

Instituto Tecnológico y de Estudios Superiores de Monterrey

Planeación de sistemas de software

Gpo 105

Arquitectura y Estándares

Diego Dávila Hernández A01285584

Fernando Morán Fougerat A01284623

Imanol Armando González Solís A00835759

Ramiro Alejandro Garza Villarreal A01178167

Rogelio Garza Rendón A01571384

Campus Monterrey

2 jun 2025

Índice

Documentación de diseño	3
Objetivo del proyecto	3
Detalles del proyecto	3
Tareas principales del equipo	3
Directrices e instrucciones claras	3
Seguimiento del estado del proyecto y elaboración de informes	3
2. Arquitectura del Sistema	4
3. Estructura del Frontend (React)	4
Introducción al documento de diseño (indicar de manera general el contenido de documento de diseño y su utilidad)	el 5
Diagramas de Casos de uso	6
Diagramas de clases	7
Diagramas de paquetes	7
Diagramas de secuencia	8
Diagramas de componentes	8
Diagramas de estados	9
Matriz de estados/eventos	11
Matriz de estados del usuario	11
Interfaces de usuario	12

Documentación de diseño

CodeCourses es una plataforma con gamificación para la resolución de desafíos de programación. Su objetivo es medir y motivar las habilidades técnicas de los usuarios a través de desafíos programáticos, recompensas y estadísticas de rendimiento, todo con una interfaz atractiva y moderna. El proyecto utiliza tecnologías modernas como React, Go, PostgreSQL y herramientas de contenerización para ofrecer una experiencia robusta, modular y escalable.

Objetivo del proyecto

Desarrollar una plataforma web gamificada que permita a los usuarios resolver desafíos de programación, visualizar su progreso, competir con otros y recibir recompensas por sus logros. La plataforma debe ser modular, escalable y segura, utilizando tecnologías modernas en frontend, backend, base de datos y contenedores.

Detalles del proyecto

- Nombre del proyecto: CodeCourses
- Propósito: Evaluación técnica y formativa para Tech Mahindra
- Duración estimada: 4 semanas
- Tecnologías clave: React, Go, PostgreSQL, Docker,, Clerk, SonarQube, Gemini API
- Entregables: Aplicación funcional, documentación técnica, repositorio en GitHub, presentación final

Tareas principales del equipo

- Diseño de arquitectura y base de datos
- Desarrollo de componentes frontend y navegación
- Implementación de API RESTful en Go
- Creación de lógica de ejecución de código en contenedores
- Diseño e integración del sistema de recompensas y medallas
- Módulo de administración de retos y usuarios
- Configuración de análisis de código con SonarQube
- Redacción y mantenimiento de documentación

Directrices e instrucciones claras

- Mantener coherencia en la estructura de carpetas y convenciones de nombres
- Validar inputs del usuario desde el frontend y backend
- Implementar control de errores en todos los endpoints
- Seguir principios de seguridad al manejar autenticación y ejecución remota de código
- Usar ramas de Git diferenciadas por feature y hacer pull requests con revisión cruzada

Seguimiento del estado del proyecto y elaboración de informes

- Bitácoras semanales por integrante
- Control de avances por tareas asignadas
- Revisión de progreso en sesiones de stand-up
- Reportes de calidad de código con SonarQube
- Documentación viva y actualizada en GitHub Wiki

2. Arquitectura del Sistema

Tipo: Microservicios **Componentes:**

Frontend: React + Tailwind
Backend: Go (HTTP REST API)
Base de Datos: PostgreSQL

Autenticación: ClerkContenedores: Docker

Calidad de Código: SonarQubeAsistencia por IA: Gemini API

• CI/CD: En planificación

3. Estructura del Frontend (React)

Librería de auth: @clerk/clerk-react

Librería de estilos: Tailwind

Componentes principales:

Home: landing de bienvenidaLoginPage: inicio de sesión

• SignUpPage: registro de usuario

• ChallengesPage: listado general de retos

• ChallengePage: resolución de retos individuales

LeaderboardPage: tabla de posicionesRewardsPage: sistema de recompensas

• ReposPage: vista de repositorios

AdminPage: gestión de retos y usuariosLayout: diseño general de navegación

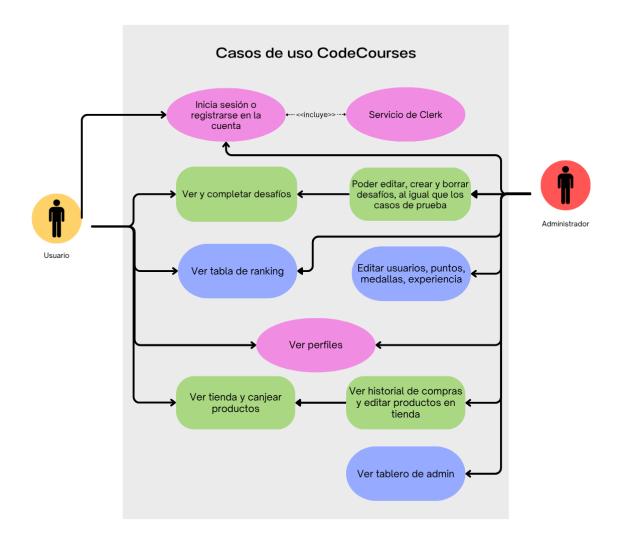
Introducción al documento de diseño (indicar de manera general el contenido del documento de diseño y su utilidad)

Este documento de diseño tiene como propósito describir la estructura técnica, lógica y visual de la plataforma DevQuest. Su función principal es servir como referencia tanto para el equipo de desarrollo como para revisores técnicos y académicos. Incluye información clave sobre la arquitectura del sistema, diseño de componentes, estructura de base de datos, flujos de interacción y vistas de usuario. También proporciona diagramas UML y de estados que permiten comprender el funcionamiento interno y externo del sistema.

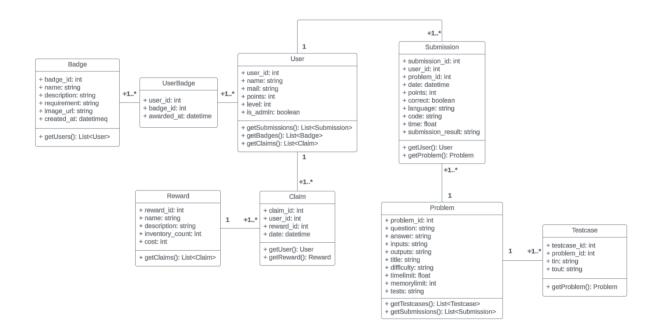
La utilidad del documento radica en facilitar el mantenimiento, escalabilidad y comprensión del sistema, tanto en etapas actuales como futuras del proyecto. También sirve como evidencia documental para procesos de evaluación académica o de implementación profesional.

Diagramas de Casos de uso

https://www.canva.com/design/DAGpP-Du78c/7jaHhq3_leZwu-AMwn7kOw/edit?utm_content=DAGpP-Du78c&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

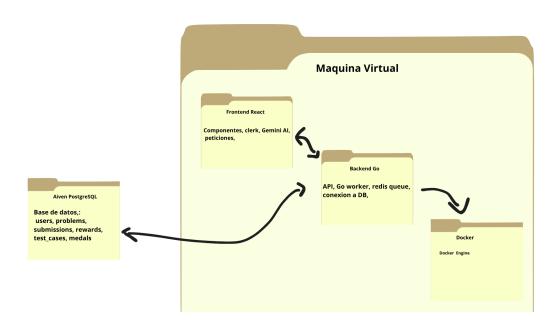


Diagramas de clases

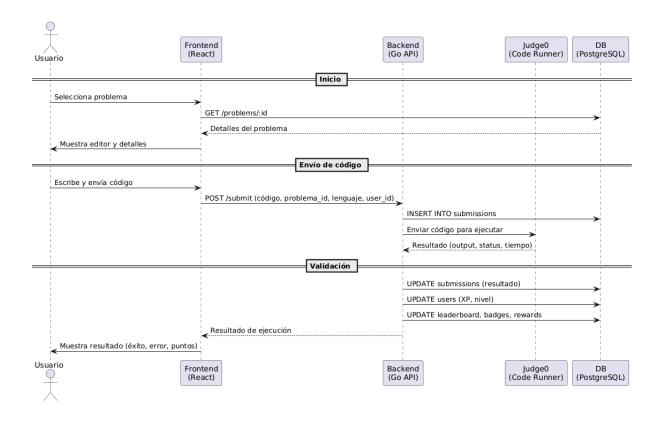


Diagramas de paquetes

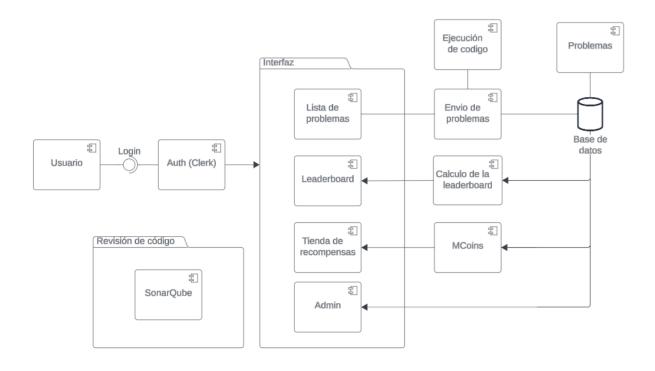
https://www.canva.com/design/DAGpQNECCRs/ppJ4FPhwr3p4pEbL_ktNVA/edit?utm_content=DAGpQNECCRs&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton



Diagramas de secuencia

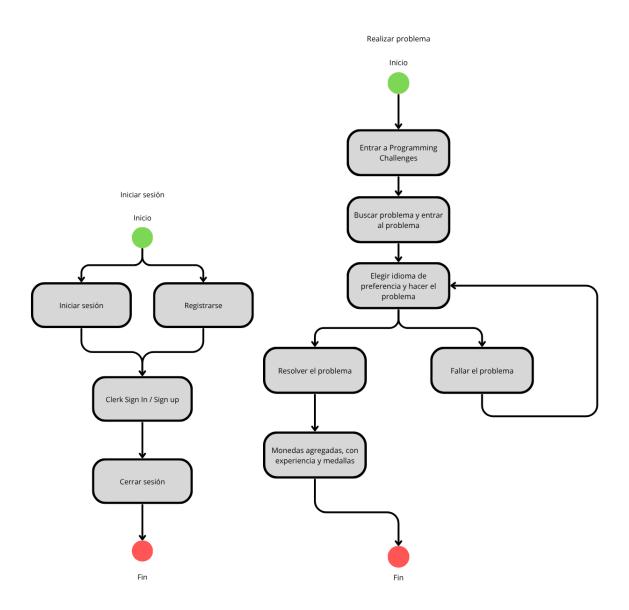


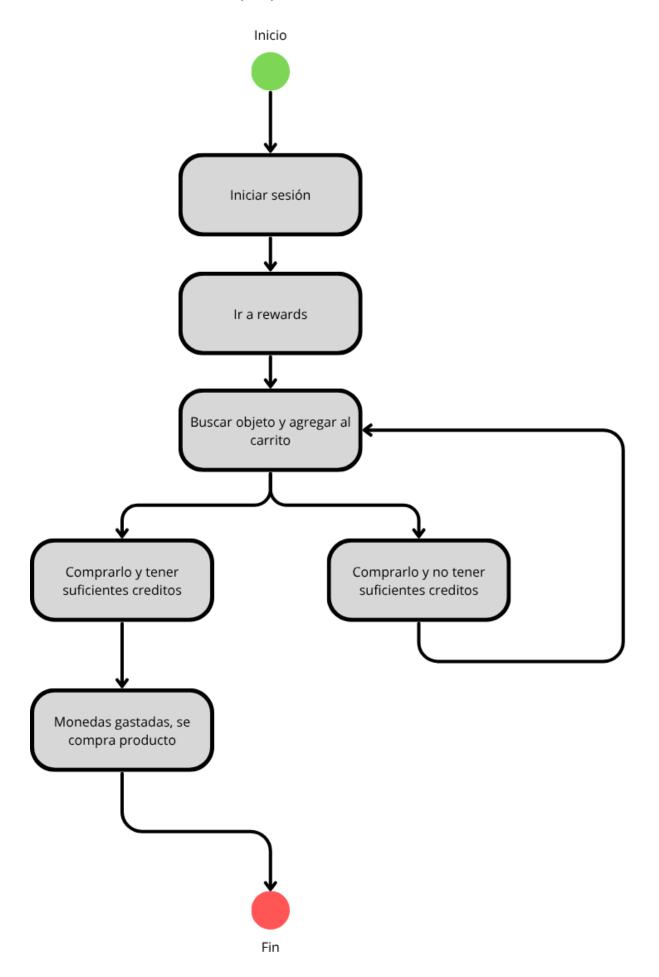
Diagramas de componentes



Diagramas de estados

https://www.canva.com/design/DAGpV8elthM/OeX0T2MWq-tjjOHFxAEVIQ/edit?utm_content=DAGpV8elthM&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton





Matriz de estados/eventos

Matriz de estados del usuario

Estado actual	Evento	Nuevo estado	Acción realizada
No autenticado	Usuario hace login	Autenticado	Se carga el dashboard
Autenticado	Elige un problema	Viendo problema	Se muestra la descripción y el editor
Viendo problema	Envía solución	Evaluando código	Se guarda el envío y se envía a Judge0
Evaluando código	Solución correcta	Problema resuelto	Se asignan puntos, XP y se actualiza perfil
Evaluando código	Solución incorrecta	Viendo problema	Se muestra mensaje de error
Problema resuelto	Selecciona otro problema	Viendo problema	Muestra nuevo reto

Matriz de estados de un problema

Estado actual	Evento	Nuevo estado	Acción realizada
Disponible	Usuario lo abre	En revisión	Se visualiza contenido del problema
En revisión	Usuario envía solución	En evaluación	Se genera una entrega y se envía a API
En evaluación	Judge0 devuelve correcto	Resuelto	Se marca como resuelto y suma XP
En evaluación	Judge0 devuelve error	En revisión	Permite reenviar la solución

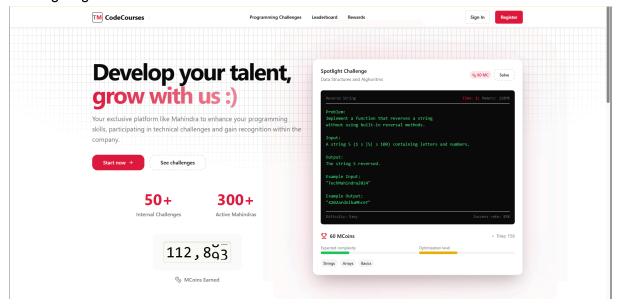
Matriz de estados de una recompensa

Estado actual	Evento	Nuevo estado	Acción realizada
Disponible	Usuario selecciona recompensa	En proceso	Se valida disponibilidad y puntos

En proceso	Inventario suficiente	Canjeado	Se resta inventario y puntos
En proceso	Inventario insuficiente	Agotada	Se muestra mensaje de error
Agotada	Reposición manual del admin	Disponible	Se actualiza inventario

Interfaces de usuario

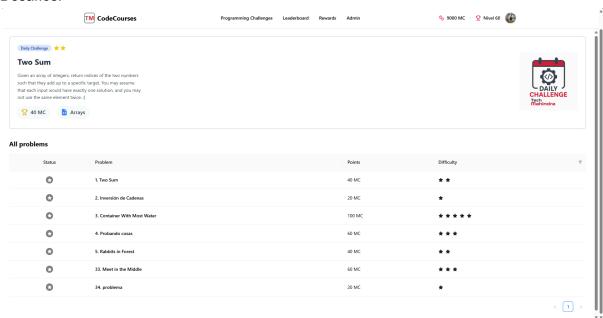
Landing Page



Register/Sign in

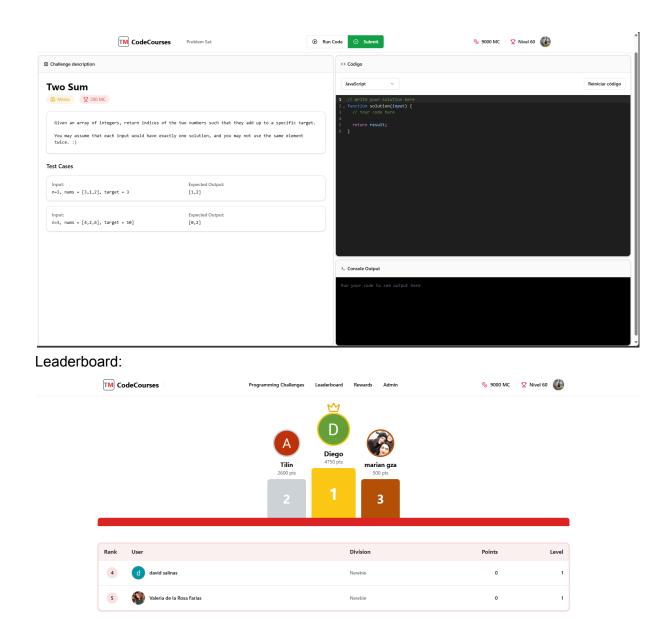


Desafíos:

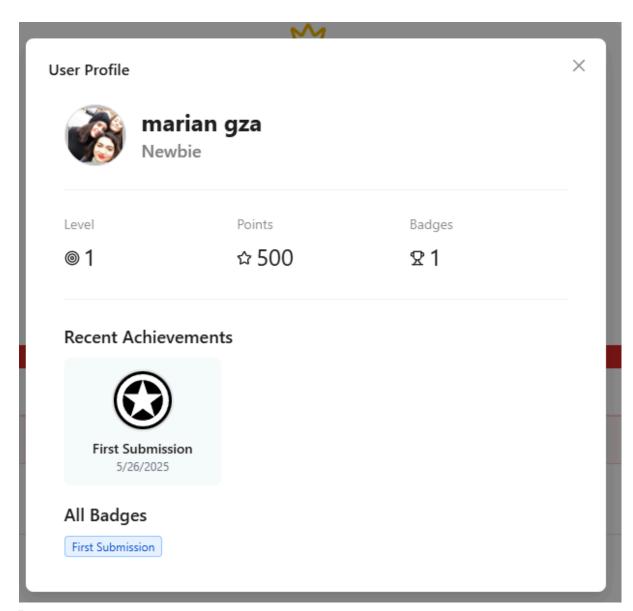


Desafío:

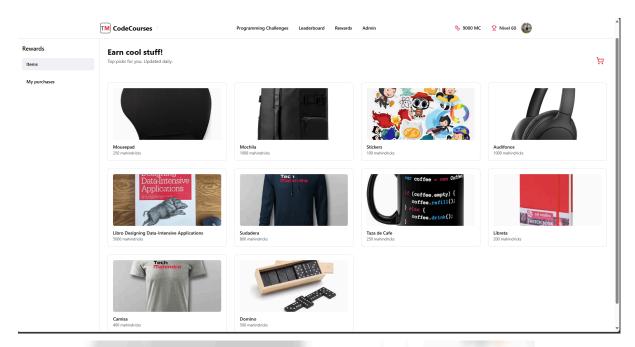
13

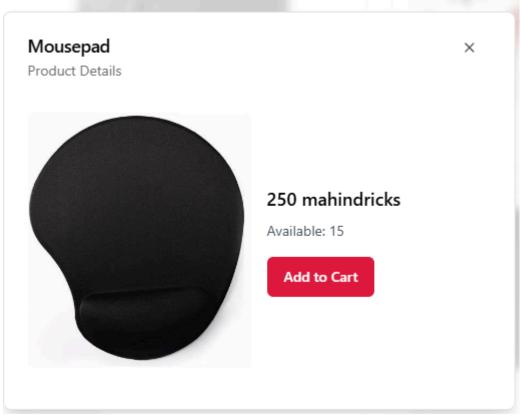


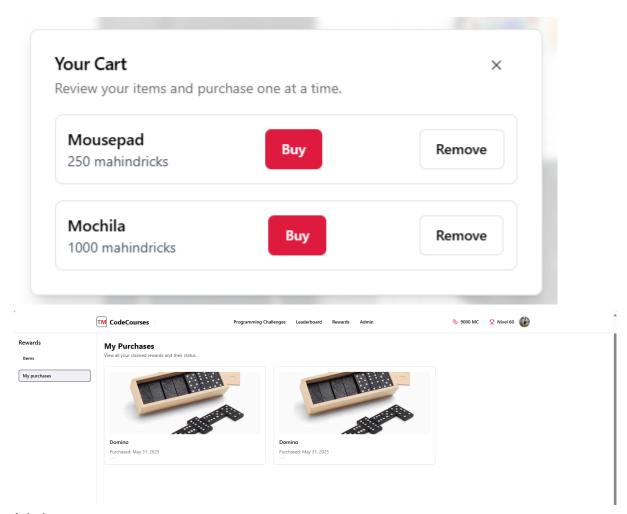
Mini Perfil de Usuario:



Recompensas:







Admin:

