Simulador de Examen Teórico para Licencia de Conducir

Universidad Autónoma de Coahuila

Facultad de Sistemas

Materia: Simulación

Docente: David Pérez Tinoco

Alumno: Roberto Yeshua Moreno Pedraza

Fecha: 23 de Abril 2025

# Introducción

El presente proyecto consiste en el desarrollo de un simulador de examen teórico de manejo, que permite a los usuarios practicar con preguntas reales para obtener su licencia. El sistema simula tanto exámenes prácticos como exámenes finales bajo condiciones reales, incluyendo tiempo límite, puntaje y almacenamiento de historial.

# Objetivos

• Crear una aplicación de escritorio interactiva para practicar exámenes teóricos de manejo.

• Permitir el registro y autenticación de usuarios.

• Controlar intentos del examen final.

• Mostrar resultados y estadísticas mediante un dashboard visual.

• Usar arquitectura estructurada y patrones de diseño.

# Tecnologías utilizadas

|  |  |
| --- | --- |
| Tecnología | Uso |
| Python 3.11 | Lenguaje de programación principal |
| Tkinter | Interfaz gráfica de usuario |
| MySQL | Base de datos |
| Navicat | Gestión de la base de datos |
| Matplotlib | Gráficas para el dashboard |
| Pandas | Manipulación de datos para estadísticas |
| Visual Studio Code | Entorno de desarrollo |
| Git/GitHub | Control de versiones |

#  Login y registro

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

#  Examen de práctica en ejecución

Interfaz de usuario gráfica, Aplicación, Tabla

El contenido generado por IA puede ser incorrecto.

#  Examen final

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

#  Dashboard

Excel

El contenido generado por IA puede ser incorrecto.

Gráfico

El contenido generado por IA puede ser incorrecto.

#  Base de datos (Navicat o consola)

Diagrama

El contenido generado por IA puede ser incorrecto.

#  Terminal con git push exitoso

Texto

El contenido generado por IA puede ser incorrecto.

#  Vista del repositorio en GitHub

# Pantalla de un video juego El contenido generado por IA puede ser incorrecto.

# Estructura del sistema

• main.py: Ventana principal y navegación

• simulador.py: Funciones del backend (consultas, registro, puntaje)

• Base de datos: Tablas usuarios, preguntas, opciones, examenes, respuestas, configuraciones

• dashboard: Generación de gráficas con estadísticas

# Funcionalidades principales

• Registro e inicio de sesión

• Simulador de examen de práctica (20 preguntas, sin límite de intentos)

• Simulador de examen final (40 preguntas, máximo 3 intentos)

• Temporizador por pregunta (60 segundos)

• Almacenamiento de respuestas y puntaje

• Dashboard con gráficas filtradas por práctica/final

• Historial de intentos visibles por usuario

• Opción para cerrar sesión y regresar al login

# Consideraciones técnicas

• Se usa 'after()' de Tkinter para simular el cronómetro.

• Las preguntas se cargan aleatoriamente desde la base de datos (ORDER BY RAND()).

• Los resultados se guardan en tiempo real y se pueden filtrar por tipo de examen.

• Las gráficas del dashboard usan Tkinter y Matplotlib integrados al frontend.

• Seguridad básica aplicada con control de intentos y validaciones.

Control de versiones y colaboración

El proyecto se encuentra alojado en GitHub en el repositorio:

https://github.com/Robert022-hub/simulador-licencia

Se realizaron commits regulares para llevar el control de avances. Se usó la rama principal `main` para mantener la versión estable.

# Conclusión

El simulador cumple con los objetivos planteados, simulando fielmente las condiciones reales de un examen de conducir. Además, incluye componentes visuales, históricos y una arquitectura modular que facilita su mantenimiento y escalabilidad.