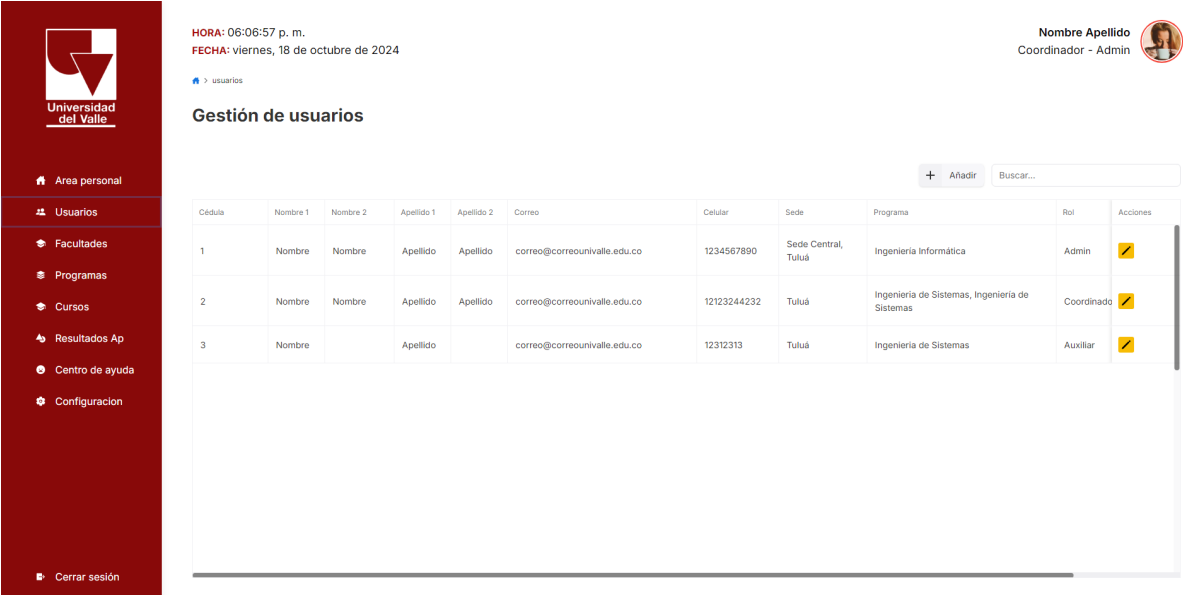


Figura 6: Mockups Gestión Usuarios



Figura 7: Mockups Gestión Resultados AP

4.8. Reportes

4.8.1. Reporte por grupo

Este reporte presenta el nivel de cumplimiento por grupo de los resultados de aprendizaje basado en las calificaciones proporcionadas por el profesor. Estas notas han sido ponderadas y asociadas a cada uno de los indicadores de logro, reflejando un porcentaje de cumplimiento para cada uno.

Gráficos El informe incluye una gráfica circular que ilustra el porcentaje de cumplimiento por resultado de aprendizaje. Cada segmento de la gráfica está representado con un color distinto, permitiendo una visualización clara del progreso alcanzado.

Tabla Resumen Este reporte permite identificar de manera visual y cuantitativa el desempeño de los estudiantes frente a los objetivos de aprendizaje establecidos.

4.8.2. Reporte por estudiante

Este reporte detalla el nivel de cumplimiento de los resultados de aprendizaje para cada estudiante, basado en las calificaciones proporcionadas por el profesor. Estas notas han sido ponderadas y asociadas a los indicadores de logro, mostrando el porcentaje de cumplimiento correspondiente a cada resultado.

Gráfico de Cumplimiento El informe presenta una gráfica circular para cada estudiante, que visualiza el porcentaje de cumplimiento por resultado de aprendizaje. Cada segmento de la gráfica utiliza un color distinto, facilitando la interpretación del nivel de logro alcanzado en cada área evaluada.

Detalle del Reporte Además del gráfico, se incluye un resumen numérico de los resultados individuales, destacando:

Resultados evaluados. Porcentaje de cumplimiento en cada indicador. Áreas que requieren mayor atención. Este informe permite identificar tanto fortalezas como áreas de oportunidad en el aprendizaje de cada estudiante, contribuyendo a un seguimiento personalizado y efectivo.

4.9. Prototipo

En este apartado se presentan diversas capturas de pantalla del prototipo, tomadas desde el rol de administrador, el cual tiene acceso completo a todas las funcionalidades del sistema. El objetivo de mostrar estas imágenes es ofrecer una visión detallada y completa de las diferentes características y funcionalidades que ofrece el prototipo. Al hacerlo desde el rol de administrador, se puede observar cómo se gestionan todos los módulos del sistema, ya que este rol tiene privilegios para interactuar con todas las áreas y configuraciones disponibles.

Cada una de las capturas destaca cómo los módulos responden y cumplen con los requisitos previamente definidos para resolver el problema planteado. Además, se evidencian aspectos clave como la navegación, la interacción con las diversas herramientas y la accesibilidad de las funciones, lo que asegura que el prototipo no solo sea funcional, sino también intuitivo y eficiente. Esta sección permite validar que todas las funcionalidades necesarias están correctamente implementadas y alineadas con los objetivos del proyecto, ofreciendo una solución robusta y bien estructurada desde la perspectiva administrativa. A continuación se mostrarán figuras de las funcionalidades mas importantes del prototipo:

De la Figura 7, a la Figura 19

En Figuras completas puede ver todas las figuras del prototipo.

Universidad del Valle

HORA: 01:32:13 a. m.
FECHA: sábado, 30 de noviembre de 2024

Austin Santos
Coordinador - Admin

usuarios

Gestión de usuarios

+ Añadir Buscar...

Cédula	Nombre 1	Nombre 2	Apellido 1	Apellido 2	Correo	Celular	Sede	Programa	Rol	Acciones
987654321	Juan	Carlos	Pérez	García	juan.perezasdawdsawd@example.com	1234567890	Sede Central, Tuluá	Ingeniería Informática	Admin	✖
12234445	Ana	Sofía	Arango	Ramírez	ana.sofia@gmail.com	12123244232	Tuluá	Ingeniería de Sistemas, Ingeniería de Sistemas	Coordinado	✖
1101001	Jose		Cosculluela		jose@hotmail.com	12312313	Tuluá	Ingeniería de Sistemas	Auxiliar	✖
10000000022	Copito		Nieve		copito@gmail.com	121212212	Tuluá	Contaduría pública	Profesor	✖
12223	Juan	Sebastian	Gonzales	Gonzales	juan.s.gonzales@correounivalle.edu.co	3126730344	Tuluá	Ingeniería en Alimentos, Ingeniería de Sistemas, Contaduría Pública, Administración de Empresas, Dietética y Nutrición	Admin, Coo Auxiliar	✖
1121831713	Prueba		Prueba		prueba3@gmail.com	1122344	Tuluá	Contaduría Pública, Tecnología en Electrónica	Profesor	✖
10101	Bryan	Steven	Martínez	Brítez	admin@admin.com	321503	Tuluá	Ingeniería de Sistemas	Coordinado	✖

Cerrar sesión

Figura 8: Modulo gestión usuario

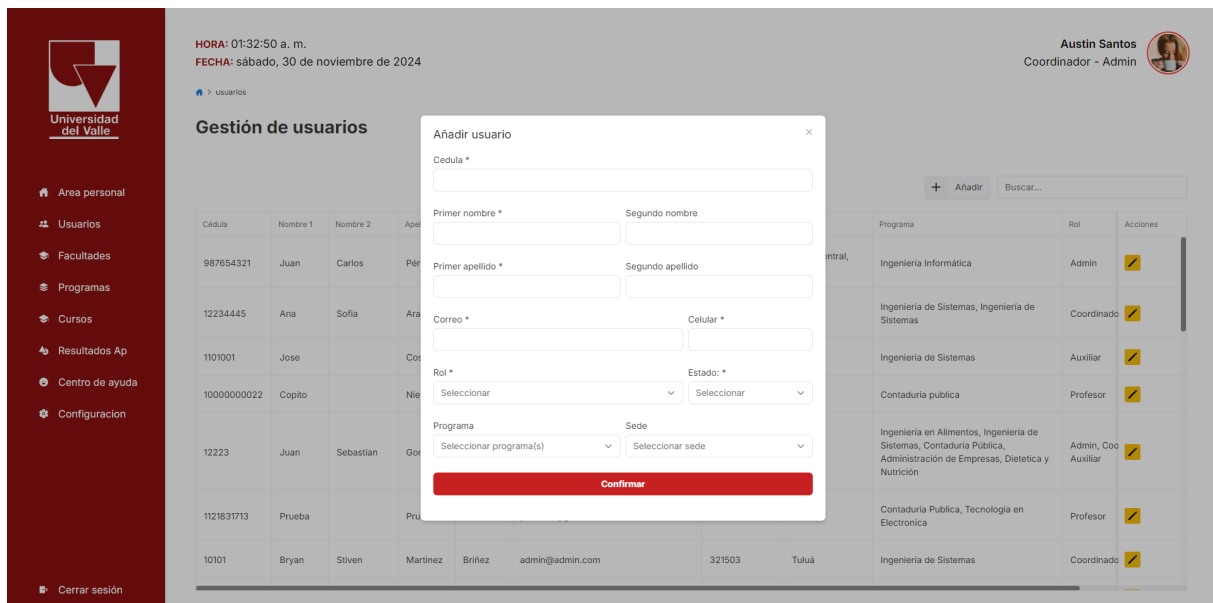


Figura 9: Creación de usuario

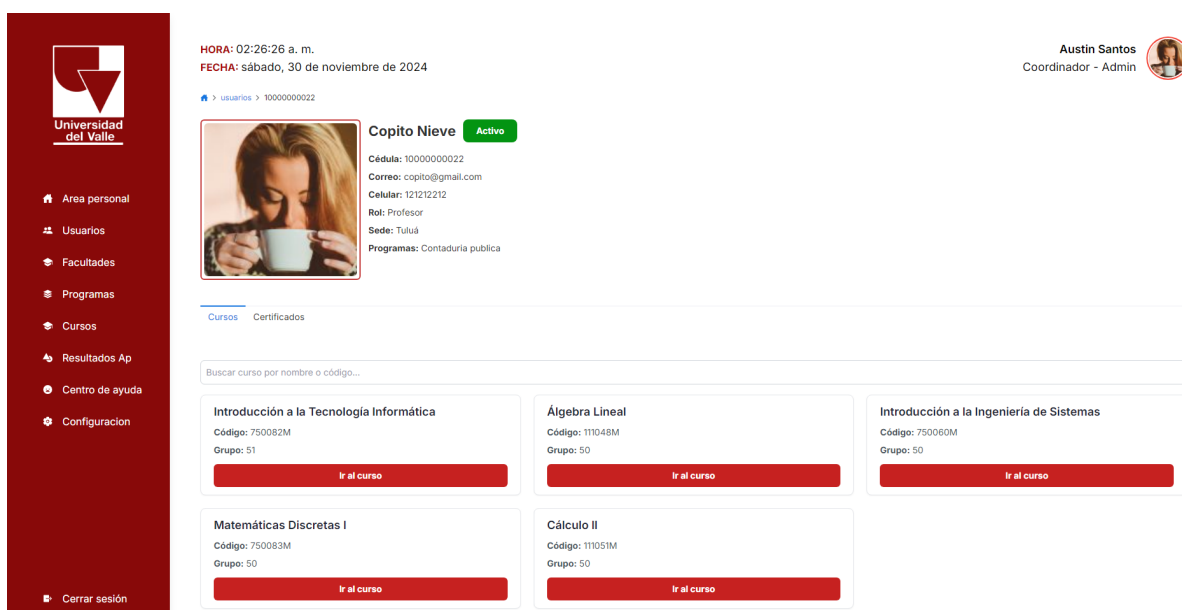


Figura 10: Usuario con sus cursos asignados



Universidad del Valle

- Area personal
- Usuarios
- Facultades
- Programas
- Cursos
- Resultados Ap
- Centro de ayuda
- Configuracion

Cerrar sesión

HORA: 02:27:32 a. m.
FECHA: sábado, 30 de noviembre de 2024

Austin Santos
Coordinador - Admin

usuarios > 10000000022



Copito Nieve Activo

Cédula: 10000000022
Correo: copito@gmail.com
Celular: 12121212
Rol: Profesor
Sede: Tulúa
Programas: Contaduría publica

Cursos Certificados

Gestión de Certificados

+ Añadir
 Buscar...


 Diploma Pregado.pdf

Figura 11: Usuario con sus certificados de estudio



Universidad del Valle

- Area personal
- Usuarios
- Facultades
- Programas
- Cursos
- Resultados Ap
- Centro de ayuda
- Configuracion

Cerrar sesión

HORA: 01:34:33 a. m.
FECHA: sábado, 30 de noviembre de 2024

Austin Santos
Coordinador - Admin

facultades

Gestión de facultades

+ Añadir
 Buscar...

Facultad	Correo	Telefono	Decano	Pagina web	Acciones
Facultad de Ciencias	email@universidad.edu.co	+57 602 3212100	Bryan Stiven Martinez Briñez	https://www.universidad.edu.co/facultad-ciencias	

Figura 12: Modulo gestión facultades

Universidad
del Valle

Area personal

Usuarios

Facultades

Programas

Cursos

Resultados Ap

Centro de ayuda

Configuracion

Cerrar sesión

HORA: 01:41:08 a. m.

FECHA: sábado, 30 de noviembre de 2024

Austin Santos

Coordinador - Admin

programas

Gestión de programas

+ Añadir

Buscar...

Codigo	Programa	Facultad	Sede	Jornada	Modalidad	Creditos	Semestres	Periodicidad	Fecha creación	Com	Acciones
2277	Contaduría Publica	Ingeniería de Sistemas	Tuluá	Diurna	Presencial	60	10	Semestral	2024-09-11	kevi	
3743	Ingeniería de Sistemas	Facultad de Ingeniería	Bogotá	Nocturna	Presencial	161	1	Semestral	2024-01-01	coo	
2177	Administración de empresas	Ciencias económicas y administrativas	Tuluá	Diurna	Presencial	60	8	Semestral	2024-09-11	adri	
3209	Tecnología en Electronica	Facultad de Ingeniería	Tuluá	Diurna	Presencial	60	8	Semestral	2024-09-11	dari	
98765	Contabilidad	Facultad de Ingeniería	Bogotá	Diurna	Presencial	160	10	Semestral	2021-05-15	coo	
2255	Dietética y Nutrición	Facultad de Ciencias de la Salud	Tuluá	Diurna	Presencial	60	8	Semestral	2003-01-17	diet	

Figura 13: Modulo gestión de programas

Universidad
del Valle

Area personal

Usuarios

Facultades

Programas

Cursos

Resultados Ap

Centro de ayuda

Configuracion

Cerrar sesión

HORA: 01:54:39 a. m.

FECHA: sábado, 30 de noviembre de 2024

Austin Santos

Coordinador - Admin

cursos

Gestión de cursos

+ Añadir

Buscar...

Codigo	Nombre	Programa	Creditos	Intensidad	Validable	Habilitable	Prerrequisitos	Estado	Acciones
750082M	Introducción a la Tecnología Informática	3743	2	2	No	No		Inactivo	
111048M	Álgebra Lineal	3743	3	5	Si	Si	111050M	Activo	
750080M	Fundamentos de Programación	3743	4	4	No	No		Activo	
750060M	Introducción a la Ingeniería de Sistemas	3743	2	2	No	No		Activo	
750083M	Matemáticas Discretas I	2177	4	4	Si	Si	750080M, 111050M	Activo	
106011M	Física I	2277	3	4	Si	Si	111050M	Activo	
111051M	Cálculo III	2177	123	123	Si	Si	750060M, 750080M	Activo	
111050M	Cálculo I	2177	4	4	Si	Si		Activo	
111051M	Cálculo II	2255	3	5	Si	No	111050M	Activo	

Figura 14: Modulo gestión cursos

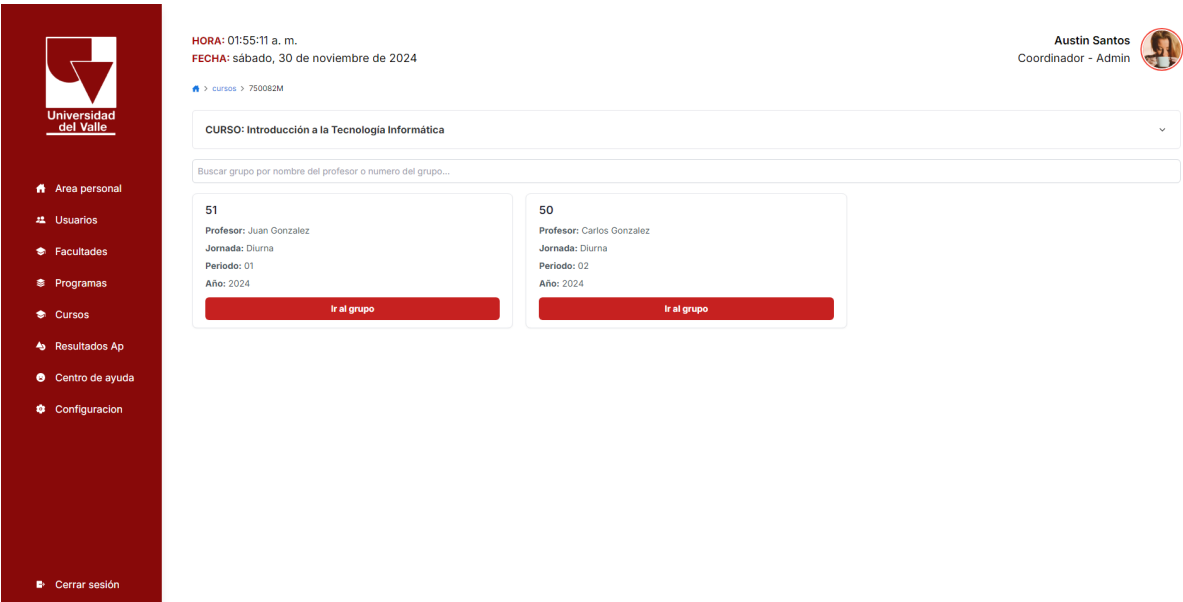


Figura 15: Cursos y sus grupos

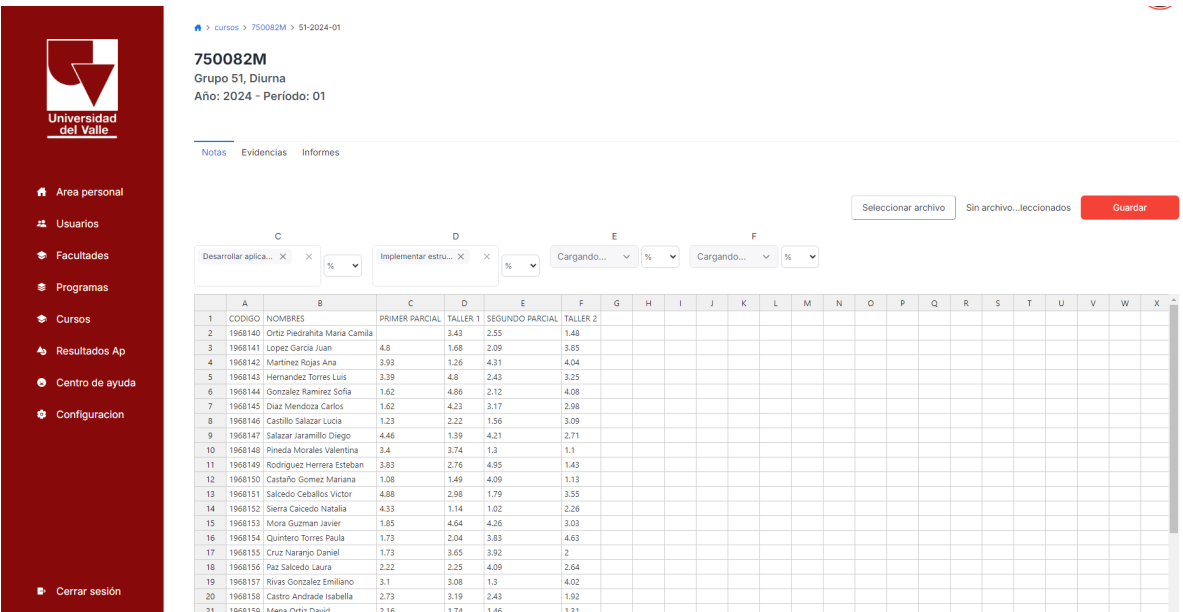


Figura 16: Gestión de un curso, apartado notas



Universidad del Valle

- Area personal
- Usuarios
- Facultades
- Programas
- Cursos
- Resultados Ap
- Centro de ayuda
- Configuracion

Cerrar sesión

HORA: 02:06:09 a. m.
FECHA: sábado, 30 de noviembre de 2024

Austin Santos
Coordinador - Admin

> cursos > 750082M > 51-2024-01

750082M
Grupo 51, Diurna
Año: 2024 - Período: 01

Notas Evidencias Informes

PRIMER PARCIAL

domingo, 13 de octubre de 2024 06:05:05

Para que los estudiantes pierdan

captura.png

Parcial Enunciado.pdf

TALLER EVALUATIVO

domingo, 13 de octubre de 2024 11:31:15

SEGUNDO PARCIAL

domingo, 13 de octubre de 2024 06:05:06

CUARTO PARCIAL

domingo, 13 de octubre de 2024 06:05:37

Figura 17: Gestión de un curso, apartado evidencias



Universidad del Valle

- Area personal
- Usuarios
- Facultades
- Programas
- Cursos
- Resultados Ap
- Centro de ayuda
- Configuracion

Cerrar sesión

Notas Evidencias Informes

Generar Informe Grupal Generar Informe Individual Subir y generar PDF

Informe Grupal de Notas por Resultados de Aprendizaje

Curso: 750082M
Grupo: 51
Año: 2024
Período: 01
Profesor: Juan Gonzalez

Resultados del Informe

Primer Parcial (Promedio: 2.67)

Resultados de Aprendizaje Asociados:RA006

Porcentaje de Evaluación: 10%

Retroalimentación: Los estudiantes presentan dificultades significativas en los resultados de aprendizaje (RA006). Esto indica que la mayoría no ha alcanzado las competencias esperadas en estas áreas. Se recomienda revisar los contenidos relacionados y realizar actividades de refuerzo, para mejorar el entendimiento de los conceptos clave.

Taller 1 (Promedio: 2.99)

Resultados de Aprendizaje Asociados:RA003

Porcentaje de Evaluación: 20%

Retroalimentación: Los estudiantes presentan dificultades significativas en los resultados de aprendizaje (RA003). Esto indica que la mayoría no ha alcanzado las competencias esperadas en estas áreas. Se recomienda revisar los contenidos relacionados y realizar actividades de refuerzo, para mejorar el entendimiento de los conceptos clave.

Figura 18: Gestión de un curso, apartado informes

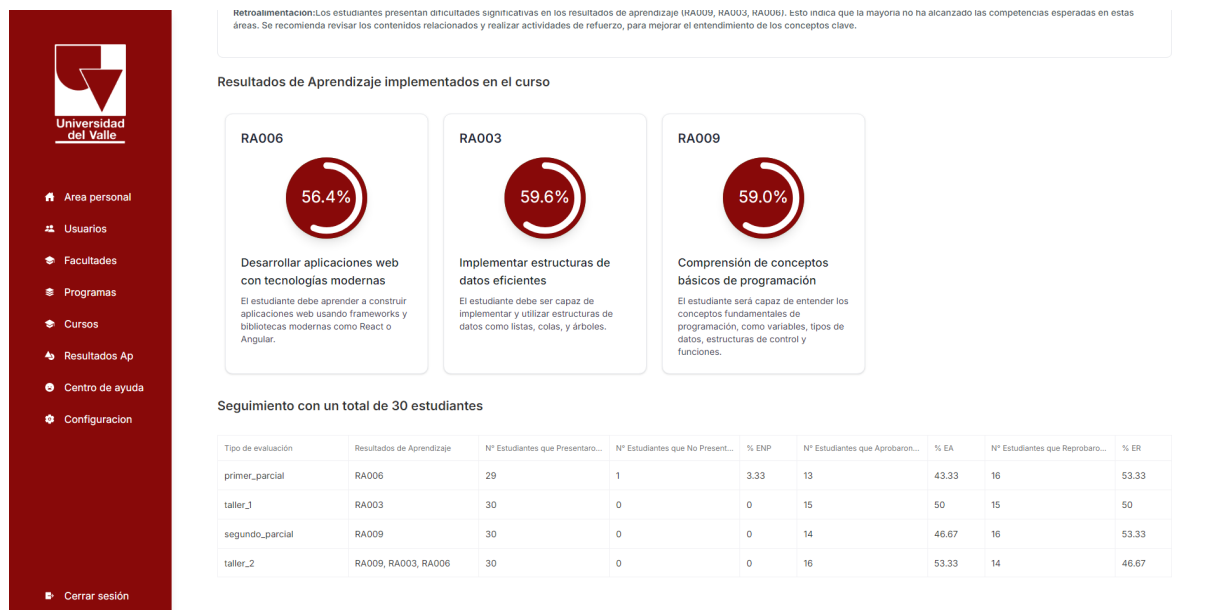


Figura 19: Gestión de un curso, apartado informes

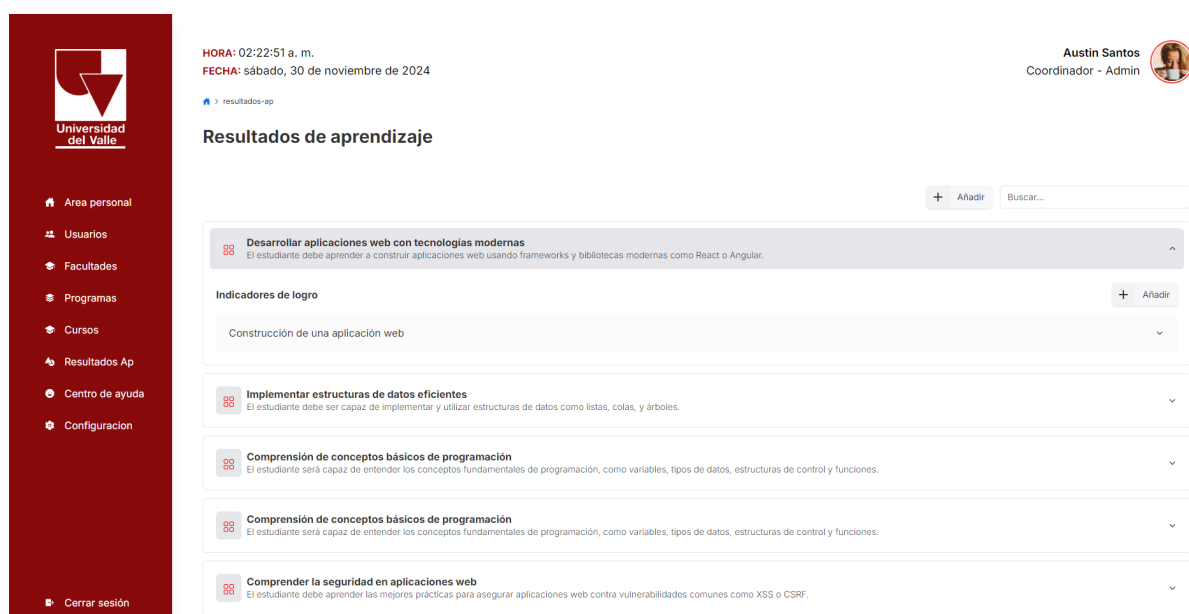


Figura 20: Gestión de los resultados de aprendizaje e indicadores de logro

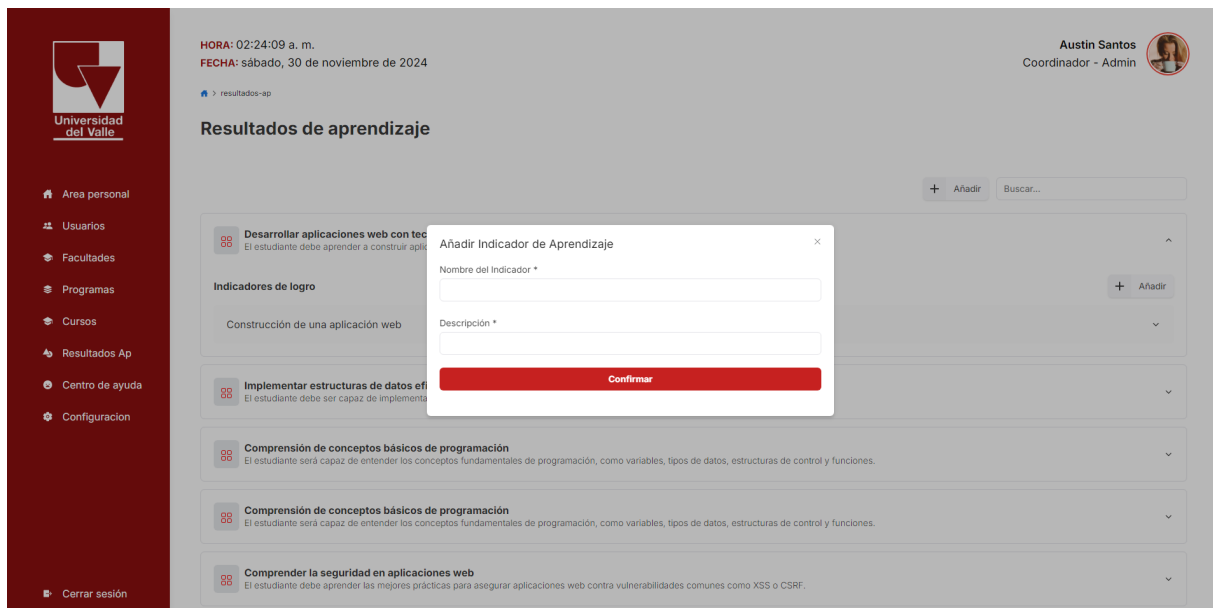


Figura 21: Creación de un indicador de logro

4.10. Resultado de Pruebas

4.10.1. Pruebas Funcionales

Las pruebas fueron hechas con Thunder Client para verificar el correcto funcionamiento de los endpoints de la API. Se probaron las solicitudes GET, POST, PUT y DELETE en diferentes escenarios, validando que los códigos de estado retornados fueran los esperados (como 200 OK o 201 Created) y que el contenido de las respuestas coincidiera con el formato JSON definido. También se realizaron pruebas con datos válidos e inválidos para asegurarse de que la API manejara correctamente errores y validaciones, como entradas incompletas o formatos incorrectos. Además, se verificó el manejo de autenticación mediante tokens y permisos de usuario, confirmando que los endpoints restringidos devolvieran respuestas adecuadas según los roles o credenciales proporcionados.

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Crear un curso con datos válidos	1. Ingresar datos completos. 2. Enviar formulario.	Código: CURS-001, Nombre: Matemáticas, Estado: Activo, Créditos: 3	El curso se guarda correctamente.
2	Intentar crear un curso con datos incompletos	1. Dejar campos obligatorios vacíos. 2. Enviar formulario.	Código vacío, Nombre vacío, Estado: Activo	Muestra error de campos obligatorios.

Cuadro 9: Pruebas para la tabla **Cursos**

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Crear un registro de facultad con datos válidos	1. Ingresar todos los datos requeridos. 2. Enviar formulario.	Decano: Juan Pérez, Email: decano@uni.edu, Nombre: Ingeniería, Teléfono: 123456789, Web: https://uni.edu	El registro se guarda correctamente.
2	Intentar crear un registro con email inválido	1. Ingresar datos con email no válido. 2. Enviar formulario.	Email: decano.uni.edu	Muestra error de validación en el campo email.

Cuadro 10: Pruebas para la tabla **Facultad**

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Crear un grupo con datos válidos	1. Ingresar todos los datos requeridos. 2. Enviar formulario.	Curso: Matemáticas, Profesor: Ana Gómez, Grupo: 1, Jornada: Diurna, Periodo: 01, Año: 2024	El grupo se guarda correctamente.
2	Intentar crear un grupo con un valor inválido en jornada	1. Ingresar una jornada no permitida. 2. Enviar formulario.	Jornada: Vespertina	Muestra error de validación en el campo jornada.

Cuadro 11: Pruebas para la tabla **Grupo**

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Crear un indicador de logro con datos válidos	1. Ingresar todos los campos requeridos. 2. Enviar formulario.	Nombre: Resolución de problemas, Descripción: Lograr resolver problemas matemáticos básicos, Código Resultado Aprendizaje: RA-001	El indicador se guarda correctamente.
2	Intentar crear un indicador sin el código resultado aprendizaje	1. Dejar el campo <code>codigo_resultado_aprendizaje</code> vacío. 2. Enviar formulario.	Nombre: Análisis de texto, Descripción: Desarrollar análisis de texto	Muestra error indicando que el campo es obligatorio.

Cuadro 12: Pruebas para la tabla Indicador Logro

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Crear una nota con datos válidos	1. Ingresar todos los datos requeridos. 2. Enviar formulario.	Curso: Matemáticas, Grupo: 1, Año: 2024, Periodo: 01, Estudiantes: Nombre: Juan Pérez, Código: 12345, Notas: [Parcial (80 %), Final (20 %)]	El registro de notas se guarda correctamente.
2	Intentar registrar una nota con porcentaje superior al 100 %	1. Ingresar un total de porcentajes que supera el 100 %. 2. Enviar formulario.	Notas: [Parcial (110 %)]	Muestra error indicando que el porcentaje no puede superar el 100 %.
3	Validar error al omitir datos de un estudiante	1. Dejar vacío el campo <code>nombre</code> de un estudiante. 2. Enviar formulario.	Estudiantes: [Código: 12345, Notas: [Parcial (80 %)]]	Muestra error indicando que el campo <code>nombre</code> es obligatorio.

Cuadro 13: Pruebas para la tabla Nota

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Crear un programa con datos válidos	1. Ingresar todos los datos requeridos. 2. Enviar formulario.	ID_Coordinador: COORD-001, Nombre Programa: Ingeniería de Software, Código: PROG-001, Facultad: Ingeniería, Duración: 5 años, Periodicidad de Admisiones: Semestral, Modalidad: Presencial, Jornada: Diurna, Créditos: 160, Registro IC-FES: 12345, Registro SNIES: 67890, Resolución MEN: 2024-01, Fecha Decisión Creación: 2020-01-01, Sede: Bogotá, Email: programa@uni.edu	El programa se guarda correctamente.
2	Validar error al registrar un programa con fecha inválida	1. Ingresar una fecha de creación no válida. 2. Enviar formulario.	Fecha Decisión Creación: 01-31-2024 (formato incorrecto)	Muestra error indicando que la fecha es inválida.
3	Intentar registrar un programa con email no válido	1. Ingresar datos con email mal formado. 2. Enviar formulario.	Email: programa.uni.edu	Muestra error indicando que el email es inválido.

Cuadro 14: Pruebas para la tabla **Programa**

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Crear un resultado de aprendizaje con datos válidos	1. Ingresar todos los campos requeridos. 2. Enviar formulario.	Nombre Resultado: Comprender conceptos básicos, Descripción: Lograr que los estudiantes comprendan conceptos introductorios de programación, Código Curso: CURS-001	El resultado se guarda correctamente.
2	Validar error al omitir el código del curso asociado	1. No ingresar el campo <code>codigo_curso</code> . 2. Enviar formulario.	Nombre Resultado: Aplicar metodologías ágiles, Descripción: Integrar Scrum en proyectos de software	Muestra error indicando que el campo <code>codigo_curso</code> es obligatorio.

Cuadro 15: Pruebas para la tabla **Resultado Aprendizaje**

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Registrar una sede con datos válidos	1. Ingresar datos en todos los campos requeridos. 2. Enviar formulario.	PBX: 1234567, Celular: 3001234567, Dirección: Calle 123 45-67, Ubicación: Bogotá	La sede se registra exitosamente en la base de datos.
2	Validar error al omitir el campo direccion	1. Dejar vacío el campo direccion . 2. Enviar formulario.	PBX: 1234567, Celular: 3001234567, Ubicación: Bogotá	Muestra error indicando que el campo direccion es obligatorio.
3	Validar error al ingresar un celular no numérico	1. Ingresar caracteres no numéricos en el campo celular . 2. Enviar formulario.	Celular: 300ABC4567	Muestra error indicando que el celular debe contener solo números.

Cuadro 16: Pruebas para la tabla **Sede**

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Registrar un usuario con datos válidos	1. Ingresar todos los campos requeridos. 2. Enviar formulario.	Código: U-001, Cédula: 123456789, Primer Nombre: Juan, Primer Apellido: Pérez, Celular: 3001234567, Correo: juan.perez@uni.edu, Rol: [Docente], Estado: Activo	El usuario se registra correctamente.
2	Validar error al omitir el campo correo	1. No ingresar un correo en el formulario. 2. Enviar formulario.	Cédula: 123456789, Primer Nombre: Juan, Primer Apellido: Pérez, Rol: [Estudiante], Estado: Activo	Muestra error indicando que el correo es obligatorio.
3	Validar error al ingresar una contraseña menor a 6 caracteres	1. Ingresar una contraseña corta. 2. Enviar formulario.	Contraseña: 12345	Muestra error indicando que la contraseña debe tener al menos 6 caracteres.
4	Validar múltiples roles en el registro del usuario	1. Ingresar varios roles en el formulario. 2. Enviar formulario.	Rol: [Docente, Administrador]	El usuario se registra correctamente con los roles asignados.
5	Validar error al registrar un celular con caracteres no numéricos	1. Ingresar caracteres alfanuméricos en el campo celular . 2. Enviar formulario.	Celular: 300ABC4567	Muestra error indicando que el celular debe contener solo números.

Cuadro 17: Pruebas para la tabla **Usuario**

4.10.2. Pruebas no funcionales

Las pruebas fueron hechas con JMeter para evaluar aspectos no funcionales del prototipo, como rendimiento, escalabilidad y capacidad de respuesta bajo diferentes niveles de carga. Se configuraron escenarios que simulaban múltiples usuarios concurrentes realizando solicitudes simultáneas a los endpoints más críticos, como los de autenticación, consulta de datos y actualización de registros. Durante las pruebas, se midieron métricas clave como tiempos de respuesta, tasas de error, uso de recursos del servidor y capacidad máxima antes de fallos. Además, se realizaron pruebas de estrés para identificar los límites del prototipo y garantizar una buena experiencia del usuario.

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Verificar el tiempo de carga de la página principal	1. Acceder a la página principal del sitio. 2. Medir el tiempo de carga de la página.	N/A	El tiempo de carga debe ser inferior a 3 segundos.

Cuadro 18: Prueba de carga de página (Rendimiento)

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Medir el tiempo de respuesta del inicio de sesión	1. Ingresar credenciales válidas. 2. Presionar el botón de inicio de sesión. 3. Medir el tiempo entre la solicitud y la respuesta exitosa.	Usuario valido y contraseña valida	El tiempo de respuesta es menor a 1 segundo.
2	Medir el tiempo de respuesta al cargar la lista de usuarios	1. Acceder a la página de usuarios. 2. Medir el tiempo desde que se realiza la solicitud hasta que se muestra la lista completa de usuarios.	Usuario autenticado con rol administrador	La lista de usuarios carga en menos de 2 segundos.

Cuadro 19: Pruebas de tiempo de respuesta (Rendimiento)

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
2	Verificar la accesibilidad del menú de navegación	1. Hacer clic en las diferentes opciones del menú. 2. Validar que las secciones cargan correctamente.	Menú: Inicio, Cursos, Contacto	El menú debe ser accesible y las secciones deben cargar sin errores.

Cuadro 20: Prueba de disponibilidad de funcionalidades en el menú (Usabilidad)

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
3	Verificar la respuesta al error de base de datos	1. Desconectar la base de datos. 2. Intentar realizar una operación que dependa de la base de datos.	N/A	El sistema debe mostrar un mensaje de error adecuado y continuar sin caídas críticas.

Cuadro 21: Prueba de respuesta a errores de base de datos (Fiabilidad)

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
4	Verificar la escalabilidad con múltiples usuarios simultáneos	1. Simular múltiples usuarios accediendo al sistema al mismo tiempo. 2. Medir el comportamiento del sistema.	N/A	El sistema debe seguir funcionando correctamente con al menos 20 usuarios simultáneos sin caídas.

Cuadro 22: Prueba de escalabilidad (Rendimiento)

ID	Caso de Prueba	Pasos	Datos de Entrada	Resultado Esperado
1	Bloqueo de intentos de acceso no autorizado	1. Realizar múltiples intentos de inicio de sesión con credenciales incorrectas. 2. Registrar cuántos intentos son bloqueados correctamente.	Usuario invalido y contraseña invalida	El sistema bloquea los 5 intentos de acceso no autorizado.
2	Verificar bloqueo después de múltiples intentos fallidos	1. Intentar iniciar sesión con credenciales incorrectas 10 veces consecutivas.	Usuario invalido y contraseña invalida	El sistema bloquea los 10 intentos de acceso no autorizado

Cuadro 23: Pruebas de tasa de intentos de acceso no autorizado bloqueados (Seguridad)

5. Conclusiones

El desarrollo del prototipo de un sistema para la gestión de la evaluación por resultados de aprendizaje en la Universidad del Valle permitió abordar de manera integral las limitaciones actuales en los procesos académicos, alineándose con las exigencias del Decreto 1330 del Ministerio de Educación Nacional. Esto lleva a las siguientes conclusiones:

- **Análisis exhaustivo de los requisitos.** El análisis detallado de los requisitos permitió identificar y priorizar las necesidades clave de los docentes y coordinadores académicos. Se destacaron funcionalidades esenciales como la trazabilidad de resultados de aprendizaje, la retroalimentación automatizada y la generación de informes personalizados. Estas herramientas no solo fortalecen la transparencia y la eficacia en la gestión académica, sino que también aseguran el cumplimiento de normativas institucionales y regulatorias. Este enfoque sistemático garantiza que el sistema no solo responda a las necesidades actuales, sino que también se anticipe a futuros desafíos en la gestión educativa.
- **Diseño de una plataforma intuitiva y funcional.** La plataforma desarrollada fue diseñada con un enfoque centrado en el usuario, asegurando una experiencia intuitiva y accesible para docentes, coordinadores y demás usuarios. Las interfaces diseñadas no solo permiten un acceso rápido y eficiente a la información clave, sino que también promueven la toma de decisiones basada en datos concretos y oportunos. Además, los mecanismos de retroalimentación inmediata mejoran significativamente la comprensión del progreso académico, optimizando así la capacidad de los docentes para intervenir y ajustar estrategias educativas en tiempo real.
- **Implementación del prototipo.** La implementación del prototipo representó un paso crucial en la materialización de los objetivos del proyecto. Este proceso integró las funcionalidades diseñadas con una arquitectura robusta y escalable basada en tecnologías de vanguardia como Next.js y Firebase. La elección de estas herramientas garantizó un desempeño eficiente incluso en escenarios de alta demanda, reduciendo tiempos de carga y aumentando la confiabilidad del sistema. Asimismo, se priorizó la seguridad de los datos mediante la implementación de autenticación avanzada y estrictos mecanismos de validación, lo que asegura la confidencialidad e integridad de la información sensible.
- **Pruebas funcionales y retroalimentación de usuarios.** Las pruebas funcionales realizadas validaron que el sistema cumple con los estándares de usabilidad, rendimiento y confiabilidad requeridos. Los usuarios finales, incluidos docentes y coordinadores, destacaron la utilidad y eficacia de las herramientas para la generación de informes, el registro y la visualización de resultados de aprendizaje. Además, subrayaron la flexibilidad del sistema para adaptarse a las diversas necesidades institucionales, posicionándolo como una solución altamente adaptable y relevante en contextos académicos dinámicos.
- **Impacto del proyecto en la Universidad del Valle.** Este proyecto marca un hito significativo para la Universidad del Valle, al proporcionar una herramienta que facilita el cumplimiento normativo y promueve un modelo educativo centrado en el aprendizaje y la mejora continua. Con esta plataforma, los docentes cuentan con un sistema eficiente para gestionar indicadores clave, evaluar el progreso académico y

optimizar la experiencia educativa de manera integral. En conjunto, el proyecto no solo mejora la gestión académica, sino que también refuerza el compromiso de la institución con la excelencia educativa y la innovación tecnológica.

5.1. Reflexión sobre el impacto y limitaciones del prototipo

El desarrollo de este sistema tiene el potencial de transformar significativamente el entorno académico al ofrecer una herramienta que facilita la gestión y el seguimiento del aprendizaje. La posibilidad de generar informes personalizados y brindar retroalimentación automatizada permite a los docentes tomar decisiones más informadas, mejorando la calidad de la educación y promoviendo un aprendizaje más efectivo. Además, la trazabilidad de los resultados de aprendizaje puede ser utilizada por la administración académica para identificar áreas de mejora, diseñar intervenciones específicas y asegurar el cumplimiento de estándares educativos.

Impactos notables

- **Reducción en la tasa de reprobación:** Los informes generados por el sistema permiten identificar de manera temprana a estudiantes con bajo rendimiento. Esto facilita la implementación de estrategias de apoyo, como tutorías adicionales, lo que disminuye las tasas de reprobación.
- **Estandarización de evaluaciones:** El prototipo asegura que todos los profesores trabajen bajo un mismo esquema de indicadores de logro y criterios de evaluación. Esto mejora la coherencia en los resultados de aprendizaje y garantiza una mayor equidad.
- **Toma de decisiones basadas en datos:** Los coordinadores y directivos pueden usar los reportes detallados para identificar tendencias, reasignar recursos o ajustar contenidos de los cursos según las necesidades detectadas.

Limitaciones identificadas

El prototipo presenta algunas limitaciones que deben ser abordadas en futuras iteraciones. Una de las principales limitaciones es la dependencia de tecnologías externas como Firebase, lo que podría generar problemas de escalabilidad a largo plazo si el volumen de datos crece exponencialmente. Asimismo, aunque se implementaron medidas de seguridad avanzadas, el sistema podría requerir auditorías periódicas y actualizaciones constantes para garantizar la protección de la información sensible.

Otra limitación relevante es que, si bien las pruebas funcionales mostraron resultados positivos, estas se realizaron en un entorno controlado y con un grupo reducido de usuarios. Será necesario realizar pruebas en un entorno real, con un mayor número de participantes y escenarios variados, para validar completamente el desempeño del sistema y su adaptabilidad a diferentes contextos educativos.

5.2. Futuras Líneas de Trabajo

Ampliación de Funcionalidades: Incorporar módulos de analítica predictiva y recomendaciones personalizadas para potenciar la toma de decisiones académicas. Estos módulos permitirán anticipar patrones de desempeño de los estudiantes y generar sugerencias proactivas sobre planes de mejora, asignación de recursos y estrategias pedagógicas. Además, se planea integrar un sistema de alerta temprana que notifique a los docentes sobre estudiantes en riesgo de bajo desempeño o deserción.

Escalabilidad: Adaptar el prototipo para su implementación en otros contextos educativos y niveles académicos. Esto incluye la posibilidad de ofrecer el sistema como una solución SaaS (Software as a Service) para instituciones educativas de diferentes tamaños, permitiendo configuraciones personalizadas según las necesidades específicas de cada institución.

Retroalimentación Continua: Promover ciclos regulares de evaluación y ajuste del sistema con base en el feedback de sus usuarios. Para ello, se implementará un módulo de encuestas automáticas que recoja la opinión de los usuarios finales de manera periódica y permita identificar áreas de mejora rápidamente.

Seguridad y Privacidad: Reforzar los módulos de seguridad del sistema mediante auditorías de código y pruebas de penetración regulares. Se busca implementar el cumplimiento de normativas internacionales de protección de datos como el GDPR (General Data Protection Regulation) para asegurar que la información sensible de los estudiantes esté siempre protegida.

Interoperabilidad: Desarrollar APIs que permitan la integración con otros sistemas de gestión educativa y plataformas de aprendizaje ya existentes, facilitando un ecosistema digital más cohesionado y eficiente.

En conclusión, este trabajo no solo representa una solución puntual, sino que también abre la puerta a nuevas oportunidades para transformar los procesos educativos en la era digital, contribuyendo al desarrollo académico integral de los estudiantes y al cumplimiento de estándares de calidad educativa.

Referencias

- [1] M. de Educación Nacional, “Decreto 1330 del 25 de julio de 2019,” *Mineducación*, jul 2019, por el cual se sustituye el Capítulo 2 y se suprime el Capítulo 7 del Título 3 de la Parte 5 del Libro 2 del Decreto 1075 de 2015 -Único Reglamentario del Sector Educación.
- [2] Moodle. [26/12/2022]. [Online]. Available: https://docs.moodle.org/all/es/Acerca_de_Moodle
- [3] Watermark. [20/10/2023]. [Online]. Available: <https://www.eduappcenter.com/apps/97>
- [4] Blackboard learn. [2023]. [Online]. Available: <https://www.anthology.com/products/teaching-and-learning/learning-effectiveness/blackboard-learn>
- [5] Cómo consultar los resultados de aprendizaje sena. [2021]. [Online]. Available: <https://senasofiaplusedu.com.co/como-consultar-los-resultados-de-aprendizaje-sena/>
- [6] J. L. G. Álvarez, M. M. Cruz, and J. M. Salvatierra, “La evaluación educativa como proceso histórico social. perspectivas para el mejoramiento de la calidad de los sistemas educativos,” 2017. [Online]. Available: <https://tuatara.co/blog/software/mvp/#:~:text=Un%20producto%20m%C3%ADnimo%20viable%20o,superior%20a%20su%20inversi%C3%B3n%20inicial>
- [7] R. W. Tyler, “Ralph w. tyler - father of educational evaluation and assessment,” 1994.
- [8] O. E. B. Gutiérrez, “Estrategias metodológicas y procedimentales de evaluación que utilizan los profesores del componente docente del p.e.b.i. de la ula-táchira,” Ph.D. dissertation, Universidad de Los Andes-Universitat Rovira i Virgili, 1995.
- [9] S. A. A. Lara, N. P. L. L, and B. G. Valero, “Modelos y épocas de la evaluación educativa,” *Educere*, vol. 23, no. 75, pp. 307–322, 2019.
- [10] Wikipedia, “Learning management system (lms),” [Online]. Available: https://en.wikipedia.org/wiki/Learning_management_system, 2023.
- [11] M. Coppola. (2023) Frontend y backend: qué son, en qué se diferencian y ejemplos. Accedido el 27 de noviembre de 2023. [Online]. Available: <https://blog.hubspot.es/website/frontend-y-backend>
- [12] Open source on aws. [Fecha de acceso: 20/10/2023]. [Online]. Available: <https://aws.amazon.com/es/what-is/open-source>
- [13] M. Blog. (2015) ¿qué es el e-portafolio? 4 junio, 2015. [Online]. Available: <https://mychela.wordpress.com/2015/06/04/que-es-el-e-portafolio/>
- [14] Multilingüe. Definición de la Real Academia Española. [Fecha de acceso: 20/10/2023]. [Online]. Available: <https://www.rae.es/dle/>
- [15] DinaHosting. (2021) ¿qué es un plugin y para qué sirve? 13 julio, 2021. [Online]. Available: <https://dinahosting.com/blog/que-es-un-plugin-y-para-que-sirve/>

- [16] muytecnologicos.com. Investigación tecnológica - qué es, definición y concepto. <https://muytecnologicos.com/diccionario-tecnologico/investigacion-tecnologica>. Accedido: 2023-11-27.
- [17] K. Cevallos. (2017) Metodología de desarrollo Ágil: Xp y scrum. [Online]. Available: <https://ingsoftwarekarlacevallos.wordpress.com/2015/05/08/metodologia-de-desarrollo-agil-xp-y-scrum/>.
- [18] <https://zup.com.br/blog/git-workflow>. Git workflow: o que é e principais tipos. [Online]. Available: <https://zup.com.br/blog/git-workflow>
- [19] M. de Educación Nacional, “Resolución 021795 de 19 de noviembre de 2020,” *Mineducación*, nov 2020, por la cual se establecen los parámetros de autoevaluación, verificación y evaluación de las condiciones de calidad de programa reglamentadas en el Decreto 1075 de 2015, modificado por el Decreto 1330 de 2019, para la obtención, modificación y renovación del registro calificado.
- [20] T. J. Fontalvo, E. J. Delahoz-Dominguez, and G. De la Hoz, “Resultados de aprendizaje y mecanismos de evaluación en los programas académicos de educación superior en colombia,” *Formación universitaria*, vol. 15, no. 1, pp. 105–114, 2022.
- [21] J. P. Royo *et al.*, “Un modelo de guía docente desde los resultados de aprendizaje y su evaluación,” *Zaragoza: Universidad de Zaragoza*, 2010.
- [22] J. E. A. Salazar. Metodología para el desarrollo de aplicaciones web ajustada al contexto de los productos mínimos viables. <https://www.redalyc.org/articulo.oa?id=509552658004>. Accedido: 2023-11-28.
- [23] tuatara.co. (n.d.) Producto mínimo viable (mvp). [Último acceso: 28 noviembre 2023]. [Online]. Available: <https://tuatara.co/blog/software/mvp/#:~:text=Un%20producto%20m%C3%ADnimo%20viable%20o,superior%20a%20su%20inversi%C3%B3n%20inicial>.
- [24] “Cambio climático: El ser humano ha calentado el planeta a un nivel nunca visto en los últimos 2000 años,” <https://news.un.org/es/story/2021/08/1494142>, 2021, accedido: 2023-12-06.
- [25] “Emisiones de dióxido de carbono,” <https://espanol.epa.gov/>, accedido: 2023-12-06.
- [26] Google. Firebase documentation. [Online]. Available: <https://firebase.google.com/docs>
- [27] NextAuth.js. Nextauth.js documentation. [Online]. Available: <https://next-auth.js.org/>
- [28] Nodemailer. Nodemailer documentation. [Online]. Available: <https://nodemailer.com/>
- [29] Zod. Zod documentation. [Online]. Available: <https://zod.dev/>
- [30] Google. Firebase authentication documentation. [Online]. Available: <https://firebase.google.com/docs/auth>
- [31] React. React documentation. [Online]. Available: <https://reactjs.org/docs/getting-started.html>

- [32] Axios. Axios documentation. [Online]. Available: <https://axios-http.com/docs/intro>
- [33] R. Icons. React icons documentation. [Online]. Available: <https://react-icons.github.io/react-icons/>
- [34] T. CSS. Tailwind css documentation. [Online]. Available: <https://tailwindcss.com/docs>
- [35] ESLint. Eslint documentation. [Online]. Available: <https://eslint.org/docs>
- [36] R. Scheduler. React scheduler documentation. [Online]. Available: <https://www.npmjs.com/package/@aldabil/react-scheduler>

6. Anexos

Anexo A, requerimientos funcionales

Presione para ir a los Requerimientos de Software.

Anexo B, diseño preliminar

Presione para ir diseño premiliminar en Figma.

Anexo C, repositorio

Presione para ir al repositorio de Github.

Anexo D, modelo entidad relación

Presione para ver el modelo entidad relación.

Anexo E, Figuras prototipo

Presione para ir al Drive donde están todas las figuras del prototipo.

Anexo F, Encuesta prototipo

Presione para ir al Drive donde está la encuesta del prototipo.