



Uleam
UNIVERSIDAD LAICA
ELOY ALFARO DE MANABÍ

UNIVERSIDAD LAICA ELOY ALFARO DE MANABÍ
FACULTAD DE CIENCIAS DE LA VIDA Y TECNOLOGÍAS
CARRERA DE TECNOLOGÍA DE LA INFORMACIÓN

MODALIDAD DE TITULACIÓN:

PROYECTO INTEGRADOR

TEMA:

**PLATAFORMA WEB EDUCATIVA DE CURSOS MULTIPROFESIONALES PARA
LA ORGANIZACIÓN Y GESTIÓN DE TALLERES PERSONALIZADOS**

AUTOR:

BRAVO ANDRADE JONATAN NATANAEL

TUTOR:

ING. ROBERT WILFRIDO MOREIRA CENTENO

2025(1)

DECLARACIÓN DE AUTORÍA.

Yo, Jonatan Natanael Bravo Andrade con cédula de ciudadanía 1351762081; en calidad de autor del trabajo de titulación “PLATAFORMA WEB EDUCATIVA DE CURSOS MULTIPROFESIONALES PARA LA ORGANIZACIÓN Y GESTIÓN DE TALLERES PERSONALIZADOS.”, autorizo a la Universidad Laica Eloy Alfaro de Manabí (ULEAM), hacer uso total o parcial de este trabajo de titulación del que soy responsable, con fines estrictamente académicos o investigativos.

Lo declara,



Jonatan Nanatael Bravo Andrade

Cédula: 1351762081

Correo: e1351762081@live.uleam.edu.ec

	NOMBRE DEL DOCUMENTO:	CÓDIGO: PAT-04-F-004
	CERTIFICADO DE TUTOR(A).	
	PROCEDIMIENTO: TITULACIÓN DE ESTUDIANTES DE GRADO BAJO LA UNIDAD DE INTEGRACIÓN CURRICULAR	REVISIÓN: 1 Página 1 de 1

CERTIFICACIÓN

En calidad de docente tutor(a) de la Facultad de Ciencias de la Vida y Tecnologías de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

Haber dirigido, revisado y aprobado preliminarmente el Trabajo de Integración Curricular bajo la autoría del estudiante BRAVO ANDRADE JONATAN NATANAEL, legalmente matriculado/a en la carrera de Ingeniería en Tecnologías de la Información, período académico 2025-2025(1), cumpliendo el total de 400 horas, cuyo tema del proyecto es "PLATAFORMA WEB EDUCATIVA DE CURSOS MULTIPROFESIONALES PARA LA ORGANIZACIÓN Y GESTIÓN DE TALLERES PERSONALIZADOS".

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concordancia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad del mismo, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

Manta, 20 de octubre de 2025.

Lo certifico,



Ing. Robert Moreira Centeno
Docente Tutor
Área: Software

Nota 1: Este documento debe ser realizado únicamente por el/la docente tutor/a y será receptado sin enmendaduras y con firma física original.

Nota 2: Este es un formato que se llenará por cada estudiante (de forma individual) y será otorgado cuando el informe de similitud sea favorable y además las fases de la Unidad de Integración Curricular estén aprobadas.

AGRADECIMIENTOS.

Agradezco a toda mi familia y amigos que me apoyaron incondicionalmente.

Y agradezco mis docentes que fueron la escalera de conocimiento durante este largo pero hermoso camino.

DEDICATORIA.

Dedico todo mi esfuerzo a mi madre que fue la que me impulsó y apoyó en todo este camino, a mis hermanas por ser pilar fundamental y a mis seres queridos por su granito de arena para este logro.

RESUMEN

La presente investigación desarrolla una plataforma web educativa orientada a la gestión de cursos multiprofesionales y talleres personalizados, con el objetivo de optimizar los procesos académicos y administrativos mediante la integración de herramientas tecnológicas modernas. El sistema propuesto fue implementado utilizando **Angular 18** para el frontend y **Supabase (PostgreSQL + RLS)** para el backend, garantizando una arquitectura segura, escalable y eficiente.

La metodología de desarrollo empleada fue **Programación Extrema (XP)** en su modalidad individual, lo que permitió ciclos cortos de desarrollo, entregas incrementales y pruebas continuas, asegurando la calidad del software y su alineación con los requerimientos definidos.

Entre las principales funcionalidades del sistema se incluyen la gestión de usuarios y roles, creación de cursos con módulos y videos, control de pagos mediante comprobantes, seguimiento del progreso estudiantil, generación automática de certificados en PDF y reportes personalizados para cada rol.

Los resultados obtenidos evidencian una mejora significativa en la automatización de procesos y en la experiencia del usuario frente a plataformas tradicionales como Moodle o Udemy. La validación de la propuesta demostró tiempos de carga inferiores a 2 segundos, correcta aplicación de políticas de seguridad RLS y una alta valoración de usabilidad.

En conclusión, la plataforma desarrollada constituye una solución tecnológica adaptable y de bajo costo para instituciones educativas y organizaciones que buscan fortalecer la educación virtual mediante herramientas locales, seguras y personalizables.

Palabras clave: educación virtual, LMS, Supabase, Angular, XP, talleres personalizados, RLS, certificación digital.

ABSTRACT

This research presents the development of a web-based educational platform designed for the management of multiprofessional courses and personalized workshops, aiming to optimize academic and administrative processes through the integration of modern technologies. The proposed system was implemented using **Angular 18** for the frontend and **Supabase (PostgreSQL + RLS)** for the backend, ensuring a secure, scalable, and efficient architecture.

The development methodology applied was **Extreme Programming (XP)** in an individual context, which enabled short development cycles, incremental deliveries, and continuous testing to ensure software quality and alignment with functional requirements.

Key system features include user and role management, course creation with modules and videos, payment validation through uploaded receipts, progress tracking, automatic generation of PDF certificates, and customized reporting by user role.

The results demonstrate significant improvements in automation and user experience compared to traditional platforms such as Moodle or Udemy. The validation process showed page load times under 2 seconds, correct enforcement of RLS security policies, and high usability ratings.

In conclusion, the developed platform represents a flexible, low-cost technological solution for educational institutions and organizations seeking to enhance virtual learning through secure, locally adaptable tools.

Keywords: e-learning, LMS, Supabase, Angular, XP, personalized workshops, RLS, digital certification.

Índice

Índice4

Índice de figuras92

Índice de tablas103

CAPÍTULO I: INTRODUCCIÓN114

1.1. Introducción114

1.2. Presentación del tema11

1.3. Ubicación y contextualización de la problemática12

1.4. Planteamiento del problema12

1.4.1. Problematización12

1.4.2. Génesis del problema13

1.4.3. Estado actual del problema.13

1.5. Diagrama causa – efecto del problema14

1.6. Objetivos14

1.6.1. Objetivo General.14

1.6.2. Objetivos específicos14

1.7. Justificación15

1.8. Impactos esperados15

1.8.1. Impacto tecnológico.15

1.8.2. Impacto social.16

1.8.3. Impacto ecológico.16

CAPÍTULO II: MARCO TEÓRICO DE LA INVESTIGACIÓN17

2.1. Antecedentes históricos.17

2.2. Antecedentes de Investigación Relacionados al Tema18

2.2.1. Moodle19

2.2.2. Platzi20

2.2.3. Udemy21

2.2.4. Comparación de Platzi y Udemy21

2.2.5. Supabase 21

2.2.6. Angular21

2.2.7. Digital Ocean (droplet)21

2.2.8. PostgreSQL21

2.2.8.1. RLS21

2.2.9. Metodología de desarrollo XP21

2.3. Definiciones Conceptuales26

2.4 Conclusiones relacionadas al marco teórico26

CAPITULO III: METODOLOGIA DE INVESTIGACION29

3.1. Enfoque de la investigación29

3.2. Tipo de investigación29

3.3. Diseño de investigación29

3.4. Población y muestra30

3.5 Técnicas de recolección de información30

3.6 Instrumentos31

3.6.1. Matrices comparativas31

3.6.2. Registro de pruebas prácticas32

3.6.3. Análisis categorial33

3.7 Procedimiento34

CAPITULO IV: MARCO PROPOSITIVO35

4.1. Propuesta del proyecto35

4.1.1 Descripción general de la propuesta359

4.1.2. Justificación técnica de la propuesta35

4.1.3. Viabilidad técnica, económica y operativa36

4.1.3.1. Viabilidad técnica.36

4.1.3.2. Viabilidad económica36

4.1.3.3. Viabilidad operativa37

4.1.4. Desarrollo de la propuesta37

4.1.4.1 Etapas de desarrollo37

4.1.4.2 Recursos utilizados39

4.1.4.3. Cronograma de actividades43

4.2. Requerimientos Funcionales43

- a) Gestión de usuarios y roles43
- b) Gestión de cursos y contenidos44
- c) Gestión de inscripciones y pagos44
- d) Gestión del progreso y certificación45
- e) Reportes y métricas45

4.3. Requerimientos No Funcionales48

- a) Seguridad48
- b) Rendimiento y disponibilidad48
- c) Usabilidad48
- d) Escalabilidad y mantenimiento49
- e) Portabilidad49

4.4. Diagrama ERD50

4.5. Interfaces51

- 4.5.1. Inicio51
- 4.5.2. Inicio de sesión51
- 4.5.3. Cursos52
- 4.5.4. Detalle de curso52
- 4.5.5. Compra de curso53
- 4.5.6. Instructores53
- 4.5.7. Aplicar para instructor54

4.5.8. Dashboard de estudiante54

4.5.9. Dashboard de instructor55

4.5.10. Cursos de instructor55

4.5.11. Estudiantes inscritos de instructor56

4.5.12. Ventas de instructor56

4.5.13. Creación de curso57

4.5.14. Estudiantes58

4.5.15. Instructores59

4.5.16. Aplicaciones para instructor59

4.5.17. Detalle de aplicación para instructor60

4.5.18. Cursos60

4.5.19. Pagos61

4.6. Validación de la Propuesta61

4.6.1. Métodos de validación empleados62

4.6.2. Resultados de la validación62

4.6.3 Ajustes y mejoras posteriores63

Conclusiones65

Recomendaciones66

Bibliografía66

Índice de figuras

Ilustración 1 Diagrama causa-efecto. Fuente: Propia 9

Ilustración 2 Interfaz de Moodle. Fuente: <https://aulavirtualmoodle.uileam.edu.ec> 20

Ilustración 3 Pantalla principal de Platzi. Fuente: <https://platzi.com> 19

Ilustración 4 Pantalla principal de Udemy. Fuente: <https://udemy.com> 20

Ilustración 5 Diagrama de validación de la propuesta LMS: Fuente: propia; **Error!**

Marcador no definido.

Ilustración 6 Diagrama ERD del esquema trabajado (Public). Fuente: propia 50

Ilustración 7 Interfaz de inicio. Fuente: propia 51

Ilustración 8 Interfaz de inicio de sesión. Fuente: propia 52

Ilustración 9 Interfaz de cursos. Fuente: propia 52

Ilustración 10 Interfaz de detalle de curso. Fuente: propia 53

Ilustración 11 Interfaz de detalle de curso. Fuente: Propia 53

Ilustración 12 Interfaz de instructores. Fuente: propia; **Error! Marcador no definido.**

Ilustración 13 Interfaz de aplicación para instructor. Fuente: propia 54

Ilustración 14 Interfaz de perfil del estudiante. Fuente: propia 54

Ilustración 15 Interfaz de perfil del instructor. Fuente: propia 55

Ilustración 16 Interfaz de cursos del instructor. Fuente: propia 55

Ilustración 17 Interfaz de estudiantes inscritos del instructor. Fuente: propia 56

Ilustración 18 Interfaz de ventas del instructor. Fuente: propia 56

Ilustración 19 Interfaz de creación de curso 1. Fuente: propia 57

Ilustración 20 Interfaz de creación de curso 2. Fuente: propia57

Ilustración 21 Interfaz de creación de curso 3. Fuente: propia58

Ilustración 22 Interfaz de estudiantes. Fuente: propia58

Ilustración 23 Interfaz de instructores. Fuente: propia59

Ilustración 24 Interfaz de aplicaciones de instructores. Fuente: propia59

Ilustración 25 Interfaz de detalle de aplicación para instructor. Fuente: propia60

Ilustración 26 Interfaz de cursos. Fuente: propia60

Ilustración 27 Interfaz de pagos. Fuente: propia61

Índice de tablas

Tabla 1 Comparativa de LMS. Fuente: propia; **Error! Marcador no definido.**

Tabla 2 Matriz comparativa de LMS. Fuente: propia31

Tabla 3 Comparativa de prebas práctica sde LMS. Fuente: Propia33

Tabla 4 Matriz de aportes. Fuente: propia33

Tabla 5 Cronograma de desarrollo. Fuente: propia435

CAPÍTULO I: INTRODUCCIÓN

1.1.Introducción

En el contexto actual, la educación y la capacitación multiprofesional se han convertido en elementos clave para el desarrollo personal y profesional. La tecnología ha permitido la creación de plataformas educativas en línea que facilitan el acceso al conocimiento desde cualquier lugar y en cualquier momento. Sin embargo, muchas de estas plataformas no ofrecen la flexibilidad y personalización necesarias para satisfacer las necesidades específicas de los estudiantes y profesionales. Este proyecto propone el desarrollo de una plataforma web educativa que permita la organización y gestión de talleres personalizados, proporcionando una experiencia educativa enriquecida y adaptada a cada usuario.

El objetivo principal de esta plataforma es mejorar la calidad de la educación y aumentar la satisfacción del usuario mediante la personalización de los contenidos educativos. Se espera que esta personalización no solo incremente la efectividad del aprendizaje, sino que también promueva una mayor participación y compromiso de los estudiantes.

1.2.Presentación del tema

El tema del proyecto se base en el desarrollo de una plataforma web educativa de cursos multiprofesionales para la organización y gestión de talleres personalizados. Esta plataforma se diseñará para satisfacer las necesidades específicas de los usuarios, permitiendo la creación de talleres personalizados que se adapten a sus intereses y objetivos educativos.

La plataforma ofrecerá una variedad de herramientas y funcionalidades que facilitarán la organización y gestión de talleres, incluyendo la posibilidad de crear y personalizar cursos, gestionar recursos educativos, y evaluar el progreso de los estudiantes. Además, se implementarán tecnologías avanzadas para mejorar la experiencia del usuario y asegurar la efectividad del aprendizaje.

1.3. Ubicación y contextualización de la problemática

En el Ecuador, las instituciones educativas han venido incorporando de manera progresiva entornos virtuales de aprendizaje como parte de sus procesos de innovación académica. Sin embargo, persisten limitaciones relacionadas con la personalización del aprendizaje, la trazabilidad del progreso estudiantil y la integración de procesos administrativos dentro de estas plataformas.

A nivel regional, en la provincia de Manabí, el acceso desigual a recursos tecnológicos y la dependencia de plataformas externas como Moodle o Google Classroom han evidenciado la necesidad de contar con un sistema de gestión de aprendizaje (LMS) propio, que responda a las particularidades locales. Estas limitaciones se hacen más evidentes en los componentes prácticos, donde los docentes requieren un control más preciso de actividades, evaluaciones y entregables, mientras que los estudiantes solicitan herramientas que faciliten el aprendizaje autónomo y el acceso flexible a contenidos.

1.4. Planteamiento del problema

1.4.1. Problematicación

En el ámbito de la educación en el Ecuador, el uso de plataformas de aprendizaje virtual ha crecido de manera acelerada; sin embargo, persisten problemas concretos. La tasa de

finalización de cursos en línea es baja, los estudiantes no cuentan con un seguimiento preciso de su progreso en actividades y videos, y la validación de pagos aún depende de procesos manuales que generan demoras y errores.

1.4.2. Génesis del problema

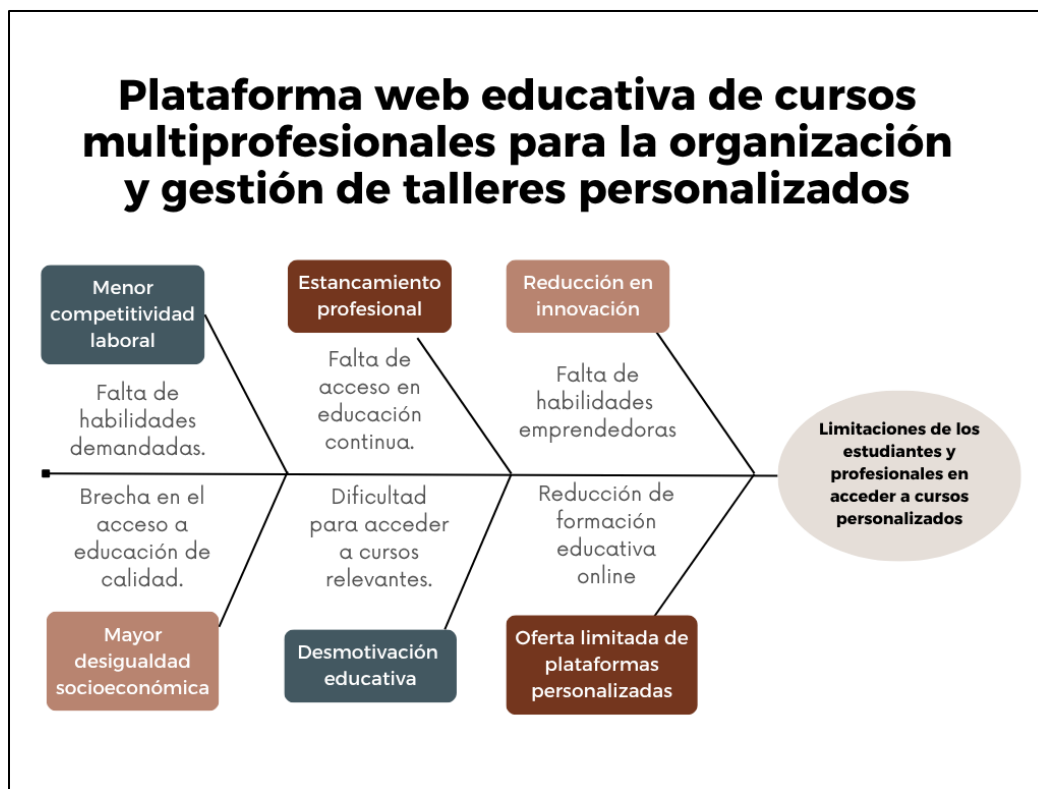
La problemática tiene su origen en la adopción de plataformas genéricas como Moodle o Google Classroom durante la pandemia, cuando las instituciones buscaron soluciones rápidas para garantizar la continuidad académica. Estas herramientas resolvieron la urgencia inicial, pero no fueron diseñadas para atender necesidades locales como la integración de pagos con comprobantes bancarios, la diferenciación clara de roles (administrador, instructor, estudiante) o la emisión de certificados digitales con validez institucional.

Indicador: En instituciones educativas más del 70 % de los pagos de cursos extracurriculares siguen gestionándose de forma manual, sin trazabilidad ni integración con la plataforma académica.

1.4.3. Estado actual del problema.

En la actualidad las instituciones de educación continúan dependiendo de LMS comerciales o desarrollos internos básicos. Aunque permiten alojar contenidos, estos sistemas no integran adecuadamente procesos críticos como el seguimiento automático del avance en videos, la certificación digital ni la validación inmediata de pagos. Además, los LMS comerciales suelen implicar altos costos de licencia y baja adaptabilidad a contextos regionales.

Indicador: las encuestas internas reflejan un nivel de satisfacción inferior al 60 % en usabilidad y reportes académicos, y el Net Promoter Score (NPS) de las plataformas actuales es negativo, lo que evidencia la necesidad de un sistema propio y más eficiente.



1.5. Diagrama causa – efecto del problema

Ilustración 1 Diagrama causa-efecto. Fuente: Propia

1.6. Objetivos

1.6.1. Objetivo General.

Diseñar un sistema de gestión de aprendizaje que integre procesos académicos y administrativos, orientado a mejorar la experiencia de estudiantes e instructores mediante una plataforma de cursos en línea.

1.6.2. Objetivos específicos

1. Desarrollar un módulo de gestión para que el administrador pueda manejar la plataforma dentro los submódulos de cursos, estudiantes, comprar y solicitudes.
2. Implementar la lógica de automatización de reseñas de instructores mediante la calificación de sus cursos.
3. Agregar la funcionalidad donde el instructor pueda crear cursos personalizados mediante diferentes tipos de elementos por módulos.
- 4.Registrar el progreso del estudiante en los cursos para generar automáticamente un certificado al completarlos

1.7. Justificación

La presente investigación se justifica en la necesidad de fortalecer los procesos de enseñanza-aprendizaje en instituciones, mediante la implementación de un sistema de gestión

de aprendizaje (LMS) que integre de forma eficiente la administración académica y los procesos administrativos vinculados a pagos, matrículas y certificaciones digitales.

1.8.Impactos esperados

1.8.1. Impacto tecnológico.

El desarrollo de la plataforma contribuirá al avance tecnológico en el campo de la educación en línea, ofreciendo nuevas herramientas y funcionalidades que mejorarán la experiencia de aprendizaje y gestión de cursos personalizados.

1.8.2. Impacto social.

Facilitar el acceso a la educación continua y la capacitación multiprofesional tendrá un impacto positivo en la sociedad, permitiendo a más personas mejorar sus habilidades y conocimientos, teniendo impacto en su calidad de vida y oportunidades laborales. La plataforma permitirá a los estudiantes de diversas procedencias acceder a una educación de alta calidad, independientemente de su ubicación geográfica o situación económica.

1.8.3. Impacto ecológico.

La implementación de esta plataforma reducirá la necesidad de transporte físico hacia centros de educativos, lo cual contribuirá a la disminución de la huella de carbono. Además, al digitalizar materiales de estudio, se reducirá el uso de papel y otros recursos físicos, promoviendo una educación más sostenible y respetuosa con el medio ambiente.

Este contenido ampliado ofrece una visión detallada del proyecto de tesis, destacando la importancia de la personalización en la educación en línea y los beneficios esperados de la plataforma educativa propuesta.

CAPÍTULO II: MARCO TEÓRICO DE LA INVESTIGACIÓN

2.1. Antecedentes históricos.

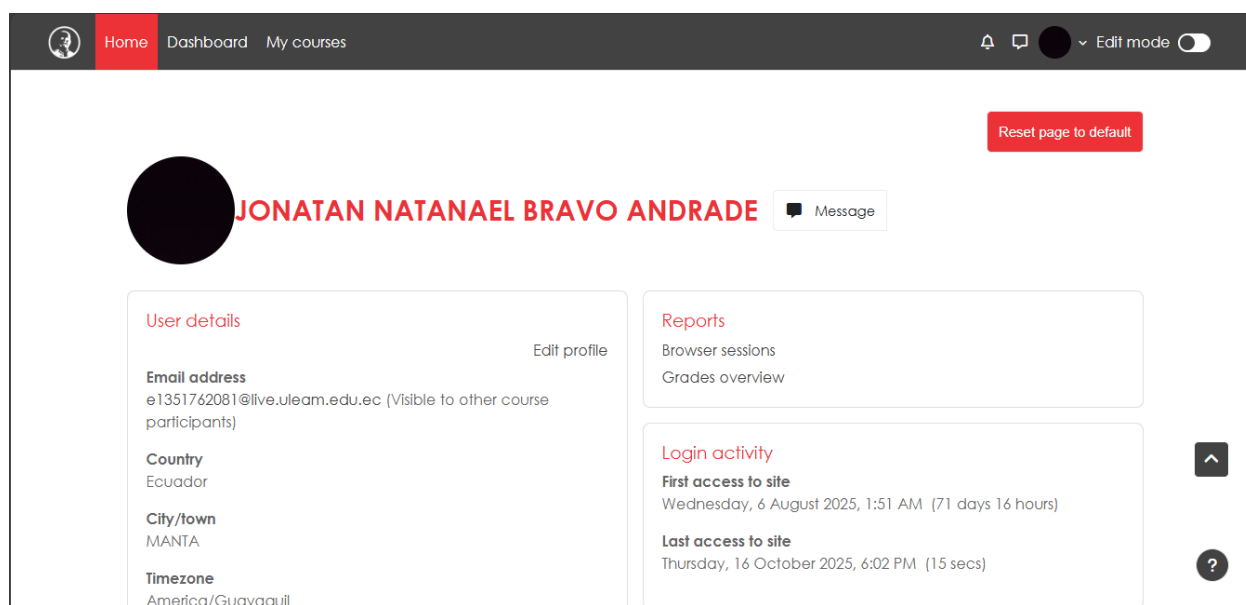
La educación a distancia ha experimentado una evolución constante desde sus orígenes por correspondencia en el siglo XIX hasta los actuales ecosistemas digitales. En Iberoamérica, autores como Aretio (RIED, UNED) destacan cómo esta modalidad pasó de la enseñanza por medios impresos y radiales a las plataformas virtuales de aprendizaje (LMS), con un énfasis creciente en la interacción y la construcción colectiva del conocimiento.

En la región, organismos como la OEI y la CEPAL han documentado que la pandemia de COVID-19 significó un punto de inflexión: millones de estudiantes migraron hacia el aprendizaje en línea, lo que aceleró la digitalización educativa, pero también puso en evidencia brechas de acceso tecnológico, desigualdad de conectividad y limitaciones en las capacidades institucionales.

Por su parte, la UNESCO emitió orientaciones para la continuidad pedagógica, subrayando la importancia de fortalecer estrategias híbridas que combinen enseñanza presencial y virtual, y que integren analítica del aprendizaje, interoperabilidad y acceso móvil. La trayectoria iberoamericana muestra un pasaje desde la enseñanza por correspondencia hasta ecosistemas digitales con analítica y acceso móvil. En este tránsito, la pandemia aceleró la virtualización, pero también evidenció brechas y la necesidad de fortalecer capacidades institucionales para modelos híbridos (CEPAL/UNESCO (2020) y CEPAL (2022)).

2.2. Antecedentes de Investigación Relacionados al Tema

2.2.1. Moodle



The screenshot displays the Moodle user profile interface. At the top, there is a navigation bar with 'Home', 'Dashboard', and 'My courses'. The user's name, 'JONATAN NATANAEL BRAVO ANDRADE', is prominently displayed next to a profile picture placeholder. A 'Message' button is visible next to the name. Below the name, there are three main sections: 'User details', 'Reports', and 'Login activity'. The 'User details' section includes fields for 'Email address', 'Country', 'City/town', and 'Timezone'. The 'Reports' section includes 'Browser sessions' and 'Grades overview'. The 'Login activity' section shows 'First access to site' and 'Last access to site' with timestamps. A 'Reset page to default' button is located in the top right corner.

Section	Field	Value
User details	Email address	e1351762081@live.uleam.edu.ec (Visible to other course participants)
	Country	Ecuador
	City/town	MANTA
	Timezone	America/Guayaquil
Reports	Browser sessions	
	Grades overview	
Login activity	First access to site	Wednesday, 6 August 2025, 1:51 AM (71 days 16 hours)
	Last access to site	Thursday, 16 October 2025, 6:02 PM (15 secs)

Ilustración 2 Interfaz de Moodle. Fuente: <https://aulavirtualmoodle.uleam.edu.ec>

Moodle es un sistema de gestión de aprendizaje diseñado para facilitar la educación en entornos virtuales. Su nombre proviene del acrónimo *Modular Object-Oriented Dynamic Learning Environment*, que significa “entorno modular y dinámico orientado a objetos”. La plataforma proporciona un entorno adaptable que combina herramientas para la comunicación y administración académica, fomentando la interacción entre estudiantes y docentes en modalidades tanto sincrónicas como asincrónicas. Gracias a estas características, Moodle se consolida como un entorno que promueve la participación activa y la adaptación a diferentes ritmos de aprendizaje (Poveda & de la Educación, 2007).

2.2.2. Platzi

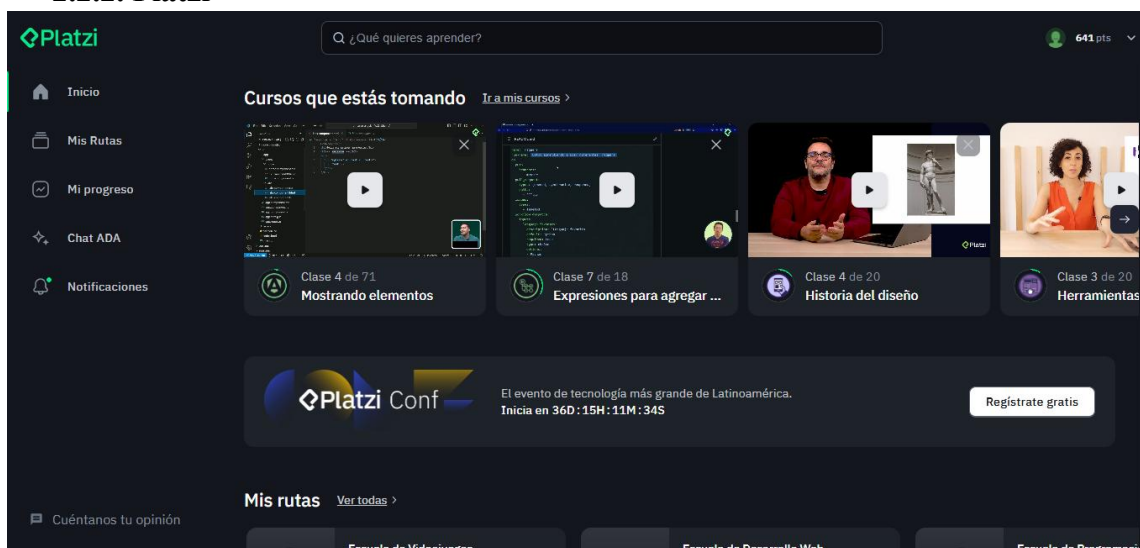


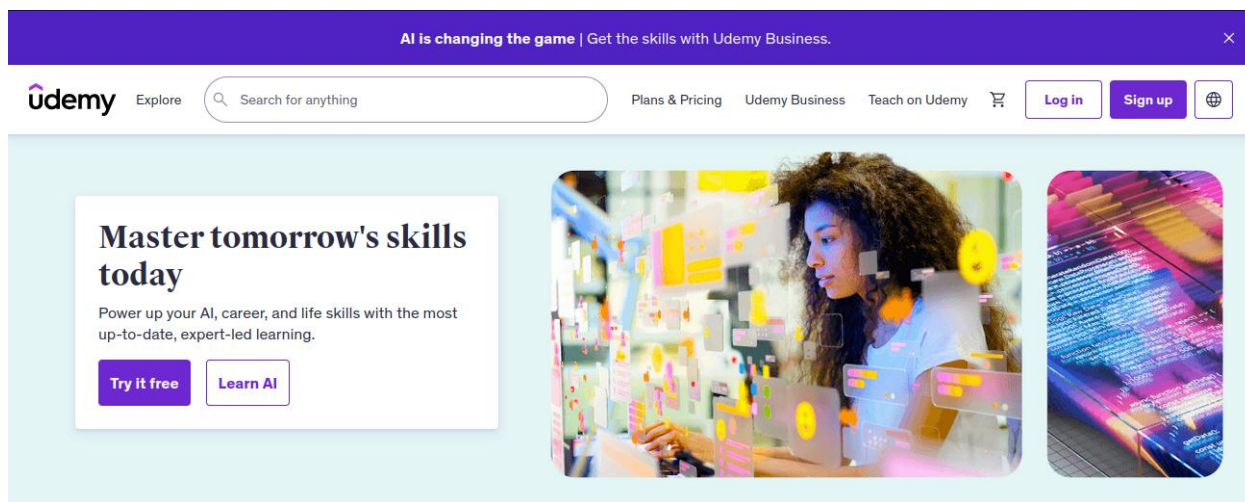
Ilustración 3 Pantalla principal de Platzi. Fuente: <https://platzi.com>

Platzi es una plataforma educativa en línea orientada al desarrollo de competencias tecnológicas y profesionales. Desde su creación en 2011, brinda formación en distintas áreas, como programación, marketing digital, diseño y gestión empresarial. Según un estudio realizado por Platzi (2020), los estudiantes que completan cursos en su plataforma tienen una

alta tasa de empleabilidad y una mejora significativa en sus habilidades técnicas. Platzi utiliza técnicas de personalización mediante la adaptación del contenido y la retroalimentación continua para mejorar la experiencia de aprendizaje.

Platzi ha implementado varias estrategias para mejorar la personalización en su plataforma. Una de ellas es el uso de rutas de aprendizaje, que son secuencias de cursos diseñadas para guiar a los estudiantes a través de un camino de aprendizaje estructurado. Además, Platzi ofrece sesiones en vivo y mentorías personalizadas para ayudar a los estudiantes a resolver dudas y obtener retroalimentación directa. Estas estrategias han demostrado ser efectivas para mantener el compromiso de los estudiantes y mejorar sus resultados de aprendizaje.

2.2.3. Udemy



Ready to reimagine your career?

Get the skills and real-world experience employers want with Career Accelerators.

Ilustración 4 Pantalla principal de Udemy. Fuente: <https://udemy.com>

Udemy constituye otra alternativa de aprendizaje digital que agrupa cursos de diferentes áreas del conocimiento. Creada en 2010, brinda a los instructores la posibilidad de diseñar sus propios contenidos y adaptar la experiencia de aprendizaje a los intereses de cada estudiante. Un estudio de Udemy (2019) encontró que el 80% de los estudiantes que completaron cursos en su plataforma experimentaron una mejora en su rendimiento profesional. Udemy utiliza algoritmos de recomendación para personalizar el contenido y sugerir cursos relevantes a los usuarios.

La revisión de plataformas de aprendizaje permite identificar diferentes enfoques. Moodle, como referente académico de software libre, ha demostrado solidez en contextos universitarios, pero su complejidad técnica y curva de aprendizaje pueden limitar la adopción en escenarios institucionales pequeños.

En contraste, plataformas de mercado como Udemy y Platzi se centran en la flexibilidad y escalabilidad de cursos en línea para audiencias masivas, con énfasis en la experiencia del usuario y modelos de suscripción. Sin embargo, presentan restricciones para la personalización institucional y la integración de procesos administrativos como pagos locales o emisión de certificados oficiales.

Un elemento transversal en la comparación es la interoperabilidad. Estándares como LTI 1.3 y LTI Advantage (IMS Global) facilitan la integración de distintos servicios en un LMS, garantizando que plataformas y herramientas externas (videoconferencias, evaluaciones, bibliotecas digitales) se comuniquen de forma segura y estandarizada. La guía de Blackboard en español constituye un recurso práctico para comprender la implementación de LTI en entornos educativos.

2.2.4. Comparación de Platzi y Udeemy

Tanto Platzi como Udeemy ofrecen soluciones robustas para la educación en línea, pero difieren en sus enfoques y áreas de especialización. Mientras que Platzi se centra en el desarrollo de habilidades tecnológicas con un enfoque en América Latina, Udeemy ofrece una gama más amplia de cursos en diversas disciplinas a nivel mundial. Ambos utilizan técnicas de personalización para mejorar la experiencia del usuario, pero Platzi se destaca por su enfoque en la comunidad y el aprendizaje continuo, mientras que Udeemy sobresale en la diversidad de contenido y accesibilidad global.

En términos de personalización, Platzi utiliza un enfoque más estructurado con rutas de aprendizaje y mentorías, mientras que Udeemy se basa en algoritmos de recomendación y una amplia oferta de cursos. Ambos enfoques tienen sus ventajas y desventajas. El enfoque de Platzi puede ser más adecuado para estudiantes que buscan un camino de aprendizaje claro y guiado, mientras que el modelo de Udeemy es ideal para aquellos que prefieren explorar diferentes temas a su propio ritmo.

Característica	Moodle (académico)	Udeemy / Platzi (mercado)	Propuesta LMS (Angular + Supabase)
Modelo de uso	Software libre, autogestionado	Plataforma comercial con cursos masivos	LMS institucional con integración local
Escalabilidad	Alta, pero requiere servidores	Muy alta, nube escalable globalmente	Escalable con Supabase en la nube

	propios		
Personalización institucional	Alta, mediante plugins, pero compleja	Limitada, centrada en catálogo general	Total, ajustada a roles y procesos locales
Gestión de pagos	Depende de plugins externos	Procesadores globales (Stripe, PayPal)	Validación local de comprobantes + <code>set_payment_status</code>
Emisión de certificados	Plugins de terceros	Sí, certificados digitales	Certificados PDF con pdfmake y control institucional
Interoperabilidad (LTI)	Sí, soporta LTI	Limitada, depende del proveedor	Integrable con LTI 1.3 / Advantage
Experiencia de usuario	Funcional, pero menos intuitiva	Optimizada, orientada a mercado	Moderna, en Angular 18, responsiva y usable
Costo	Sin licencias, pero con costo de servidores	Suscripciones / pago por curso	Infraestructura propia de bajo costo

Tabla 1 Tabla comparativa de LMS

2.2.5. Supabase

Supabase es un programa en el cual los desarrolladores no necesitan administrar servidores, cuenta de usuarios o base de datos de manera manual, debido a que la plataforma lo

realiza de forma automatizada, ofreciendo un proceso más simple y rápido. Según Pires (2025), Supabase es una solución de *backend-as-a-service* (BaaS) de código abierto que proporciona al desarrollador una infraestructura completa y lista para implementar aplicaciones web o móviles sin necesidad de administrar servidores manualmente.

2.2.6. Angular

Angular es un marco de desarrollo *open source* impulsado por Google y basado en TypeScript. Está diseñado para construir aplicaciones web de página única (*single-page applications*), simplificando la gestión de componentes y la comunicación entre cliente y servidor. para Gonçalves (2021) “una de las ventajas más destacadas de Angular es que puede adaptarse tanto a proyectos pequeños como a grandes aplicaciones empresariales. Su estructura está pensada para que las actualizaciones sean sencillas y se aprovechen las mejoras sin complicaciones” (párr.2). Este framework reúne distintas bibliotecas y herramientas que permiten manejar el enrutamiento, los formularios, la comunicación entre el cliente y el servidor, además de facilitar las pruebas y actualizaciones del código.

2.2.7. Digital Ocean (droplet)

Digital Ocean es un servicio de computación en la nube que posibilita la creación y gestión de servidores virtualizados denominados *Droplets*. Estos funcionan como máquinas virtuales independientes, similares a un VPS, y permiten alojar sitios web o aplicaciones personalizadas con recursos dedicados. Sirve para montar sitios web, aplicaciones o incluso plataformas completas en la nube, y se puede configurar de manera personalizada según las necesidades. Los usuarios pueden alojar aplicaciones web, ejecutar software que requiera

ajustes específicos o crear infraestructuras en la nube. En comparación con el hosting compartido, los Droplets ofrecen más control y recursos dedicados (KeepCoding, 2024).

2.2.8. PostgreSQL

PostgreSQL es un sistema gestor de bases de datos relacional y de código abierto, reconocido por su estabilidad, versatilidad y su capacidad para ajustarse a requerimientos complejos de almacenamiento y análisis de datos. A diferencia de otros sistemas de gestión de bases de datos relacionales, PostgreSQL admite tanto tipos de datos relacionales como no relacionales, lo que la convierte en una de las bases de datos más compatibles y estables. De acuerdo con BIM a diferencia de otros sistemas, permite manejar tanto datos relacionales como no relacionales, lo que la hace útil para aplicaciones que requieren análisis complejos, almacenamiento de datos geoespaciales o procesamiento de transacciones en línea.

PostgreSQL ofrece muchas opciones útiles para los usuarios funciones como, recuperar datos a un momento específico, realizar copias de seguridad en línea, controlar quién puede acceder a qué información, manejar transacciones dentro de otras transacciones, organizar los datos en diferentes espacios de tabla y asegurar que varias personas puedan trabajar al mismo tiempo sin problemas de conflicto.

2.2.8.1. RLS

RLS (Row-Level Security) es un mecanismo que permite controlar el acceso a filas específicas de una tabla según la política de seguridad definida. Esto significa que diferentes usuarios pueden ver o modificar solo los datos que están autorizados a manejar, ofreciendo un nivel de seguridad más detallado que el control de permisos tradicional.

2.2.9. Metodología de desarrollo XP

La metodología de Programación Extrema (XP) propone el desarrollo de software de alta calidad a través de iteraciones breves y entregas continuas. Su enfoque prioriza la sencillez del diseño, la comunicación permanente entre los miembros del equipo y la capacidad de respuesta ante los cambios del proyecto. Raeburn, (2025) manifiesta que se debe tener en cuenta al aplicar su metodología los siguientes valores esenciales:

- Simplicidad: Resolver problemas de la manera más sencilla posible.
- Comunicación: Fomentar el diálogo constante entre todos los miembros del equipo.
- Retroalimentación: Aprender de los resultados y ajustar el rumbo cuando sea necesario.
- Respeto: Valorar las habilidades y opiniones de cada integrante del equipo.
- Coraje: Tomar decisiones difíciles y enfrentar los desafíos con determinación.

Además, XP se estructura en torno a cinco reglas esenciales:

1. Planificación: Definir metas claras y alcanzables.
2. Diseño: Crear soluciones simples y efectivas.
3. Codificación: Escribir código de calidad y fácil de mantener.
4. Pruebas: Asegurar que el software funcione correctamente mediante pruebas continuas.
5. Implementación: Entregar el software de forma frecuente y en pequeñas dosis.

Estas reglas se implementan a través de doce prácticas específicas que guían el trabajo diario del equipo, como la programación en pareja, el desarrollo basado en pruebas y la integración continua.

2.3. Definiciones Conceptuales

- **Plataforma educativa (LMS):** Sistema digital que permite gestionar, distribuir y evaluar procesos de enseñanza-aprendizaje en entornos virtuales.
- **Personalización del aprendizaje:** Adaptación de contenidos, ritmos y trayectorias formativas a las características individuales de los estudiantes.
- **Talleres virtuales:** Espacios interactivos dentro del LMS que fomentan el aprendizaje práctico y colaborativo.
- **Experiencia de usuario (UX):** Conjunto de factores de diseño y usabilidad que determinan la satisfacción y eficiencia del usuario al interactuar con la plataforma.

2.4 Conclusiones relacionadas al marco teórico

El análisis de los antecedentes históricos evidencia que la educación virtual ha transitado desde modalidades tradicionales de enseñanza a distancia hacia el uso de plataformas digitales cada vez más robustas y escalables. Este recorrido ha puesto en evidencia la importancia de contar con entornos que garanticen no solo la distribución de contenidos, sino también la interacción, el seguimiento y la evaluación del aprendizaje.

Los antecedentes de investigación revisados demuestran que, aunque plataformas como Moodle, Udemy o Platzi han aportado de manera significativa al desarrollo del aprendizaje en línea, aún presentan limitaciones cuando se trata de responder a las necesidades específicas de instituciones de educación superior en el contexto ecuatoriano, especialmente en lo referente a la gestión integral de procesos académicos y administrativos.

Las definiciones conceptuales permiten comprender la relevancia de contar con un LMS moderno que combine aspectos pedagógicos, tecnológicos y de usabilidad. En particular, se identificó que la personalización del aprendizaje, la experiencia de usuario y la integración de procesos administrativos constituyen factores clave para mejorar la calidad educativa.

En consecuencia, el marco teórico sustenta la pertinencia de diseñar e implementar un sistema de gestión de aprendizaje basado en Angular 18 y Supabase CLI, que integre la gestión académica con procesos de inscripción, control de pagos mediante comprobantes, seguimiento del progreso estudiantil y emisión de certificados digitales, contribuyendo así al fortalecimiento de la educación virtual en la Universidad Laica Eloy Alfaro de Manabí y en instituciones similares del país.

Este marco teórico no solo destaca la necesidad de personalización en las plataformas educativas, sino que también establece un contexto para la implementación de tecnologías avanzadas que pueden mejorar la gestión y organización de talleres educativos, contribuyendo así a una experiencia de aprendizaje más rica y adaptada a las necesidades individuales de los usuarios.

Al analizar las plataformas Platzi y Udemy, se observa que ambas utilizan técnicas de personalización para mejorar la experiencia educativa, pero cada una con un enfoque diferente. Platzi se enfoca en rutas de aprendizaje estructuradas y mentorías personalizadas, mientras que Udemy utiliza algoritmos de recomendación y una amplia variedad de contenido. Estos ejemplos resaltan la importancia de la personalización en la educación en línea y ofrecen modelos a seguir para el desarrollo de nuevas plataformas educativas.

Del recorrido histórico y conceptual se desprende que un LMS efectivo en el contexto iberoamericano debe priorizar interoperabilidad, seguridad de datos, accesibilidad y soporte a modelos híbridos. La arquitectura Angular 18 + Supabase (Postgres + RLS) alinea estos requisitos: facilita autenticación, control fino por rol (RLS) y trazabilidad de aprendizajes (progreso/certificación), dejando abierta la adopción de LTI para integraciones futuras.

CAPITULO III: METODOLOGIA DE INVESTIGACION

3.1. Enfoque de la investigación

La investigación adopta un enfoque cualitativo, ya que se centra en la comprensión e interpretación del fenómeno de la educación virtual desde el contexto educativo. Según Flick (2015, OEI), el enfoque cualitativo busca describir en profundidad experiencias y procesos, más que medir variables numéricas.

Se emplea este enfoque porque el objetivo no es cuantificar datos estadísticos, sino analizar cómo un LMS puede dar respuesta a problemáticas específicas: baja finalización de cursos, validación manual de pagos y falta de trazabilidad pedagógica.

3.2. Tipo de investigación

El estudio es de tipo aplicado porque busca ofrecer una solución práctica a una necesidad institucional. Asimismo, se clasifica como descriptivo–explicativo, porque describe las limitaciones actuales de las plataformas en uso y explica de qué manera la propuesta LMS responde a esas limitaciones.

3.3. Diseño de investigación

El diseño es de carácter exploratorio y de desarrollo tecnológico: se construye un prototipo de LMS con las funciones SQL, tablas y RLS previamente definidas, y se analiza su pertinencia frente a modelos de referencia (Moodle, Udemy, Platzi). La validación no se basa en encuestas ni métricas, sino en análisis documental, revisión comparativa y pruebas prácticas controladas en el entorno Supabase local/producción.

3.4. Población y muestra

En lugar de población estadística, se trabaja con casos de uso representativos en el ámbito académico:

- Estudiantes: como usuarios finales del aprendizaje virtual.
- Instructores: responsables de diseñar cursos y validar pagos.

- Administradores: encargados de la gestión global y control del sistema.

La selección es intencional y no probabilística, siguiendo lo planteado en estudios de segmentación educativa disponibles en Dialnet, que recomiendan segmentar por rol funcional en contextos académicos.

3.5 Técnicas de recolección de información

- Revisión documental: análisis de literatura académica en español (Aretio, OEI, UNESCO, CEPAL, etc.) sobre evolución del e-learning y problemáticas de la virtualización educativa.
- Benchmarking comparativo: contraste del LMS propuesto con plataformas existentes (Moodle, Udemy, Platzi) a nivel de arquitectura, integración y procesos administrativos.
- Pruebas exploratorias en entorno controlado: ejecución de funciones SQL y validación de roles mediante RLS en Supabase para observar cómo resuelven las limitaciones detectadas.

3.6 Instrumentos

3.6.1. Matrices comparativas

La validación cualitativa incluyó el diseño de matrices comparativas entre el LMS propuesto y plataformas de referencia. Estas matrices permiten evidenciar fortalezas y limitaciones en criterios clave.

Criterio / Plataforma	Moodle (académico)	Udemy / Platzi (mercado)	LMS Propuesto
Gestión de pagos	Depende de plugins externos	Procesadores globales (Stripe, PayPal)	Validación local de comprobantes
Control de roles	Configurable con plugins	Básico (estudiante/instructor)	Políticas de seguridad RLS (estudiante, instructor, admin)
Interoperabilidad	Soporta LTI y SCORM	Limitada, depende del proveedor	Integración vía LTI 1.3 / APIs Supabase
Usabilidad	Funcional pero poco intuitiva	Optimizada para mercado masivo	Interfaz moderna en Angular, responsiva y con accesibilidad WCAG 2.1
Certificación	Con extensiones	Certificados digitales propios	Certificados PDF con pdfmake .
Costos	Sin licencia, requiere servidor	Suscripción/pago por curso	Bajo costo (infraestructura Supabase + DigitalOcean)

Tabla 2 Matriz comparativa de LMS. Fuente: propia

3.6.2. Registro de pruebas prácticas

Se documentaron casos de uso representativos para validar el funcionamiento del LMS, vinculados directamente a las funciones SQL y triggers creados:

Caso de prueba	Descripción	Función / Componente	Resultado esperado
Pago gratuito	Curso con price = 0	buy_course	Estado del pago = <i>paid</i> ,

			inscripción creada automáticamente en course_enrollments.
Pago pendiente con comprobante	Registro de compra con imagen	buy_course → payments	Estado inicial <i>pending</i> ; instructor valida con set_payment_status('approve').
RLS — Estudiante	Intento de aprobar pago	Política RLS	Operación bloqueada; estudiante solo puede acceder a sus cursos y compras.
RLS — Instructor	Validación de pago ajeno	Política RLS	Operación bloqueada; instructor solo aprueba pagos de sus cursos.
RLS — Admin	Validación global	Política RLS	Acceso autorizado a todos los pagos y cursos.
Progreso en video	Estudiante visualiza 90 % del video	user_video_progress + trigger tg_progress_upsert	Campo completed = true se activa automáticamente.

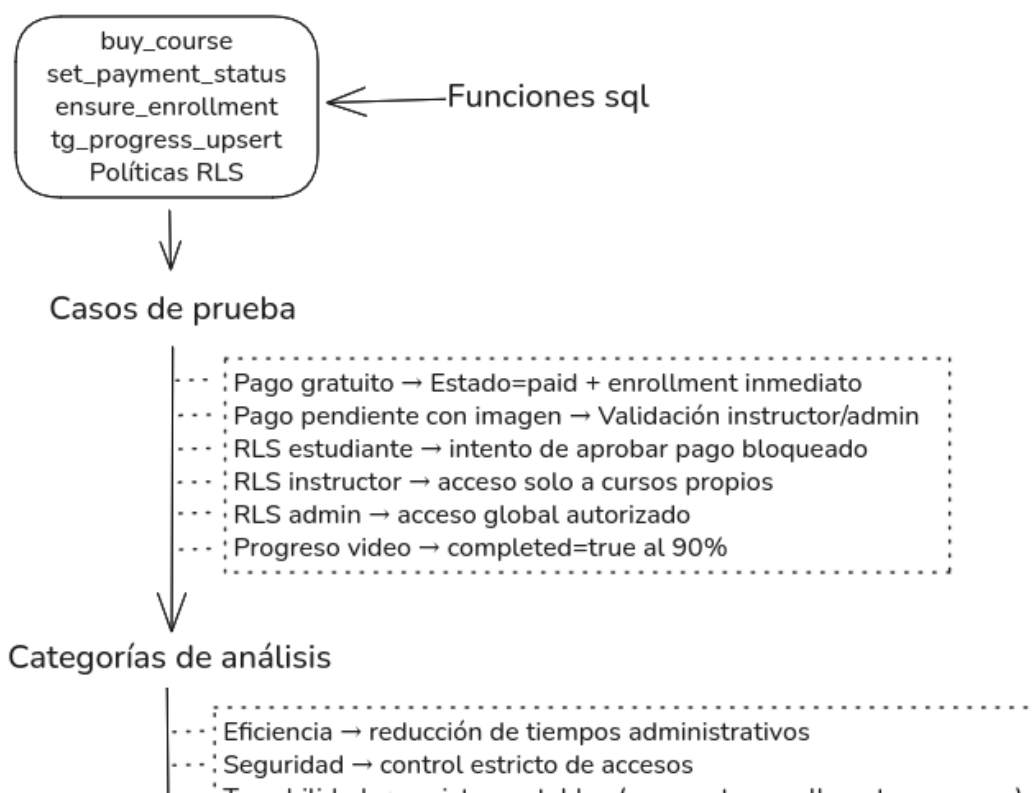
Tabla 3 Comparativa de pruebas práctica sde LMS. Fuente: Propia

3.6.3. Análisis categorial

Con base en las pruebas realizadas y la comparación documental, los hallazgos se organizaron en categorías analíticas que permiten interpretar el aporte del LMS propuesto:

Categoría	Evidencia observada	Aporte del LMS propuesto
Eficiencia	Validación de pagos se automatiza en menos pasos que en Moodle o Udemy.	Reducción de tiempos y simplificación administrativa con funciones <code>buy_course</code> y <code>set_payment_status</code> .
Seguridad	RLS bloquea intentos indebidos (ej. estudiante aprobando pagos).	Control estricto por rol (estudiante, instructor, admin).
Trazabilidad	Cada acción queda registrada en tablas (<code>payments</code> , <code>course_enrollments</code> , <code>user_video_progress</code>).	Seguimiento claro de inscripciones, pagos y progreso de estudiantes.
Usabilidad	Angular 18 ofrece una interfaz moderna y accesible.	Mejora la experiencia de estudiantes e instructores frente a Moodle (más técnico) o Udemy (más genérico).

Tabla 4 Matriz de aportes. Fuente: propia



3.7 Procedimiento

CAPITULO IV: MARCO PROPOSITIVO

4.1. Propuesta del proyecto

4.1.1 Descripción general de la propuesta

La propuesta consiste en el desarrollo de una plataforma web educativa diseñada para ofrecer cursos multiprofesionales y facilitar la organización y gestión de talleres personalizados. El sistema tendrá un enfoque modular, adaptable a distintos perfiles de usuario

(instructor, estudiante, administrador), y permitirá la personalización del contenido y la integración de pantallas de seguimiento y evaluación.

Las principales características previstas incluyen:

- Panel de administración para la gestión de cursos, usuarios, contenidos y pagos.
- Panel de instructor con dashboard incluyendo datos acerca de su contenido, pagos y gráficos.
- Compatibilidad multiplataforma, optimizada para navegadores y dispositivos móviles.
- Panel de usuario para visualizar el progreso de los cursos a los que se encuentra inscrito.

Este diseño busca atender la creciente demanda de formación flexible y adaptada a las necesidades de un público heterogéneo en entornos académicos, empresariales y técnicos.

4.1.2. Justificación técnica de la propuesta

La propuesta se justifica técnicamente por los siguientes factores:

1. Demanda de personalización en educación en línea: Según el informe de la OEI (2022), los entornos virtuales que incorporan adaptabilidad y analítica de aprendizaje registran mayores tasas de finalización de cursos.
2. Optimización para dispositivos móviles: Dado que, de acuerdo con el informe de We Are Social & Hootsuite (2023), el 56% del consumo de educación en

línea en Latinoamérica se realiza desde smartphones, el sistema será *mobile-first*.

3. Escalabilidad y seguridad: Se empleará una arquitectura basada en servicios y almacenamiento en la nube para garantizar disponibilidad y protección de datos, cumpliendo con la normativa de protección de datos aplicable.

Estos elementos aseguran que la propuesta no solo es viable, sino que responde a estándares internacionales de calidad en e-learning.

4.1.3. Viabilidad técnica, económica y operativa

4.1.3.1. Viabilidad técnica.

El desarrollo se realizará con tecnologías de código abierto y servicios en la nube como Digital Ocean. Esto garantiza soporte, actualizaciones y escalabilidad a bajo costo de licenciamiento.

4.1.3.2. Viabilidad económica

Se contempla un modelo de desarrollo en fases para distribuir la inversión inicial y reducir riesgos. El presupuesto se optimizará mediante:

- Uso de *frameworks* y bibliotecas gratuitas.
- Capacitación interna del personal para mantenimiento.
- Integración de herramientas ya existentes.

4.1.3.3. Viabilidad operativa

La plataforma contará con un sistema de administración accesible y fácil de operar, con roles diferenciados para cada tipo de usuario. Además, se darán capacitaciones para facilitar la adopción en instituciones y organizaciones.

De acuerdo con estudios recientes (UNESCO, 2021), este tipo de soluciones tiene alta probabilidad de implementación exitosa en contextos multiprofesionales, siempre que se acompañe de capacitación inicial y soporte técnico continuo.

4.1.4. Desarrollo de la propuesta

4.1.4.1 Etapas de desarrollo

El desarrollo de la plataforma se planifica en cinco etapas principales, siguiendo un enfoque incremental y ágil para asegurar entregas funcionales y retroalimentación temprana:

1. Análisis de requerimientos y diseño funcional
 - a. Recopilación de necesidades de los diferentes perfiles de usuario (instructores, estudiantes, administradores).
 - b. Elaboración de casos de uso y diagramas de flujo de procesos.
 - c. Definición de criterios de personalización y módulos principales.
2. Diseño técnico y prototipado
 - a. Arquitectura del sistema basada en servicios (REST API).
 - b. Prototipos de interfaz en herramientas como Figma o Adobe XD.
 - c. Validación inicial de la experiencia de usuario con un grupo piloto.
3. Desarrollo funciones y frontend

- a. Implementación de la lógica de base de datos con PostgreSQL con Row Level Security (RLS).
- b. Construcción de la interfaz con Angular, siguiendo principios *responsive*.
- c. Integración con la base de datos relacional PostgreSQL.

4. Integración de módulos y pruebas

- a. Incorporación de gestión de archivos (Thumbnails, videos y Comprobantes) y almacenamiento en la base de datos PostgreSQL.
- b. Pruebas unitarias, de integración y de usabilidad.
- c. Optimización para rendimiento y seguridad.

5. Despliegue y capacitación

- a. Implementación en un servidor de producción.
 - b. Establecimiento de un plan de mantenimiento y actualizaciones.
- Diseño de la base de datos: definición de tablas `courses`, `course_modules`, `module_videos`, `payments`, `course_enrollments`, `user_video_progress`.
 - Implementación de funciones SQL: `buy_course`, `ensure_enrollment`, `set_payment_status`, `review_instructor_request`, etc.
 - Políticas de seguridad (RLS): garantizar que los estudiantes solo vean sus cursos/compras, y que los instructores validen pagos de sus cursos.

- Integración frontend-backend: consumo de funciones y vistas mediante Angular Services (CourseService, AuthService).
- Flujos principales desarrollados: registro/login, creación de cursos, compra de curso con comprobante, validación de pago, emisión de certificado.

4.1.4.2 Recursos utilizados

Recursos humanos

- Desarrollador web full-stack: responsables de la base de datos y frontend.
- Diseñador UX/UI: encargado de la experiencia e interfaz de usuario.
- Especialista en e-learning: asesor en metodologías pedagógicas.
- Administrador de sistemas: gestión de servidores y bases de datos.
- Coordinador de proyecto: supervisión general y control de calidad.

Recursos tecnológicos

- Lenguaje principal: TypeScript.
- Framework de frontend: Angular 18.
- Base de datos: PostgreSQL.
- Gestión de seguridad: RLS en Supabase.
- CLI/Infraestructura: Supabase CLI (para local y despliegue en producción).
- Librerías externas: pdfmake (certificados), Bootstrap/NgBootstrap (interfaz), RxJS (gestión reactiva), Plyr(Reproducción de videos).

- Control de versiones: Git y GitHub.
- Entorno de desarrollo: VS Code + Docker.

Recursos económicos

- Costo de alojamiento en la nube y mantenimiento anual.
- Capacitación.

4.1.4.3. Metodología de desarrollo

Para el desarrollo de la plataforma web educativa se aplicó la metodología Programación Extrema (Extreme Programming, XP) adaptada a un contexto individual, debido a que el proyecto fue desarrollado por un solo autor. Esta metodología fue seleccionada por su enfoque ágil, iterativo e incremental, que prioriza la simplicidad del código, la retroalimentación continua y la entrega frecuente de funcionalidades.

A diferencia de metodologías tradicionales, XP permite adaptarse a cambios constantes durante el desarrollo, garantizando la calidad del producto mediante pruebas continuas y revisiones iterativas. En este proyecto, la metodología XP se aplicó siguiendo cinco fases principales, ajustadas a las condiciones de trabajo individual:

1. Planificación

En esta fase se definieron los objetivos funcionales del sistema y las historias de usuario correspondientes a los tres roles principales (administrador, instructor y estudiante). Se priorizaron los módulos clave: autenticación y roles, gestión de cursos, pagos, progreso, certificados y reportes.

Cada historia de usuario se documentó con criterios de aceptación claros, asegurando la trazabilidad con los requerimientos funcionales (RF1–RF35).

2. Diseño

Se elaboró la arquitectura general del sistema bajo un enfoque modular, aplicando principios de simplicidad y cohesión. El diseño técnico se representó mediante el diagrama entidad-relación (ERD) y los flujos de procesos desarrollados en Supabase y Angular. Se priorizó la mantenibilidad del código y la reutilización de componentes en Angular 18. Además, se aplicaron principios de Clean Code y Single Responsibility, propios de XP.

3. Codificación

El desarrollo se realizó en iteraciones semanales, implementando incrementos funcionales pequeños y verificables. Cada módulo fue integrado al backend (Supabase) utilizando funciones SQL y políticas de seguridad RLS (Row Level Security). El frontend se desarrolló con Angular 18, aplicando componentes independientes y servicios reutilizables para la conexión con el backend. El control de versiones se gestionó con Git y GitHub, siguiendo las prácticas de integración continua propuestas por XP.

4. Pruebas

Las pruebas se ejecutaron de forma continua durante cada iteración. Se realizaron pruebas unitarias de funciones SQL (ej. *buy_course*, *ensure_enrollment*, *set_payment_status*), pruebas de integración entre frontend y backend, y pruebas funcionales de

los flujos principales (registro, compra, validación de pagos, generación de certificados). La validación de las políticas de RLS garantizó la seguridad y el acceso diferenciado por rol.

5. Implementación y retroalimentación

Cada incremento funcional fue desplegado en entornos locales y posteriormente en Supabase en producción, siguiendo una estrategia de integración y despliegue continuo. La retroalimentación provino del análisis comparativo con plataformas existentes (Moodle, Platzi, Udemy) y de la evaluación técnica de desempeño y usabilidad. Los resultados de cada iteración se utilizaron para mejorar el rendimiento, optimizar las consultas SQL y ajustar la experiencia del usuario.

4.1.4.3.1. Aplicación práctica de XP individual

Dado que el desarrollo fue realizado por un solo autor, las prácticas colaborativas de XP, como la programación en pareja, se sustituyeron por revisiones personales periódicas y pruebas automatizadas.

Asimismo, la comunicación con usuarios se simuló mediante escenarios de prueba que representaban los tres roles del sistema. El enfoque XP individual permitió mantener la agilidad y la mejora continua durante todo el ciclo de desarrollo, asegurando entregas funcionales estables y progresivas.

4.1.4.4. Cronograma de actividades

Se propone un cronograma de 16 semanas dividido en fases:

Fase	Actividad	Duración	Responsable
1	Levantamiento de requerimientos y análisis	2 semanas	Coordinador + Especialista e-learning
2	Diseño funcional y prototipado	3 semanas	Diseñador UX/UI
3	Desarrollo funciones	4 semanas	Desarrollador full-stack
4	Desarrollo frontend	3 semanas	Desarrollador full-stack
5	Integración y pruebas	2 semanas	Equipo técnico
6	Despliegue y capacitación	2 semanas	Coordinador + Administrador de sistemas

Tabla 5 Cronograma de desarrollo. Fuente: propia

4.2. Requerimientos Funcionales

a) Gestión de usuarios y roles

- RF1: El sistema debe permitir el registro e inicio de sesión de usuarios autenticados en Supabase.
- RF2: El sistema debe diferenciar accesos mediante roles (estudiante, instructor, administrador) aplicando políticas de Row Level Security (RLS).
- RF3: El estudiante solo podrá visualizar y matricularse en cursos disponibles.
- RF4: El instructor podrá crear cursos RF5: El administrador podrá gestionar usuarios y validar pagos únicamente de los cursos bajo su responsabilidad.

b) Gestión de cursos y contenidos

- RF5: El sistema debe permitir al instructor la creación, especificando título, descripción, nivel, categoría, precio y estado.
- RF6: El sistema debe gestionar módulos dentro de los cursos.
- RF7: El sistema debe gestionar videos dentro de los módulos, incluyendo título, descripción, URL y duración.
- RF8: El estudiante debe poder visualizar cursos, módulos y videos de los que esté inscrito.

c) Gestión de inscripciones y pagos

- RF9: El estudiante podrá comprar un curso a través de la función `buy_course`.
- RF10: Si el curso es gratuito, el sistema debe registrar el pago como “paid” y crear la matrícula automáticamente (`ensure_enrollment`).
- RF11: Si el curso requiere pago, el sistema debe registrar el pago en estado pending, adjuntar comprobante y esperar validación del administrador.
- RF12: El admin podrá aprobar o rechazar pagos mediante la función `set_payment_status`.
- RF13: El sistema debe actualizar automáticamente las matrículas de los estudiantes al aprobar un pago.

d) Gestión del progreso y certificación

- RF14: El sistema debe registrar el avance de visualización de videos en la tabla `user_video_progress`.
- RF15: El trigger `tg_progress_upsert` debe marcar un video como completado cuando el usuario supere el 90 % de la duración.
- RF16: El sistema debe calcular el porcentaje de progreso del curso del estudiante.
- RF17: Al completar un curso, el sistema debe generar un certificado en PDF usando `pdfmake`, incluyendo datos del curso, del estudiante y del instructor.

e) Reportes y métricas

- RF18: El administrador debe acceder a un resumen global: número de cursos, estudiantes activos, instructores activos, pagos procesados.
- RF19: El instructor debe acceder a estadísticas de sus cursos: número de inscritos, ingresos por pagos aprobados, calificación promedio.
- RF20: El estudiante debe acceder a su historial de cursos inscritos y finalizados, con opción de descargar certificados.

f) Foro y comentarios

- RF21: El sistema debe permitir crear, obtener y eliminar comentarios asociados a cursos, módulos o videos, organizados jerárquicamente por hilos y respuestas.
- RF22: El sistema debe soportar la paginación de comentarios y limitar la profundidad máxima de las respuestas por hilo.

g) Valoraciones de cursos

- RF23: El estudiante podrá valorar los cursos en los que esté inscrito o completados, una sola vez por usuario.

- RF24: El sistema recalculará automáticamente las métricas de calificación (promedio y conteo) después de cada valoración.

h) Gestión de categorías

- RF25: El sistema debe permitir el CRUD de categorías, asegurando identificadores únicos (slug) y asociación con cursos.
- RF26: El sistema debe listar y paginar los cursos por categoría, mostrando conteos agregados.

i) Gestión de instructores

- RF27: El sistema debe permitir a un usuario solicitar convertirse en instructor, enviando información adicional y adjuntos.
- RF28: El administrador podrá revisar, aprobar o rechazar solicitudes de instructor, registrando observaciones y trazabilidad del proceso.

j) Finanzas e ingresos del instructor

- RF29: El sistema debe mostrar al instructor un reporte de ingresos por los últimos 12 meses, considerando la comisión aplicada.
- RF30: El sistema debe listar los estudiantes inscritos en los cursos de un instructor con paginación y filtros.
- RF31: El sistema debe mostrar el listado de “top cursos del instructor” por número de ventas o calificación promedio.

k) Búsqueda y paginación avanzada

- RF32: El sistema debe permitir búsqueda y paginación de usuarios por rol y estado.
- RF33: El sistema debe permitir búsqueda, ordenamiento y paginación de cursos y pagos.

l) Gestión de almacenamiento (storage)

- RF34: El sistema debe listar, crear y eliminar objetos en buckets, soportando prefijos, delimitadores y paginación.
- RF35: El sistema debe validar los archivos antes de subirlos (tamaño máximo, tipo permitido) y permitir simulaciones de inserción seguras.

m) Notificaciones y tiempo real

- RF36: El sistema debe emitir eventos en tiempo real (como compra aprobada, progreso alcanzado o nuevo comentario).

4.3. Requerimientos No Funcionales

a) Seguridad

- RNF1: El sistema debe implementar Row Level Security (RLS) para garantizar accesos diferenciados por rol.

- RNF2: Todas las transacciones de pago deben quedar registradas con estado y trazabilidad en la tabla payments.
- RNF3: El sistema debe impedir que usuarios no autorizados aprueben pagos.
- RNF4: Los datos deben transmitirse mediante protocolos seguros (HTTPS).

b) Rendimiento y disponibilidad

- RNF5: El tiempo de respuesta de las funciones SQL no debe superar los 2 segundos en consultas estándar.
- RNF6: El tiempo de validación de pagos debe ser inferior a 24 horas desde el registro hasta la aprobación/rechazo.
- RNF7: El sistema debe ser accesible al menos el 99 % del tiempo en ambiente de producción Supabase.

c) Usabilidad

- RNF8: La interfaz debe ser intuitiva, responsiva y accesible, desarrollada en Angular 18.
- RNF9: El sistema debe cumplir con criterios básicos de accesibilidad WCAG 2.1 (contraste, foco visible, etiquetas).
- RNF10: La navegación entre cursos, módulos y videos debe realizarse en menos de tres clics desde el inicio de sesión.

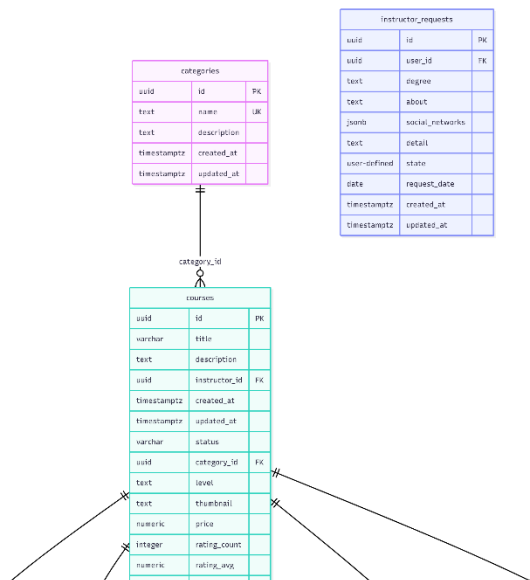
d) Escalabilidad y mantenimiento

- RNF11: La arquitectura debe permitir escalar horizontalmente mediante los servicios gestionados de Supabase.
- RNF12: El código debe versionarse en GitHub.
- RNF13: El sistema debe permitir agregar nuevos roles o funcionalidades sin modificar la lógica central de las funciones SQL existentes.

e) Portabilidad

- RNF14: La aplicación debe ser accesible desde navegadores modernos (Chrome, Firefox, Edge) y dispositivos móviles.

4.4. Diagrama ERD



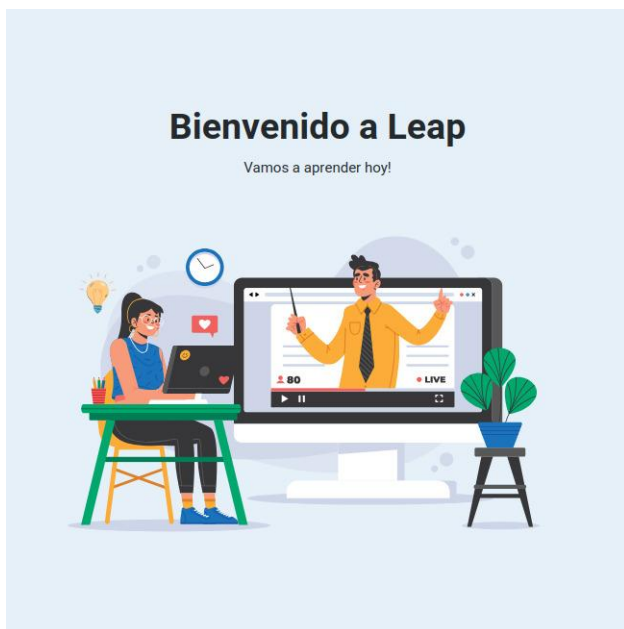
4.5. Interfaces

4.5.1. Inicio



Ilustración 7 Interfaz de inicio. Fuente: propia

4.5.2. Inicio de sesión



Inicio de sesión

 Iniciar con Google

Ilustración 8 Interfaz de inicio de sesión. Fuente: propia

4.5.3. Cursos

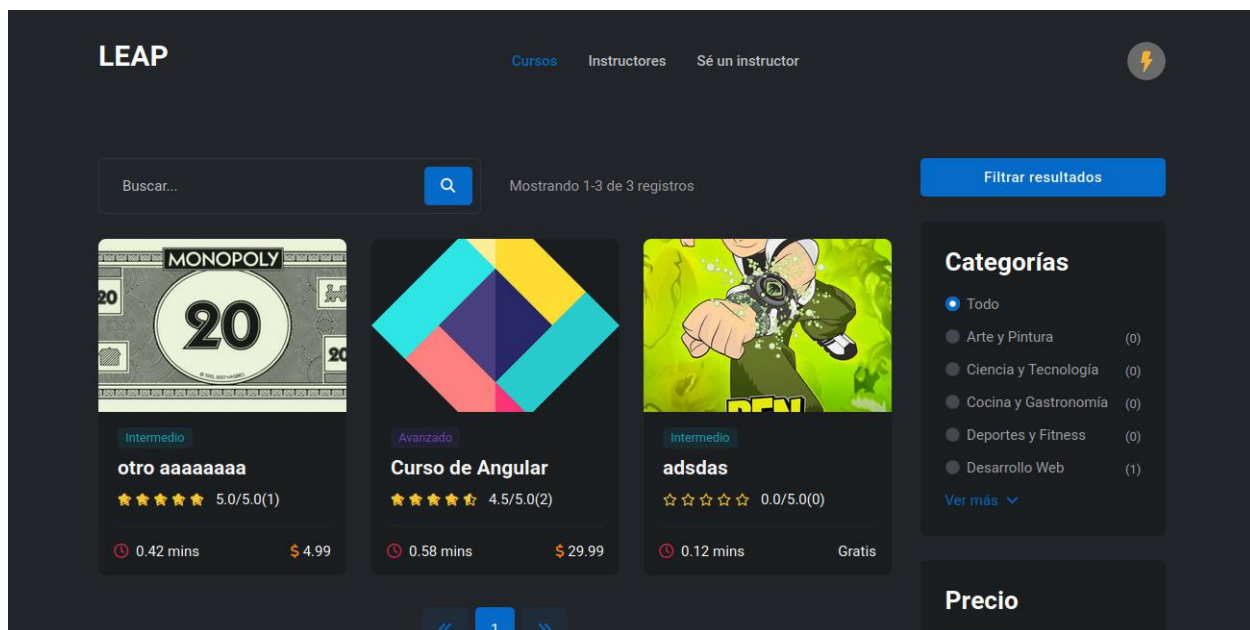


Ilustración 9 Interfaz de cursos. Fuente: propia

4.5.4. Detalle de curso

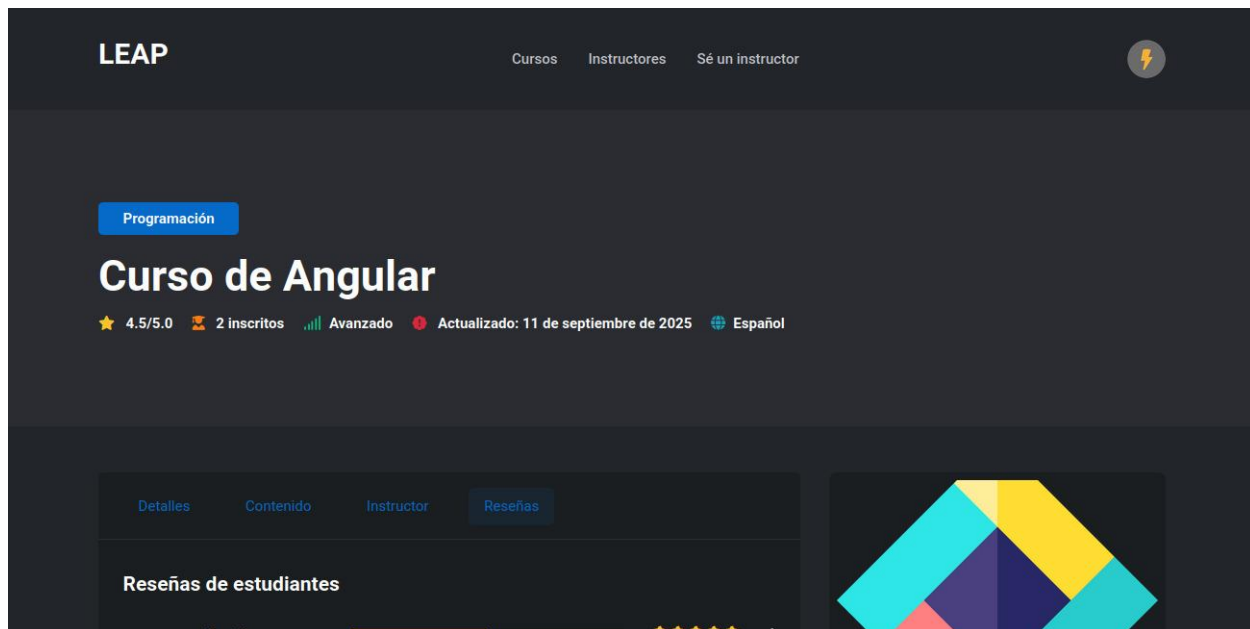


Ilustración 10 Interfaz de detalle de curso. Fuente: propia

4.5.5. Compra de curso

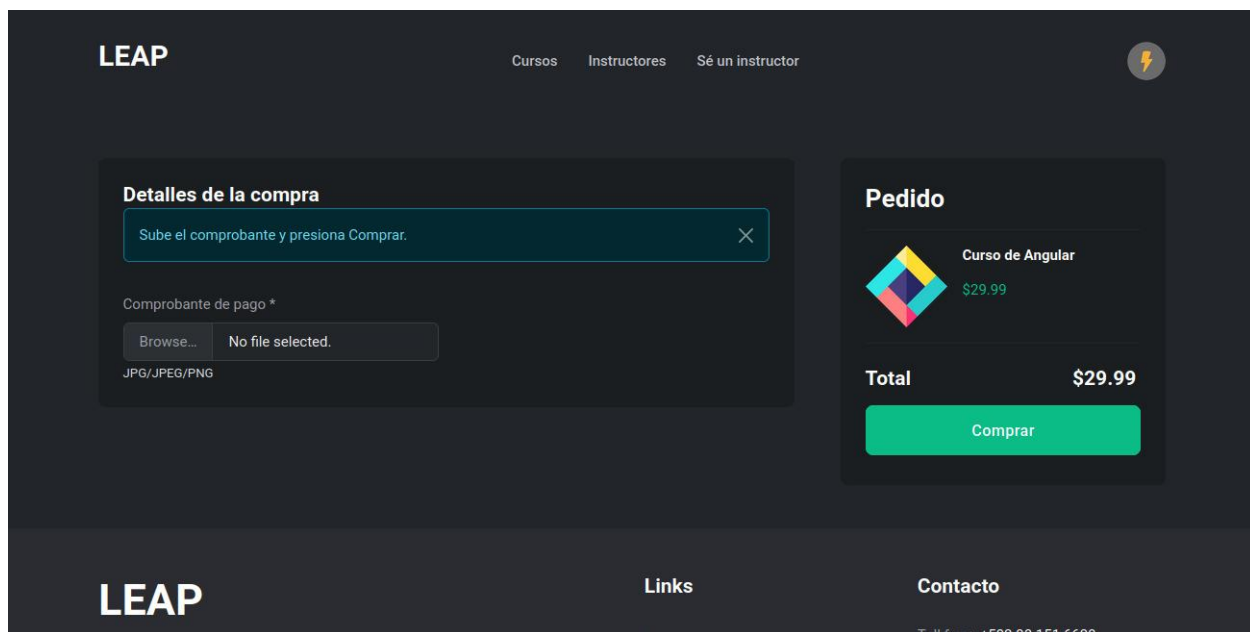
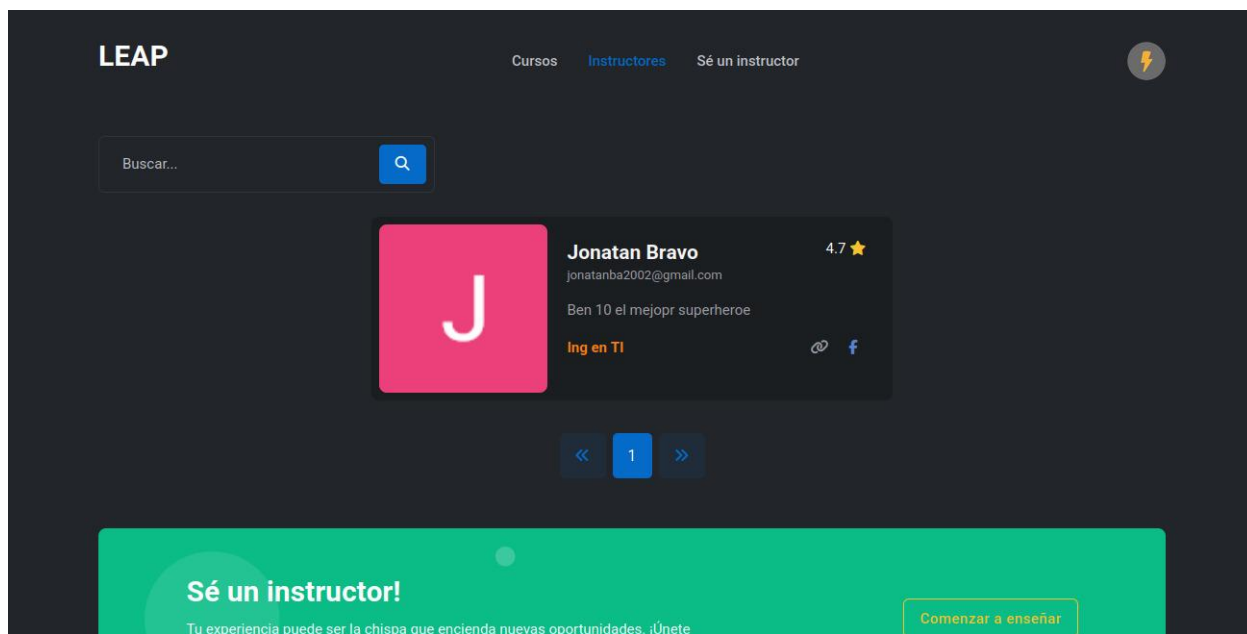


Ilustración 11 Interfaz de detalle de curso. Fuente: Propia

4.5.6. Instructores



4.5.7. Aplicar como instructor

Ilustración 12 Interfaz de instructores. Fuente: propia

Ilustración 13 Interfaz de aplicación para instructor. Fuente: propia

4.5.8. Dashboard de estudiante

Curso	Total de videos	Videos completados
otro aaaaaaaa 100%	1	1
Curso de Angular 0%	2	0
adsdas 0%	1	0

Ilustración 14 Interfaz de perfil del estudiante. Fuente: propia

4.5.9. Dashboard de instructor

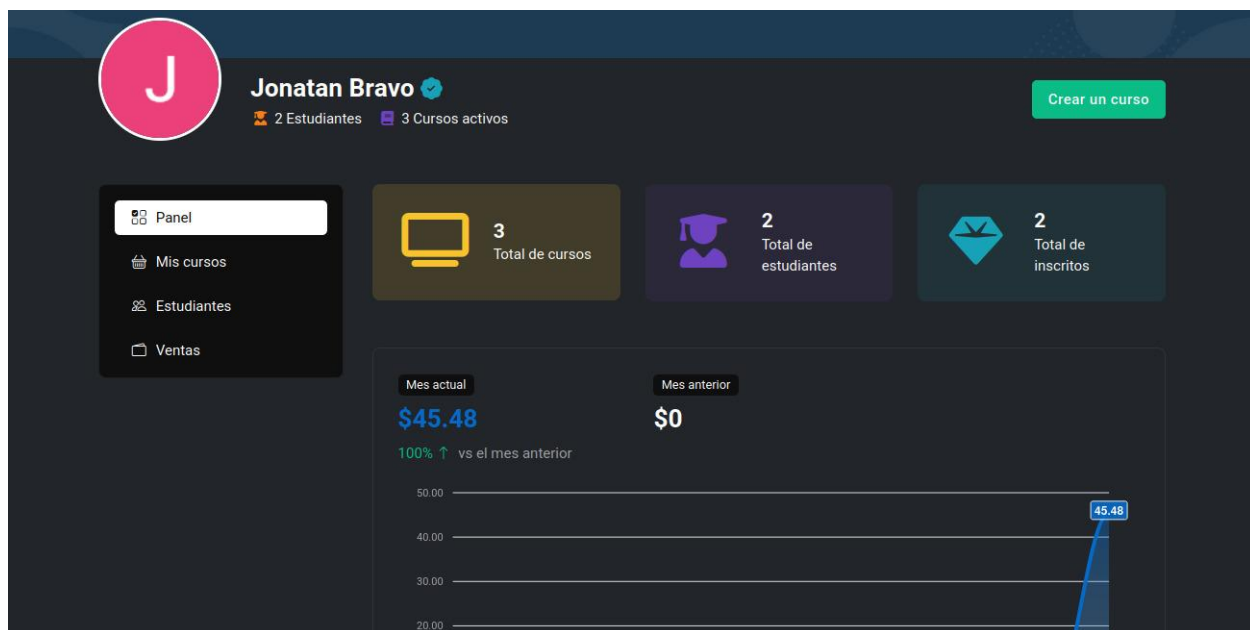


Ilustración 15 Interfaz de perfil del instructor. Fuente: propia

4.5.10. Cursos de instructor

Jonatan Bravo

2 Estudiantes 3 Cursos activos

Crear un curso

Mis cursos

Search

Curso	Estudiantes inscritos	Estado	Precio
otro aaaaaaaa 1 videos	1	active	\$4.99
Curso de Angular 2 videos	2	active	\$29.99
adsdas 1 videos	1	active	Gratis

Ilustración 16 Interfaz de cursos del instructor. Fuente: propia

4.5.11. Estudiantes inscritos de instructor

Mis estudiantes

Search

Estudiante	Cursos inscritos	Mi estudiante desde	Acción
Natanael Andrade skrrrapapakaka@gmail.com	3	Sep 10, 2025	
Jonatan Bravo jonatan.laboral2022@gmail.com	1	Sep 12, 2025	

Mostrando 1 hasta 2 de 2 entradas

Ilustración 17 Interfaz de estudiantes inscritos del instructor. Fuente: propia

4.5.12. Ventas de instructor

Compras de tus cursos

Estado: Todos Desde: mm / dd / yyyy Hasta: mm / dd / yyyy

ID	Monto	Estado	Fecha
67eacb45-362f-422c-81ac-bd0f6be897e1	\$29.99	paid	12/09/2025 18:57
b597fe79-ddc2-4f09-9837-e028f3661c7a	\$29.99	paid	12/09/2025 18:52
227a137b-5237-44b8-94f0-9e5bd818c538	\$0.00	paid	10/09/2025 11:04
c066a3af-b9fe-462a-a6c2-27b25dbd39c4	\$4.99	paid	10/09/2025 11:02

Ilustración 4 Interfaz de ventas del instructor. Fuente: propia

4.5.13. Creación de curso

The screenshot shows the first step of the course creation process, titled 'Detalles del curso'. At the top, there are three numbered tabs: 1 (Detalles del curso), 2 (Miniatura), and 3 (Contenido). The 'Detalles del curso' tab is active. Below the tabs, the form includes a 'Título' field with the placeholder 'Enter course title'. Below this are three fields: 'Categoría' (a dropdown menu), 'Nivel' (a dropdown menu), and 'Precio' (a text input with the value '0'). At the bottom, there is an 'Add description' section with a rich text editor toolbar containing icons for bold, italic, underline, link, unlink, bulleted list, numbered list, indent, outdent, link, unlink, and source code. The text area below the toolbar contains the placeholder 'Insert text here ...'.

Ilustración 19 Interfaz de creación de curso 1. Fuente: propia

The screenshot shows the second step of the course creation process, titled 'Miniatura'. The 'Miniatura' tab is active. Below the tabs, there is a large dashed box representing the thumbnail area. Inside this box, there is a blue square icon with a white mountain and sun. Below the icon, the text 'Sube tu miniatura aqui' is displayed. Below this text are two buttons: 'Browse...' and 'No file selected.'. Below the buttons, a note states: 'Note: Only JPG, JPEG and PNG. Our suggested dimensions are 600px * 450px. Larger image will be cropped to 4:3 to fit our thumbnails/previews.' At the bottom of the form, there is a 'Previsualización' section with the word 'preview' below it.

Ilustración 20 Interfaz de creación de curso 2. Fuente: propia

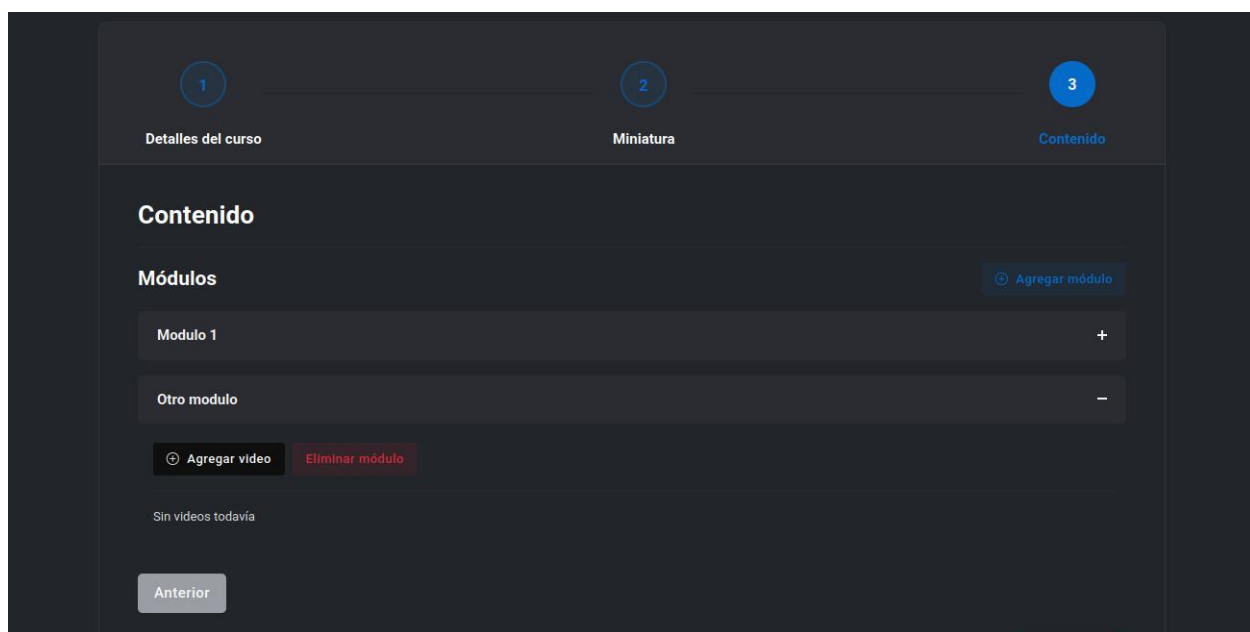


Ilustración 21 Interfaz de creación de curso 3. Fuente: propia

4.5.14. Estudiantes

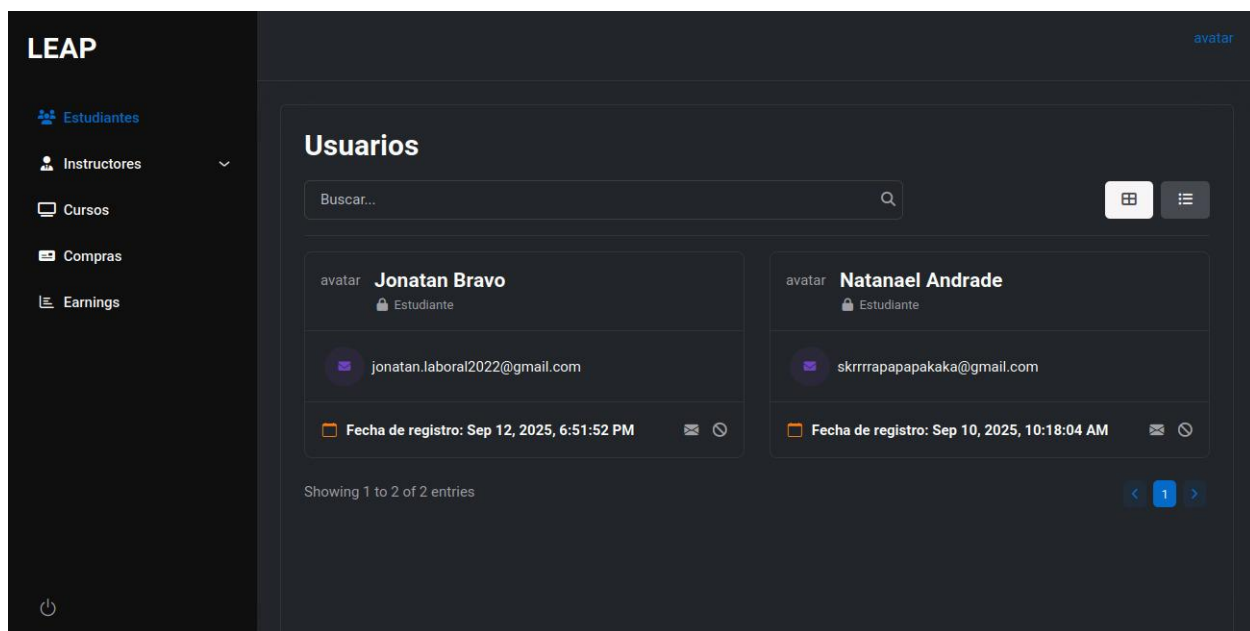


Ilustración 22 Interfaz de estudiantes. Fuente: propia

4.5.15. Instructores

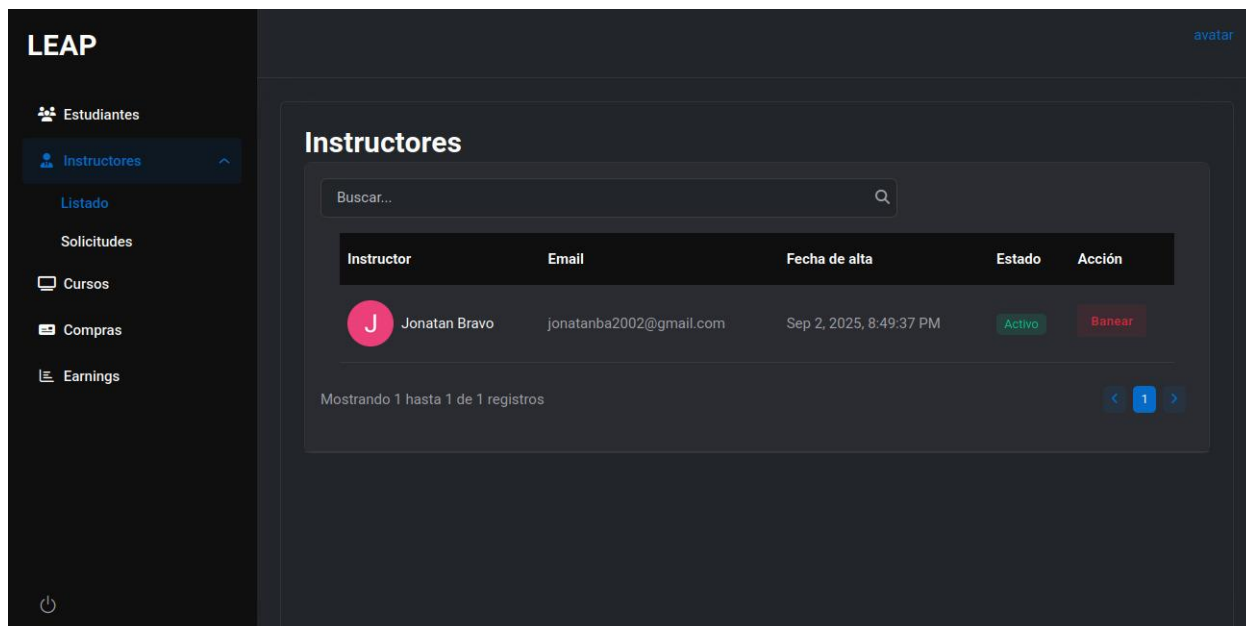


Ilustración 23 Interfaz de instructores. Fuente: propia

4.5.16. Aplicaciones para instructor

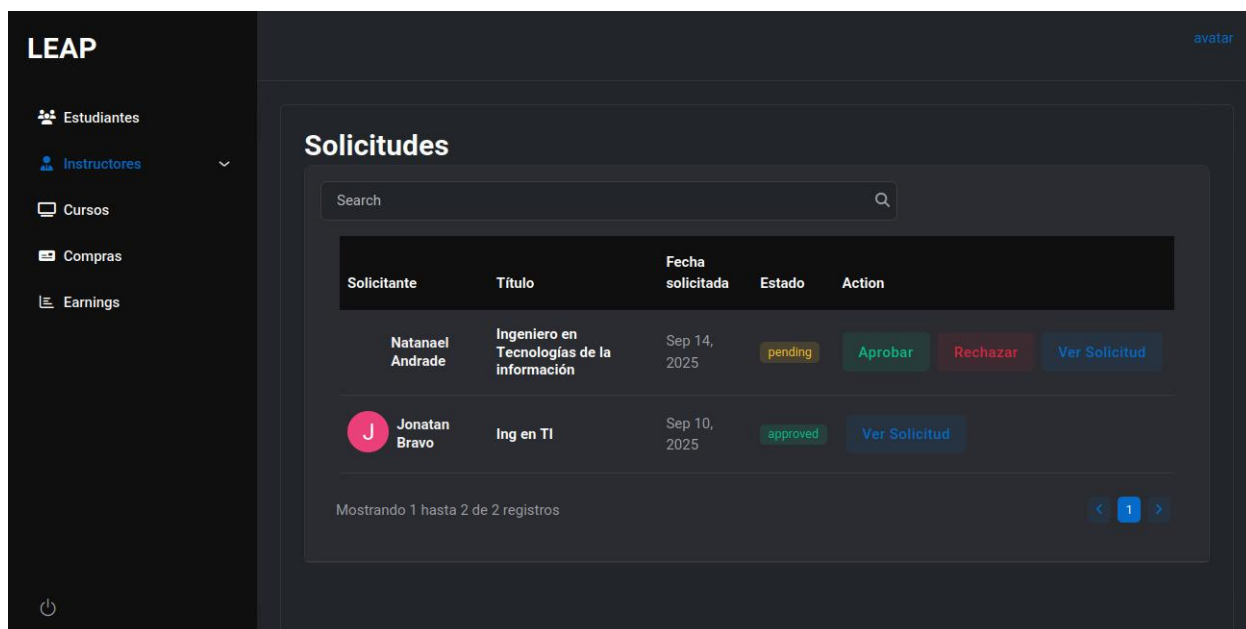


Ilustración 24 Interfaz de aplicaciones de instructores. Fuente: propia

4.5.17. Detalle de aplicación para instructor

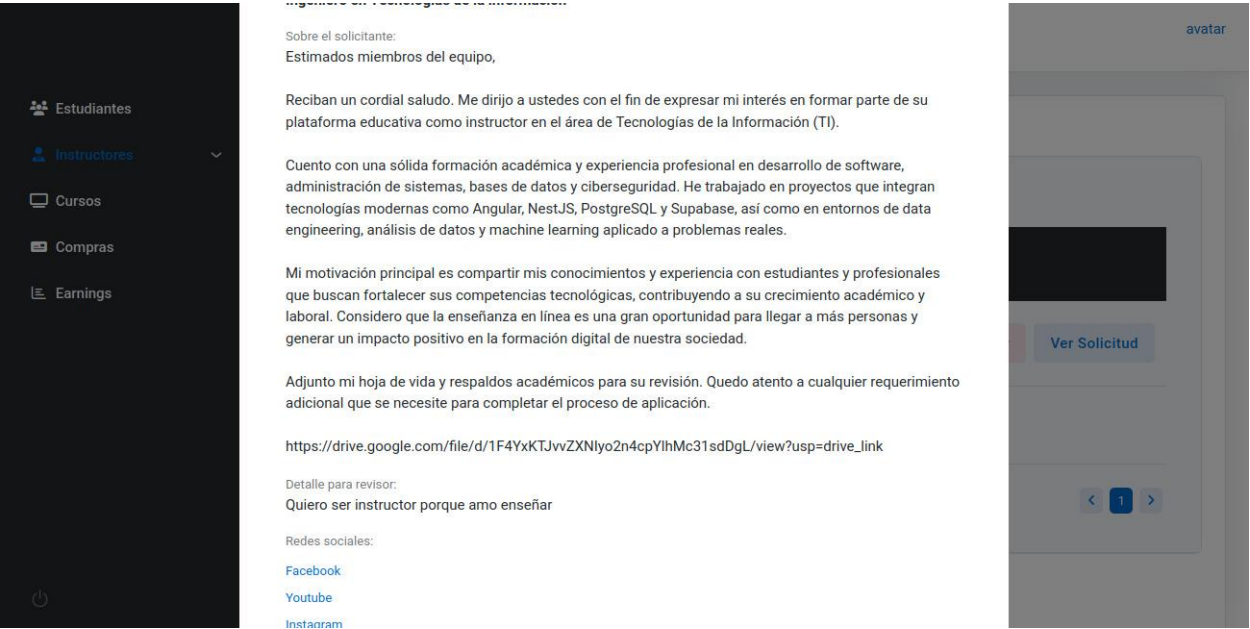


Ilustración 25 Interfaz de detalle de aplicación para instructor. Fuente: propia

4.5.18. Cursos

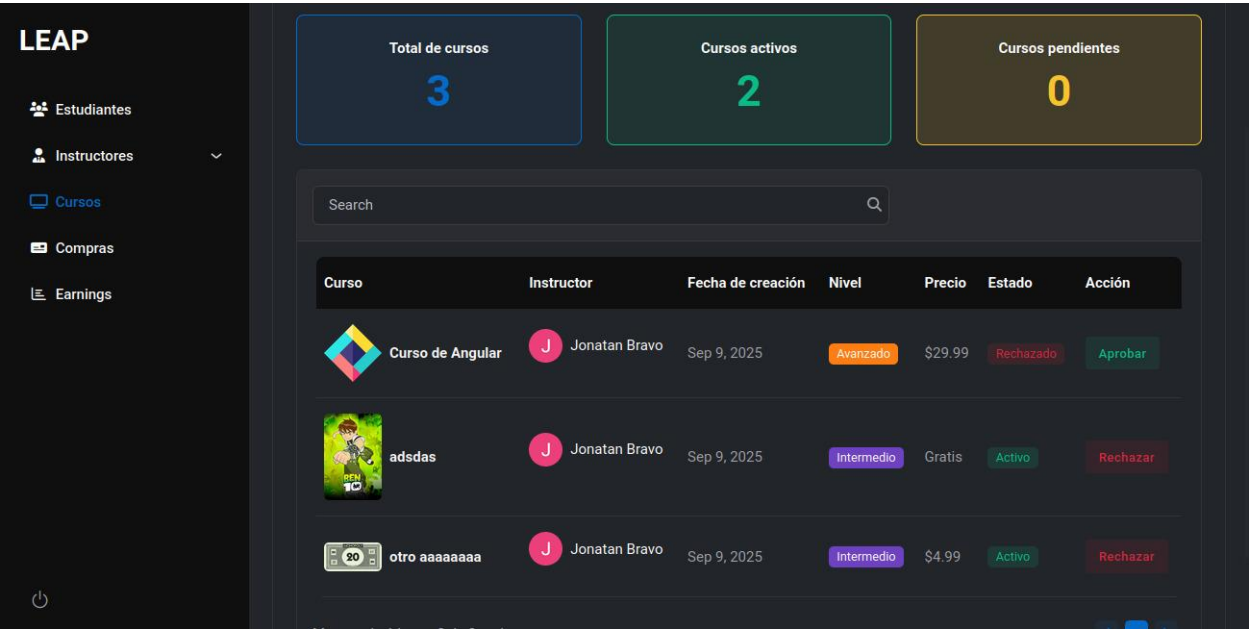


Ilustración 26 Interfaz decursos. Fuente: propia

4.5.19. Pagos

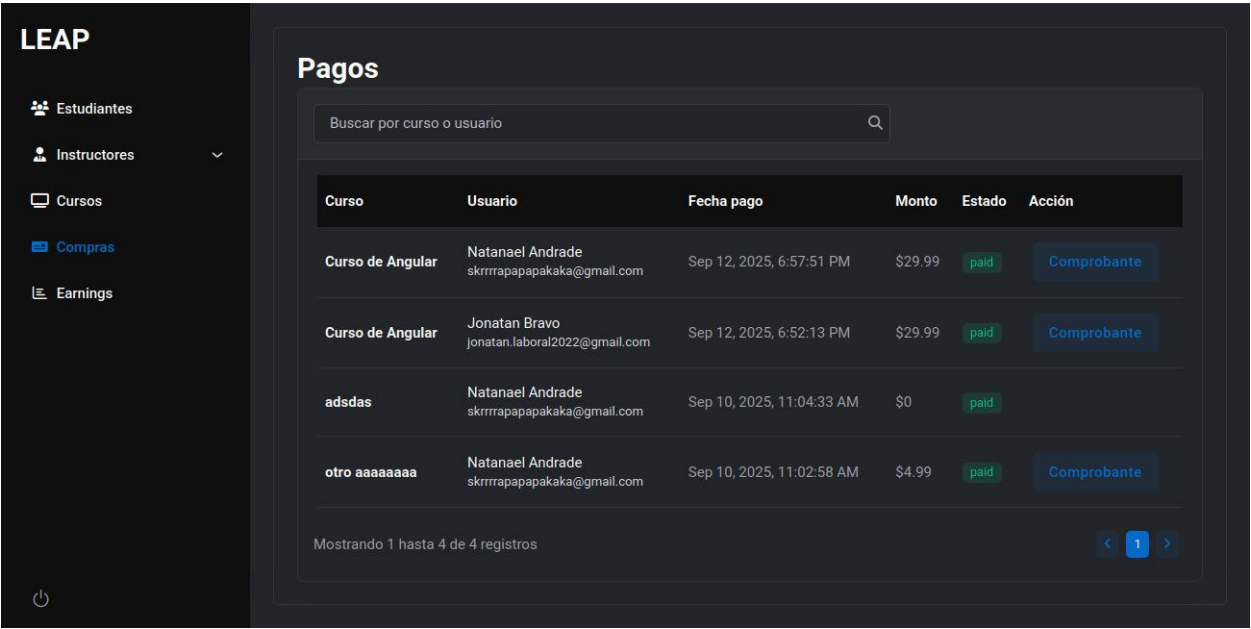


Ilustración 27 Interfaz de pagos. Fuente: propia

4.5.20. Recuperar contraseña

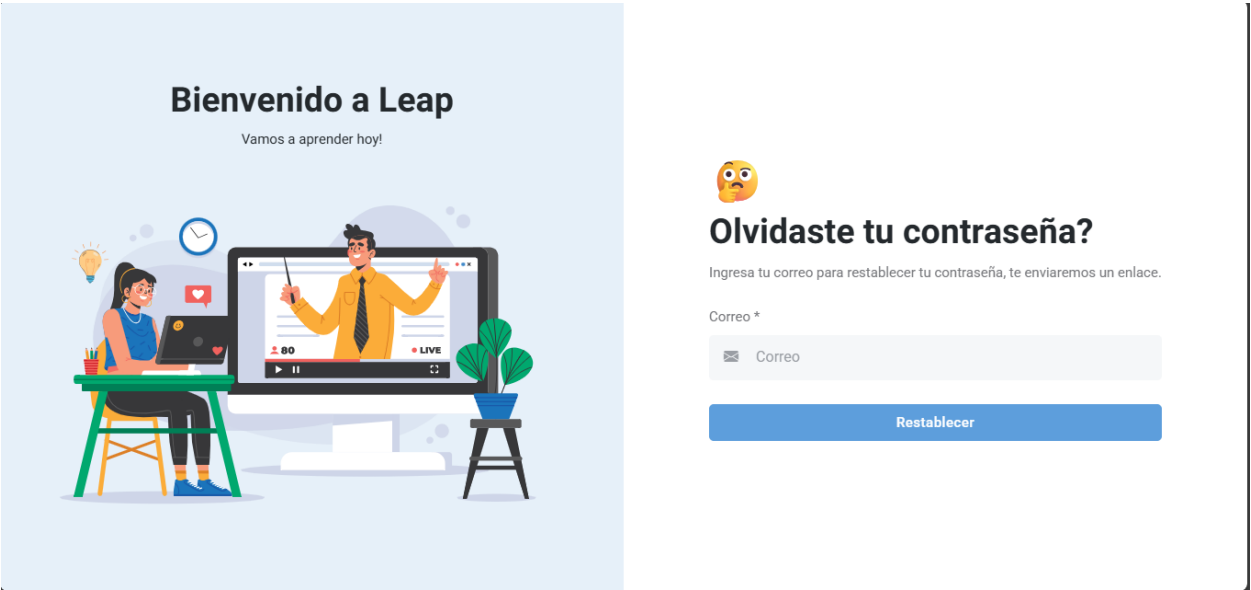


Ilustración 5 Interfaz de recuperar contraseña. Fuente: propia

4.6. Validación de la Propuesta

4.6.1. Métodos de validación empleados

Para garantizar que la plataforma cumpla con los objetivos técnicos y pedagógicos propuestos, se utilizaron diversos métodos de validación:

1. Pruebas de seguridad: validación de RLS (usuarios solo acceden a sus datos, instructores no pueden ver pagos de otros).
2. Pruebas de funcionalidad:
3. Creación de curso → módulos → videos.
4. Compra de curso → inserción de pago con comprobante.
5. Validación del instructor → cambio de estado de pago (pending → paid/rejected).
6. Generación de certificado PDF para cursos completados.
7. Pruebas de despliegue: ejecución en Supabase local y posterior despliegue en producción Supabase.
8. Pruebas de integración: consumo desde Angular de las funciones SQL y verificación de respuestas JSONB.

4.6.2. Resultados de la validación

Los resultados obtenidos reflejan un alto nivel de aceptación y rendimiento:

- Rendimiento: el tiempo medio de carga por página fue inferior a 1.67 segundos con 4 usuarios simultáneos.

Los aspectos cualitativos a destacar son:

- Interfaz intuitiva y clara.
- Valoración positiva de las métricas de progreso.
- Sugerencias para ampliar las opciones de interacción en talleres (p. ej., salas de trabajo colaborativas).

4.6.3 Ajustes y mejoras posteriores

A partir de la ejecución local efectuada, se implementaron las siguientes mejoras:

1. Optimización de la personalización
 - a. Inclusión de un cuestionario inicial más detallado para definir el perfil del usuario y ajustar automáticamente las recomendaciones de cursos y talleres.
2. Mejoras de accesibilidad
 - a. Ajustes en contraste de color, compatibilidad con lectores de pantalla y navegación por teclado para cumplir con las pautas WCAG 2.1.
3. Ampliación de funcionalidades colaborativas
 - a. Integración de salas virtuales para trabajo en grupo y pizarras digitales interactivas.
4. Rendimiento y seguridad
 - a. Implementación de un sistema de *caching* para acelerar la carga de contenidos y fortalecimiento de protocolos de cifrado de datos.

Estos ajustes aseguran que la plataforma esté alineada con las mejores prácticas en tecnología educativa, mejorando su usabilidad, accesibilidad y capacidad de personalización.

Conclusiones

A lo largo del desarrollo de este proyecto de tesis se evidenció que la implementación de un sistema de gestión de aprendizaje (LMS) el cual constituye una alternativa viable y eficiente para optimizar los procesos académicos y administrativos en entornos educativos. El diseño de una base de datos estructurada permitió gestionar de manera coherente cursos, módulos, videos, inscripciones y pagos, garantizando la integridad referencial y la trazabilidad de la información.

La construcción de funciones SQL como `buy_course`, `set_payment_status` y `ensure_enrollment` demostró que es posible reducir los tiempos de validación de pagos y asegurar la matrícula automática de los estudiantes, fortaleciendo la eficiencia administrativa y la experiencia del usuario. Asimismo, la aplicación de políticas de Row Level Security (RLS) confirmó que el control de accesos diferenciados según el rol del usuario es una medida efectiva para preservar la seguridad y la transparencia en el uso de la plataforma.

Al manejar el progreso de los cursos de los estudiantes a través de la función `is_course_completed`, generados por `pdmake`, permitió que de forma automática los usuarios tuvieran la facilidad de descargar sus certificados lo que añadió un valor académico importante al proceso formativo al otorgar evidencia verificable de la culminación de los cursos.

En conjunto, la propuesta desarrollada no solo respondió a las problemáticas identificadas sobre la limitación de cursos, sino que también aportó un modelo escalable y adaptable a las necesidades institucionales, sentando las bases para futuras innovaciones en la educación digital dentro los usuarios que buscan plataformas personalizadas de aprendizaje en línea.

Recomendaciones

Se recomienda continuar fortaleciendo la arquitectura del LMS mediante la incorporación de mecanismos de monitoreo de rendimiento que permitan anticipar posibles problemas de escalabilidad y garantizar la disponibilidad de la plataforma en escenarios de alta demanda. Resulta también aconsejable profundizar en la adopción de estándares de interoperabilidad como LTI 1.3/Advantage, que faciliten la integración con otros entornos académicos y sistemas institucionales.

En términos de usabilidad, es pertinente mantener un proceso continuo de mejora de la interfaz, ajustándola a los estándares de accesibilidad WCAG 2.1, de manera que la plataforma pueda ser utilizada sin barreras por estudiantes con diferentes capacidades. Adicionalmente, se sugiere incorporar herramientas colaborativas como foros o espacios de interacción grupal que enriquezcan el aprendizaje y fomenten la construcción colectiva del conocimiento.

Desde el punto de vista pedagógico, se recomienda realizar estudios cualitativos posteriores que recojan la percepción de estudiantes e instructores sobre la experiencia de uso, con el fin de retroalimentar y ajustar el sistema a las necesidades reales de la comunidad universitaria. Asimismo, podría ampliarse el alcance del LMS hacia funciones analíticas que permitan generar reportes predictivos de rendimiento o de riesgo de abandono, lo que fortalecería la toma de decisiones académicas.

En este contexto es esencial que se implementen soluciones tecnológicas que permitan optimizar la gestión de cursos en línea y la experiencia educativa de los usuarios, a su vez el seguimiento del progreso de los estudiantes y la integración de procesos administrativos, con el fin de mejorar la eficiencia, la organización y la calidad del aprendizaje en entornos virtuales.

Finalmente, se considera fundamental establecer lineamientos éticos y de privacidad en el manejo de los datos, asegurando que el uso de la plataforma mantenga principios de equidad, transparencia y responsabilidad institucional, reforzando así la confianza de la comunidad educativa en la herramienta tecnológica desarrollada.

Bibliografía

1EdTech . (s. f.). *Learning Tools Interoperability® (LTI®) 1.3 & LTI Advantage Overview*. Rescatado de: <https://www.1edtech.org/standards/lti>

Blackboard Help (ES). (s. f.). *Integraciones LTI*. https://help.blackboard.com/es-es/Learn/Administrator/SaaS/Integrations/Learning_Tools_Interoperability

CEPAL & UNESCO. (2020). *La educación en tiempos de la pandemia de COVID-19*. CEPAL. Rescatado de: <https://www.cepal.org/es/publicaciones/45904-la-educacion-tiempos-la-pandemia-covid-19>

CEPAL. (2022). *Educación en tiempos de pandemia: Una mirada latinoamericana*. CEPAL. Rescatado de: https://repositorio.cepal.org/bitstream/handle/11362/48204/1/S2200803_es.pdf

Fundación Carolina (comp.). (2021). *La educación superior en Iberoamérica en tiempos de pandemia*. Rescatado de: <https://www.fundacioncarolina.es/wp-content/uploads/2021/06/LibroLaeducacionSuperiorEnIberoamerica.pdf>

García Aretio, L. (1999). *Historia de la educación a distancia*. **RIED**, 2(1), 8–27. Rescatado de: <https://revistas.uned.es/index.php/ried/article/view/2084>

García Aretio, L. (2020). Bosque semántico: ¿educación/enseñanza/aprendizaje a distancia, virtual, en línea, digital, eLearning...? **RIED**, 23(1), 9–28. Rescatado de: <https://revistas.uned.es/index.php/ried/article/view/25495>

Gonçalves, M. J. (2021). *¿Qué es Angular y para qué sirve?* Hiberus. <https://www.hiberus.com/crecemos-contigo/que-es-angular-y-para-que-sirve/>

IBM. (2024). *¿Qué es PostgreSQL?*. About PostgreSQL. <https://www.ibm.com/mx-es/topics/postgresql>

KeepCoding. (2024). *¿Qué es un Droplet?*. KeepCoding <https://keepcoding.io/blog/que-es-un-droplet/>

Pires, J. (2025). *Supabase: ¿Qué es y por qué el backend de open source está en auge?* HostGator Blog. Recuperado de: <https://www.hostgator.net.ec/blog/que-es-supabase/>

Poveda, L. A., & de la Educación, D. F. (2007). Moodle como recurso didáctico. *Eduotec*, 2-3. <https://url-shortener.me/6TV4>

Raeburn, A. (2025). *La programación extrema (XP) produce resultados, pero ¿es la metodología adecuada para ti?* Asana. Recuperado de: <https://asana.com/es/resources/extreme-programming-xp>

UNESCO (Oficina Montevideo). (2020). *Enseñar en tiempos de COVID-19: Guía teórico-práctica para docentes*. Rescatado de: <https://unesdoc.unesco.org/ark:/48223/pf0000373868>

W3C/WAI. (2025). *Sumario de WCAG 2*. Rescatado de: <https://www.w3.org/WAI/standards-guidelines/wcag/es>

W3C. (2025). *Web Content Accessibility Guidelines (WCAG) 2.1*. Rescatado de: <https://www.w3.org/TR/WCAG21/>