



DUOLINGO DATABASE POSTER

BRAYAN ESTIVEN AGUIRRE ARISTIZABAL¹ 20231020156
MARLON YECID RIVEROS GUIO² 20231020208



UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

INTRODUCTION

Language learning platforms demand robust systems capable of managing large volumes of data quickly and reliably. Inspired by the popular Duolingo platform, this project addresses the challenge of designing a database model that efficiently integrates users, courses, lessons, achievements, and progress tracking. The model tackles critical issues such as the conversion of many-to-many relationships into one-to-many associations and ensures scalability and consistency through modern deployment practices.

GOAL

- Develop and implement a robust, scalable database model that:
- Efficiently manages interactions among users, courses, and educational content.
 - Answers the research question: How can we structure a data architecture that supports continuous evolution and growth in language learning platforms?
 - The expected outcome is an integrated system validated both theoretically (with relational algebra) and practically through full CRUD operations.

PROPOSED SOLUTION

The solution is based on an Entity-Relationship (ER) model that defines key entities (User, Course, Lesson, Achievement, and Progress) and their interrelationships. And implement the model with Sql and python



RESULTS

- Theoretical Validation: The database model successfully supports complex queries using relational algebra, confirming its robustness.
- CRUD Operations: The system efficiently performs create, read, update, and delete operations without compromising data integrity.
- Standardized Deployment: Docker ensures a controlled, reproducible environment that facilitates smooth transitions from development to production.
- Performance Comparison: When compared to previous solutions, this model excels in modularity, scalability, and efficient data handling.

CONCLUSION

This project demonstrates that integrating theoretical concepts (ER modeling and relational algebra) with modern tools (Docker, Python, DAO, and Faker) results in a robust and adaptable database architecture for language learning platforms. The clear separation of concerns and practical validation confirm that the system not only meets current requirements but is also well-prepared for future enhancements and technological challenges.

BIBLIOGRAPHY

HOME." DOCKER DOCUMENTATION. ACCESSED: FEB. 5, 2025. [ONLINE]. AVAILABLE: [HTTPS://DOCS.DOCKER.COM/](https://docs.docker.com/)

"FAKER DOCUMENTATION." WELCOME TO FAKER'S DOCUMENTATION! — FAKER 36.1.1 DOCUMENTATION. ACCESSED: FEB. 5, 2025. [ONLINE]. AVAILABLE: [HTTPS://FAKER.READTHEDOCS.IO/EN/MASTER/](https://faker.readthedocs.io/en/master/)

"PSYCOPG2." PYPI. ACCESSED: FEB. 5, 2025. [ONLINE]. AVAILABLE: [HTTPS://PYPI.ORG/PROJECT/PSYCOPG2/](https://pypi.org/project/psycopg2/)

