SQL MENTOR

AI-Powered SQL Educational Platform

presented by:

Eyad Hassan Bargouth

Ahmad Refaat Kitaz

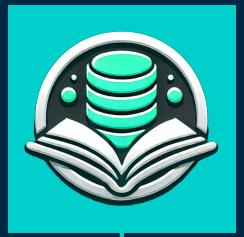
Muhammad Zaid Al Nahhas

Supervision of:

Eng. Nisreen Al-Madi

INTRODUCTION

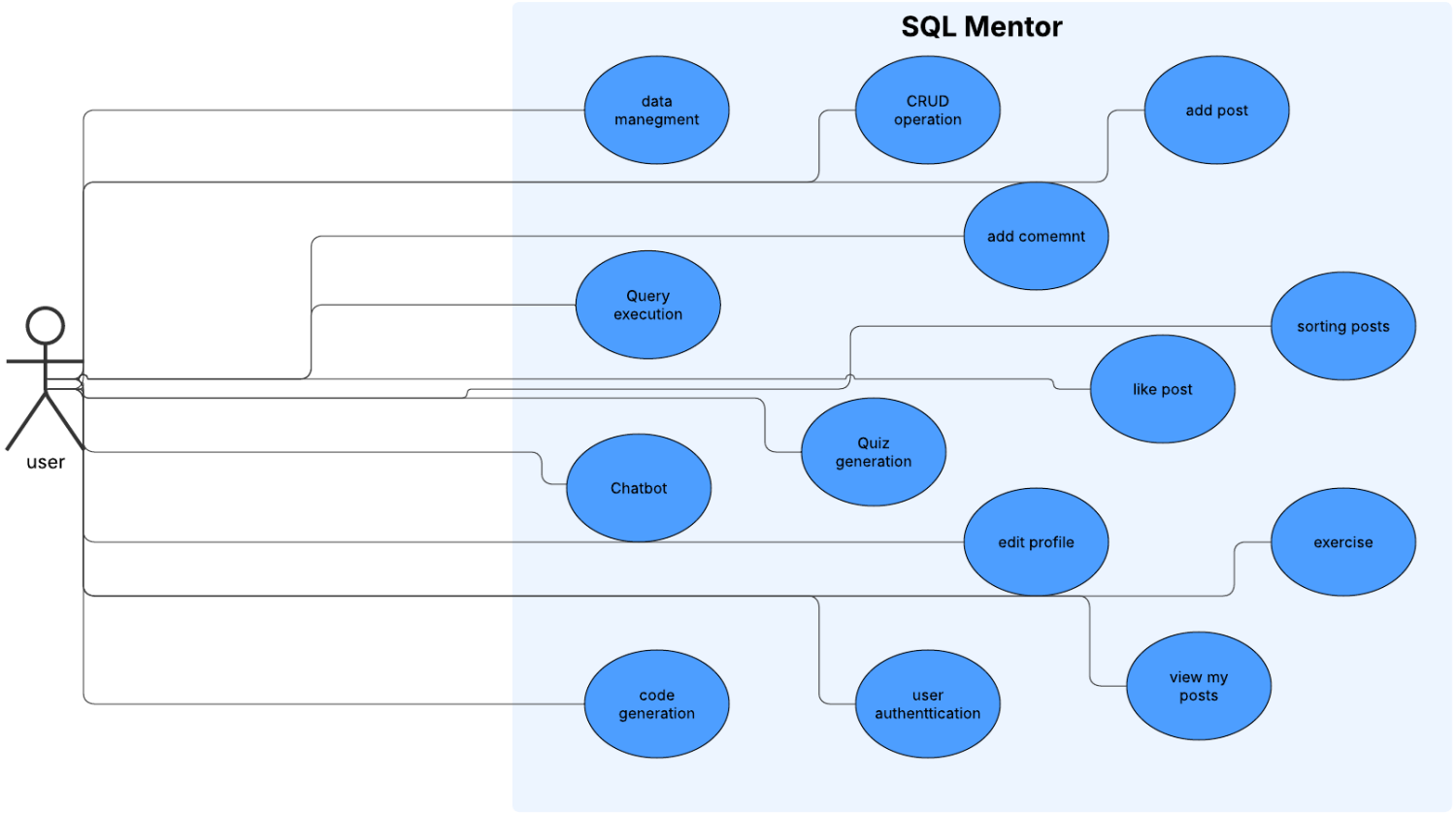
Students face a gap between the theoretical concepts of SQL and its practical projects.



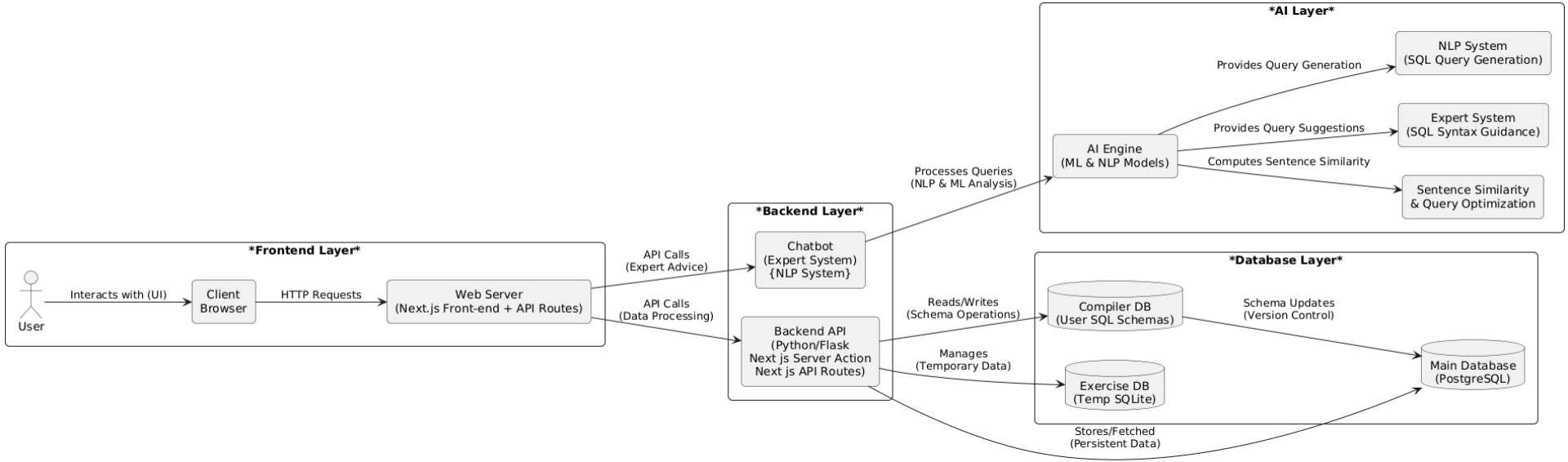
AIM

SQL MENTOR aims to bridge this gap with three main pillars a mentor AI, adaptive quizzes, and a special compiler.

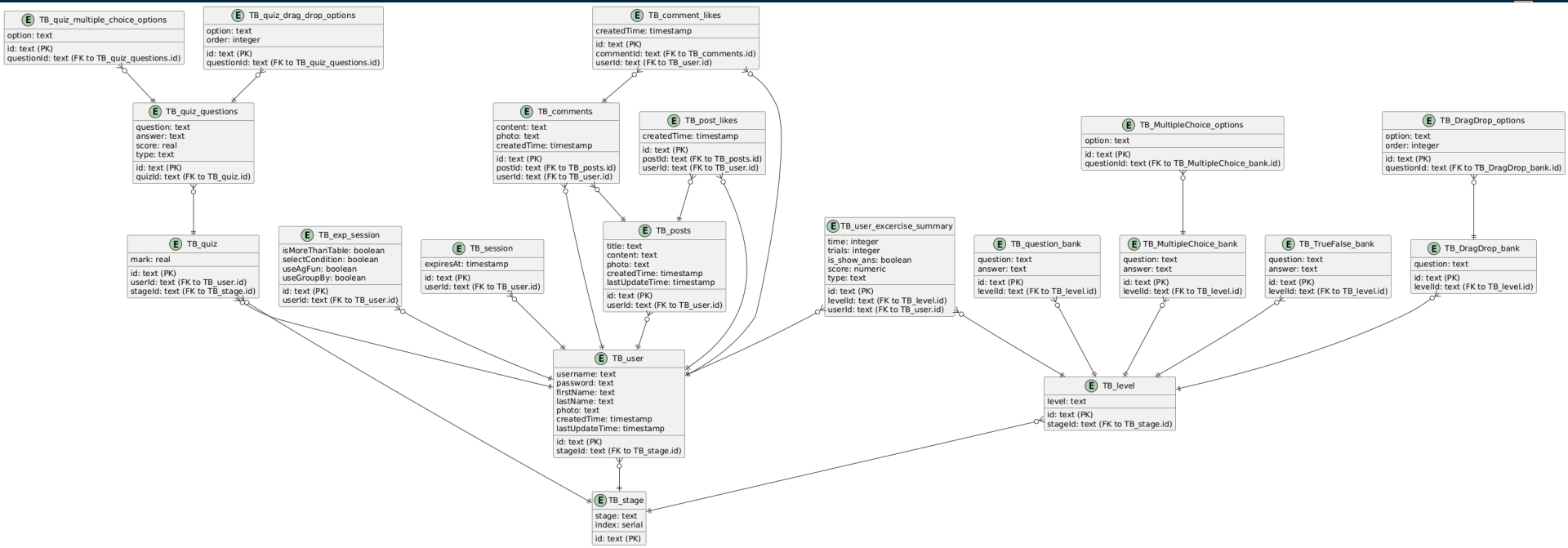
Use Case


SQL Mentor
user
data managment
CRUD operation
add post
add comemnt
Query execution
sorting posts
like post
Quiz generation
Chatbot
edit profile
exercise
view my posts
code generation
user authenttication

Block diagram



ERD


TB_quiz_multiple_choice_options
option: text
id: text (PK)
questionId: text (FK to TB_quiz_questions.id)
TB_quiz_drag_drop_options
option: text
order: integer
id: text (PK)
questionId: text (FK to TB_quiz_questions.id)
TB_comment_likes
createdTime: timestamp
id: text (PK)
commentId: text (FK to TB_comments.id)
userId: text (FK to TB_user.id)
TB_quiz_questions
question: text
answer: text
score: real
type: text
id: text (PK)
quizId: text (FK to TB_quiz.id)
TB_comments
content: text
photo: text
createdTime: timestamp
id: text (PK)
postId: text (FK to TB_posts.id)
userId: text (FK to TB_user.id)
TB_post_likes
createdTime: timestamp
id: text (PK)
postId: text (FK to TB_posts.id)
userId: text (FK to TB_user.id)
TB_MultipleChoice_options
option: text
id: text (PK)
questionId: text (FK to TB_MultipleChoice_bank.id)
TB_DragDrop_options
option: text
order: integer
id: text (PK)
questionId: text (FK to TB_DragDrop_bank.id)
TB_quiz
mark: real
id: text (PK)
userId: text (FK to TB_user.id)
stageId: text (FK to TB_stage.id)
TB_exp_session
isMoreThanTable: boolean
selectCondition: boolean
useAgFun: boolean
useGroupBy: boolean
id: text (PK)
userId: text (FK to TB_user.id)
TB_session
expiresAt: timestamp
id: text (PK)
userId: text (FK to TB_user.id)
TB_posts
title: text
content: text
photo: text
createdTime: timestamp
lastUpdateTime: timestamp
id: text (PK)
userId: text (FK to TB_user.id)
TB_user_excercise_summary
time: integer
trials: integer
is_show_ans: boolean
score: numeric
type: text
id: text (PK)
levelId: text (FK to TB_level.id)
userId: text (FK to TB_user.id)
TB_question_bank
question: text
answer: text
id: text (PK)
levelId: text (FK to TB_level.id)
TB_MultipleChoice_bank
question: text
answer: text
id: text (PK)
levelId: text (FK to TB_level.id)
TB_TrueFalse_bank
question: text
answer: text
id: text (PK)
levelId: text (FK to TB_level.id)
TB_DragDrop_bank
question: text
id: text (PK)
levelId: text (FK to TB_level.id)
TB_user
username: text
password: text
firstName: text
lastName: text
photo: text
createdTime: timestamp
lastUpdateTime: timestamp
id: text (PK)
stageId: text (FK to TB_stage.id)
TB_level
level: text
id: text (PK)
stageId: text (FK to TB_stage.id)
TB_stage
stage: text
index: serial
id: text (PK)

System Requirements Specifications

Functional Requirements:

- User Authentication
- CRUD Operations
- Data Management
- Query Execution
- Quiz Generation
- Chatbot EXP/ Code Generation
- Post Management
- Exercise Management

System Requirements Specifications

Non-Functional Requirements:

- Performance
- Scalability
- Usability
- Security
- Reliability
- Maintainability
- Compatibility

SIMILAR APPLICATION

HackerRank
wide range of SQL challenge & exercises.

coddy.tech
covering diverse theoretical topics

SQLTUTORIAL
guide for beginners and intermediate

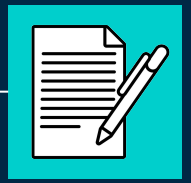
W3Schools
step-by-step to help users learn SQL

Sql Zoo
helps users practice SQL queries

COMPARISON

	Sql mentor	Hacker Rank	Coddy .tech	SQL Tototial	Sql zoo	W3Schools
Chatbot	✓	✗	✗	✗	✗	✗
Quiz generation	✓	✗	✗	✗	✗	✓
DB editor	✓	✗	✗	✗	✗	✗
Score calculation	✓	✗	✗	✗	✗	✗
Real output	✓	✓	✗	✓	✓	✓
Real examples	✓	✓	✓	✓	✗	✓

Algorithms Sections



01

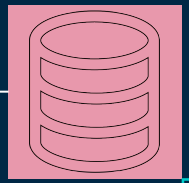
QUIZ GENERATION

To generate questions appropriate for the student

02

COMMUNITY

To share questions with people

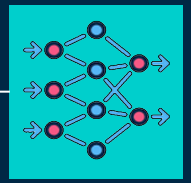


03

SQL COMPILER

To create a private database and execute instructions

Artificial Intelligence Sections



01

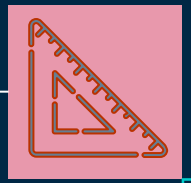
NEURAL NETWORK

Text to SQL
Query generation

02

EXPERT SYSTEM

Syntax generation
Chat bot



03

COSINE SIMILARITY

Score
calculator

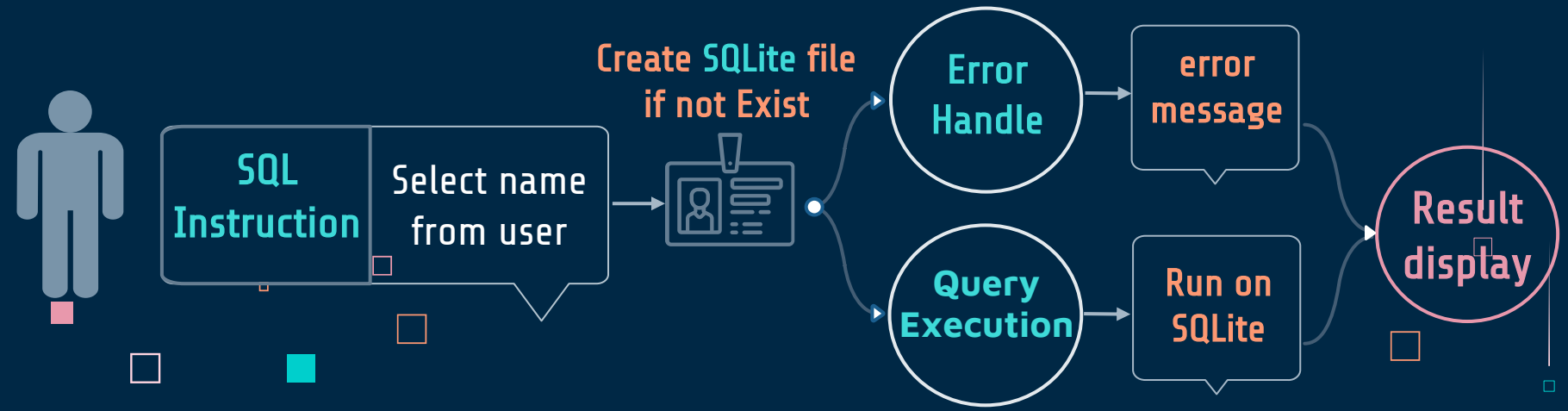
Compiler

SQLMENTOR allows execute queries with instant feedback errors. helps learners practice SQL .

01

Compiler

Executes SQL queries within the platform, providing real-time feedback on syntax and logic errors.



Result

Executes SQL queries, providing immediate feedback. but performance may be affected by complex queries.

Query Generation

It helps uses create accurate SQL queries based on natural language input.

02

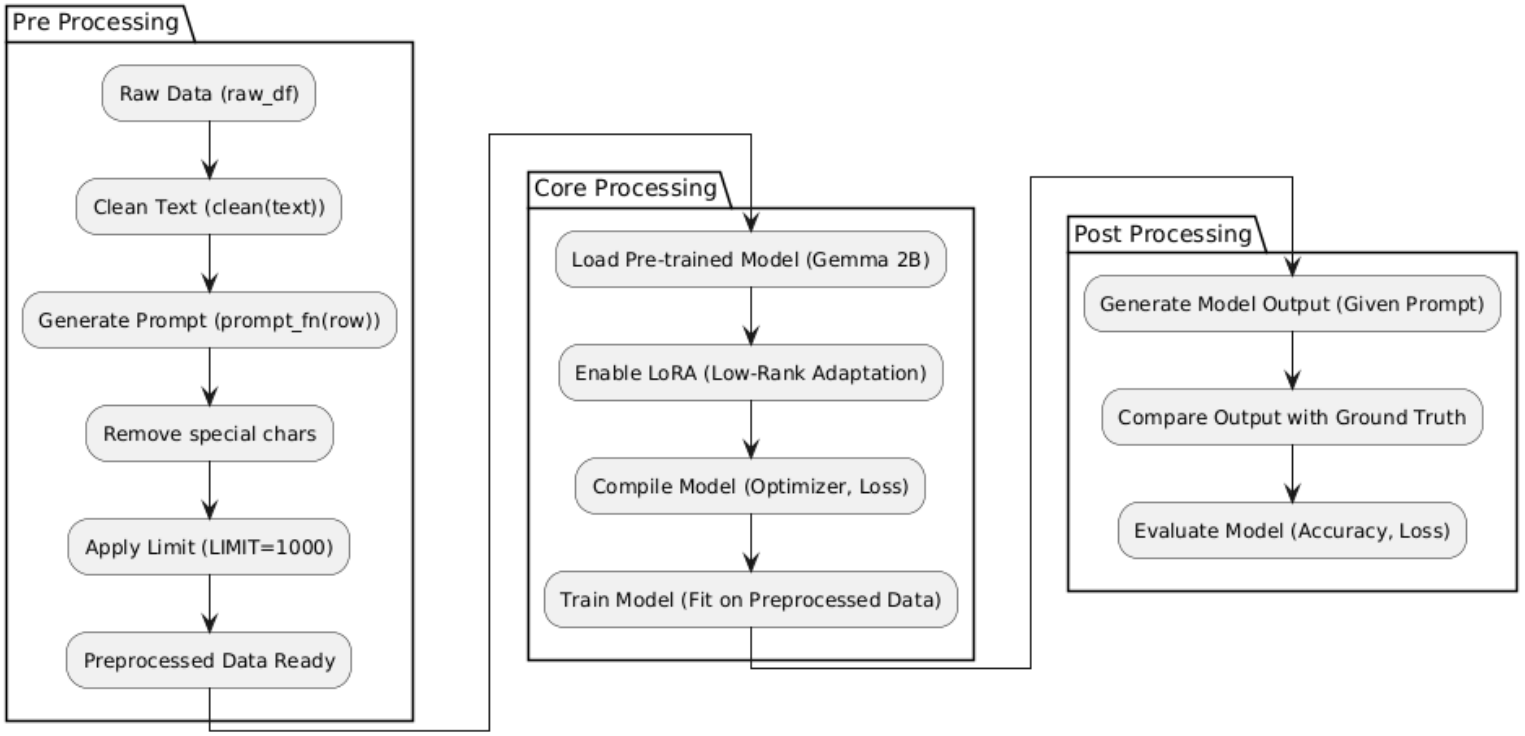
Related work for Query Generation

Reference	Year	Data	Preprocessing	Model	Result
(Khadija & Mustapha, 2024)	2024	Spider		RAT-SQL	62.7
				Bertrand-DR	57.9
				Photon	63.2
				Athena++	87.82
		M-SQL		TableQA	92.1
		IE-SQL		WikiSQL	94.2
(Dinu, Mihăescu, & Rebedea, 2023)	2023	Spider	Schema encoding and linking	RAT-SQL	80
				RoBERTa	Improved performance
(Wong, et al., 2024)	2024	spider		T5	73
				Graphix-T5	74
				RASAT	72.6

Related work for Query Generation

Reference	Year	Data	Preprocessing	Model	Result
(Song, Wong, Zhao, & Jiang, 2024)	2024	WikiSQL	Extracted 96 log-scaled Mel-band energies from speech	CNN (Speech Encoder)	54.15
				GNN (Schema Encoder)	Improves multi-table query accuracy by 0.69% over vanilla RNN.
		Spider	Converted database schema into graphs with nodes and edges	Transformer (Relation-aware Encoder)	18.5
				LSTM (SQL-aware Decoder)	Improves SemQL grammar decoding performance by 28.35% over vanilla RNN decoder.

Query Generation Block Diagram



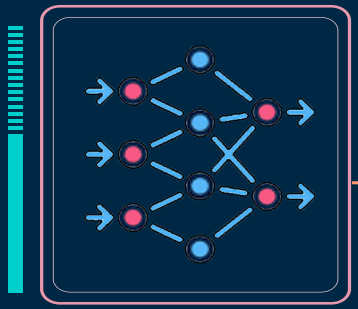
DataSet

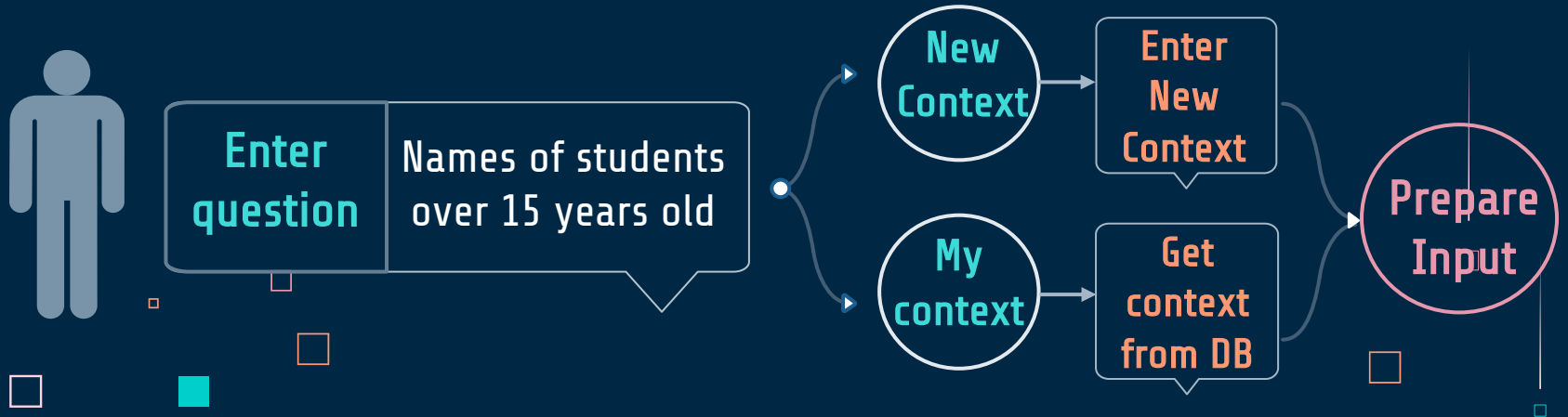
The dataset underwent preprocessing, including cleaning missing/incorrect values, normalizing text formats, structuring SQL queries, tokenization, and encoding to optimize it for accurate SQL generation during model training.

Question	Context	Answer
Names of students over 15 years old	Create table student (name TEXT , Age Number);	Select name from student where age > 15
what's the result with district being louisiana 2	CREATE TABLE x (result VARCHAR, district VARCHAR)	SELECT result FROM x WHERE district = "Louisiana 2"
who are the candidates with first elected being 1977	CREATE TABLE x (candidates VARCHAR, first-elected VARCHAR)	SELECT candidates FROM x WHERE first-elected = 1977

Query Generation

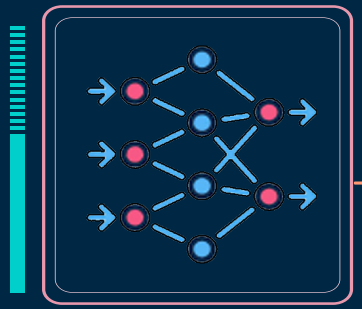
- **Data Input**:
  - Extract user's natural language query + database schema (table context).

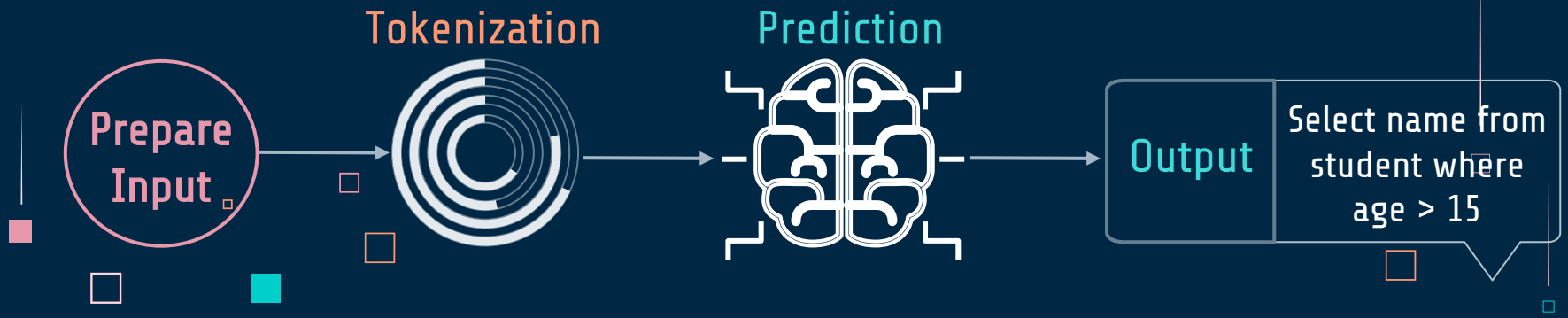




Query Generation

- **Fine-tuned Model**: LoRA-adapted Gemma 2B.
- **Model Processing**:
  - Tokenization & semantic analysis.
  - Entity recognition for tables/columns.



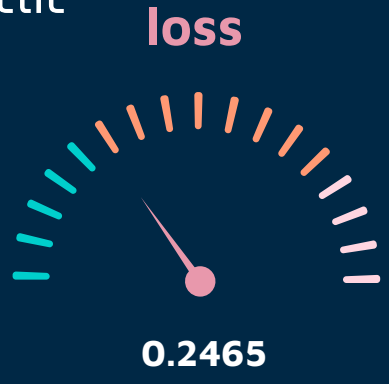


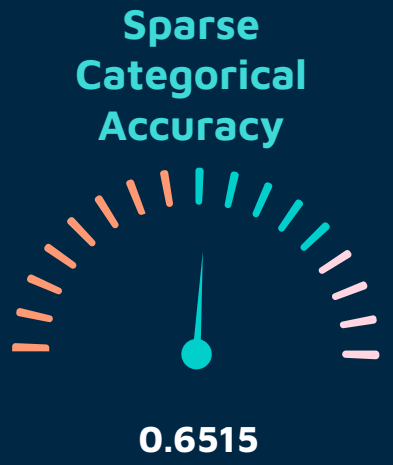
Query Generation

Results:

- The model achieved **0.6515 SparseCategoricalAccuracy** and **0.2465 loss** in SQL query generation.

- It successfully extracted table context and relevant columns, ensuring syntactic correctness.



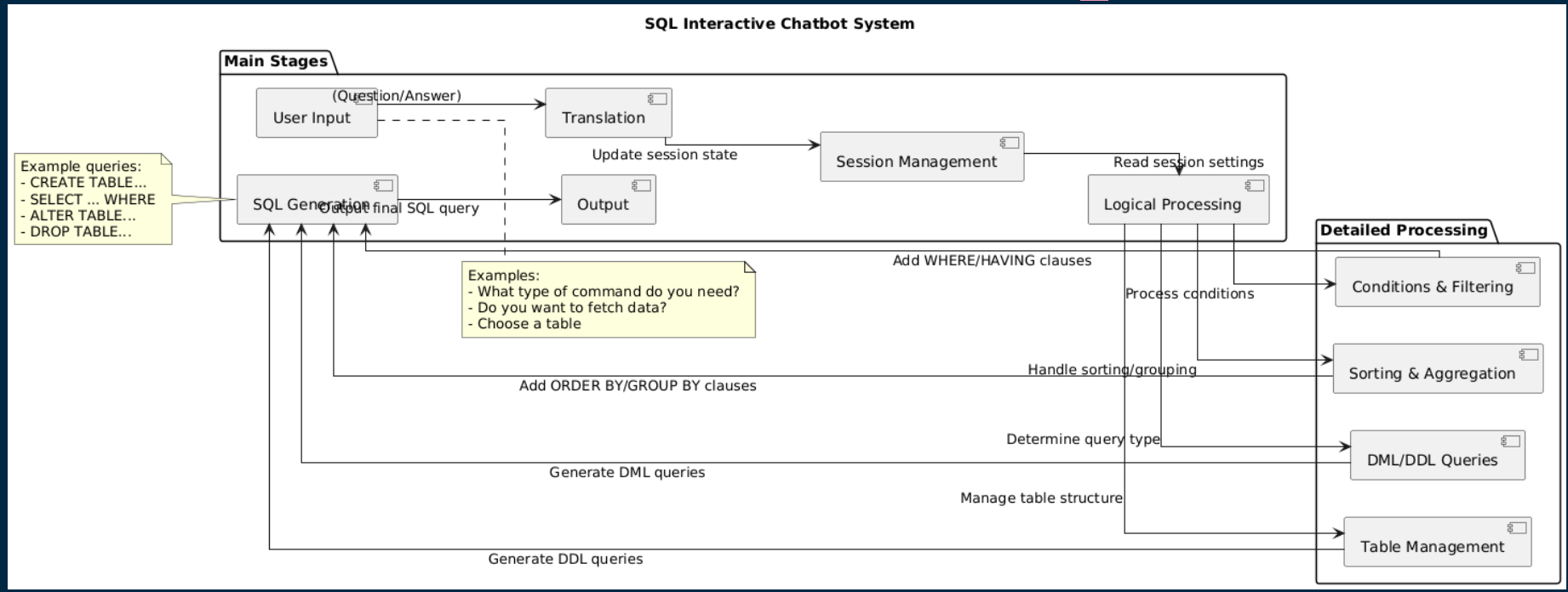


Chatbot Syntax

03

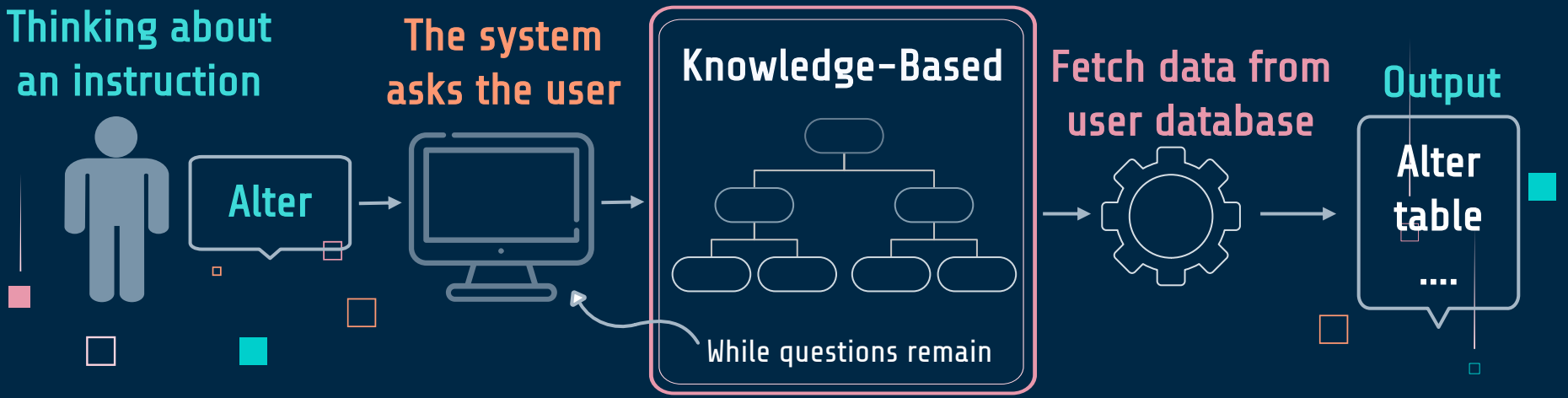
provides users with correct SQL syntax for various commands.

Expert Diagram



Chatbot Syntax

- Interactive Knowledge-Based System for generating SQL queries.
- Supports DML & DDL.



Result & Discussion

- Users can generate **valid SQL queries** for both **DML and DDL commands**.
- The system handled **edge cases** by prompting users for clarification on missing or incorrect inputs.
- The **interactive dialogue** helped beginners learn SQL step-by-step without prior syntax knowledge.

Score Calculator

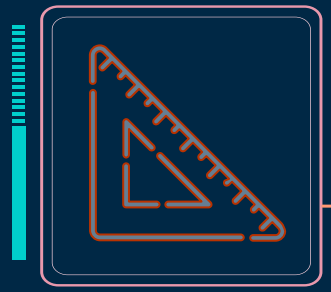
04

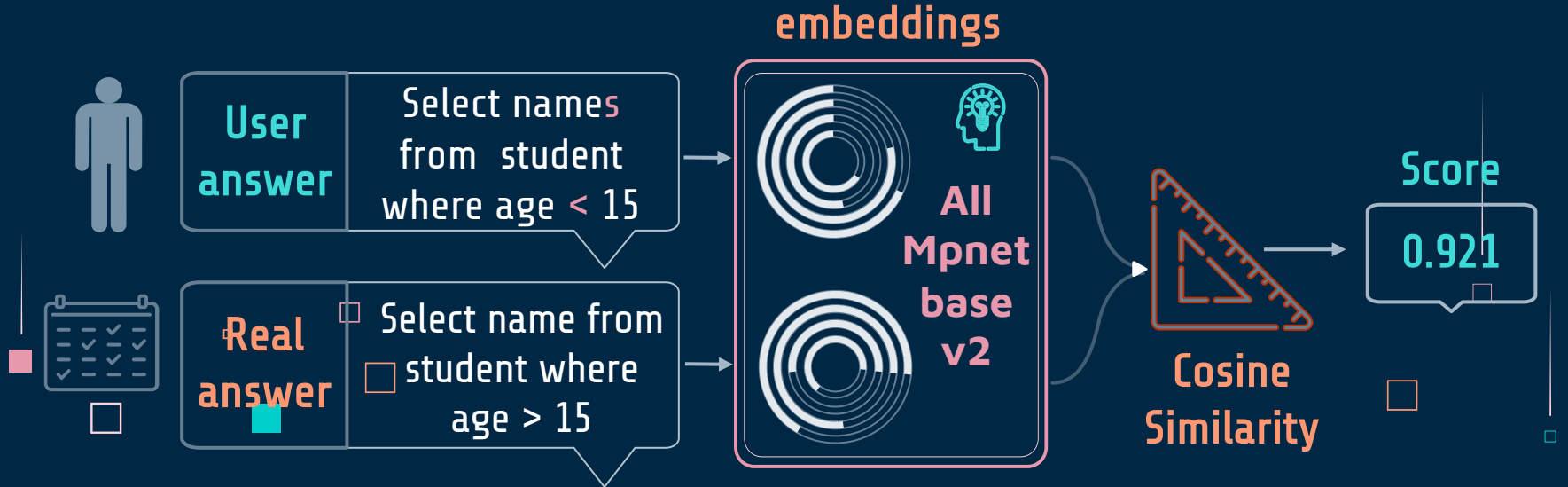
evaluates users' SQL queries by comparing them to correct answers.

Score Calculator

Computes similarity between two sentences using Cosine Similarity.

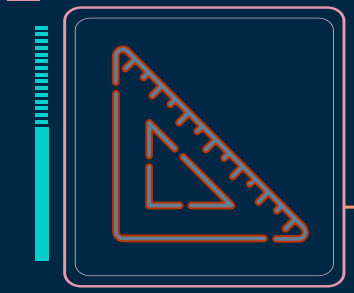
Evaluates user responses in quizzes & interactive exercises.





Result & Discussion

- The model evaluates SQL answers well but struggles with semantically similar, syntactically different queries. Adjusting thresholds improves classification accuracy.



Quiz Generation

05

generates dynamic quizzes tailored to users' levels.

Quiz Generation

Creates interactive SQL quizzes to assess users' knowledge and reinforce learning by adapting to their performance.

Use Cases

- Adaptive Quiz Generation.
- Performance Evaluation.
- Balanced Question Distribution.
- Answer Evaluation.
- Quiz Scoring.

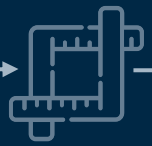
Quiz Generation

Track user performance at each level of exercise at the current stage.

Request a Quiz For the current stage

Time taken
Number of attempts
Is show answer
Exercise score

Normalization



0...1

Questions Calculation

W1 = 0.2

W2 = 0.2

W3 = 1000

W4 = -0.3

output

Number Of question For each level

Quiz Generation

output

Number Of question For each level

Retrieve questions from the question bank and display them to the user

Waiting for the student to finish answering

Calculate the score for each question using the score calculator.

Show quiz result

Result

The system creates a dynamic feedback system based on user performance, tracking attempts and accuracy of answers. Several questions are determined by an increasing formula for personal success.

Community

06

An environment where users can share and discuss SQL topics and questions.

Community

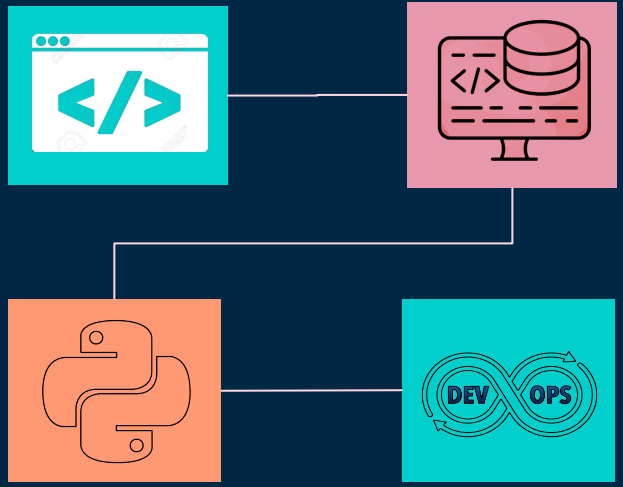
Creates an interactive social environment where users can engage, share, and discuss topics while ensuring a safe experience through AI moderation.

Result

The system Creates an interactive social environment where users can engage, share, and discuss topics while ensuring a safe experience

Implementation

Frontend
Next js
React
TypeScript



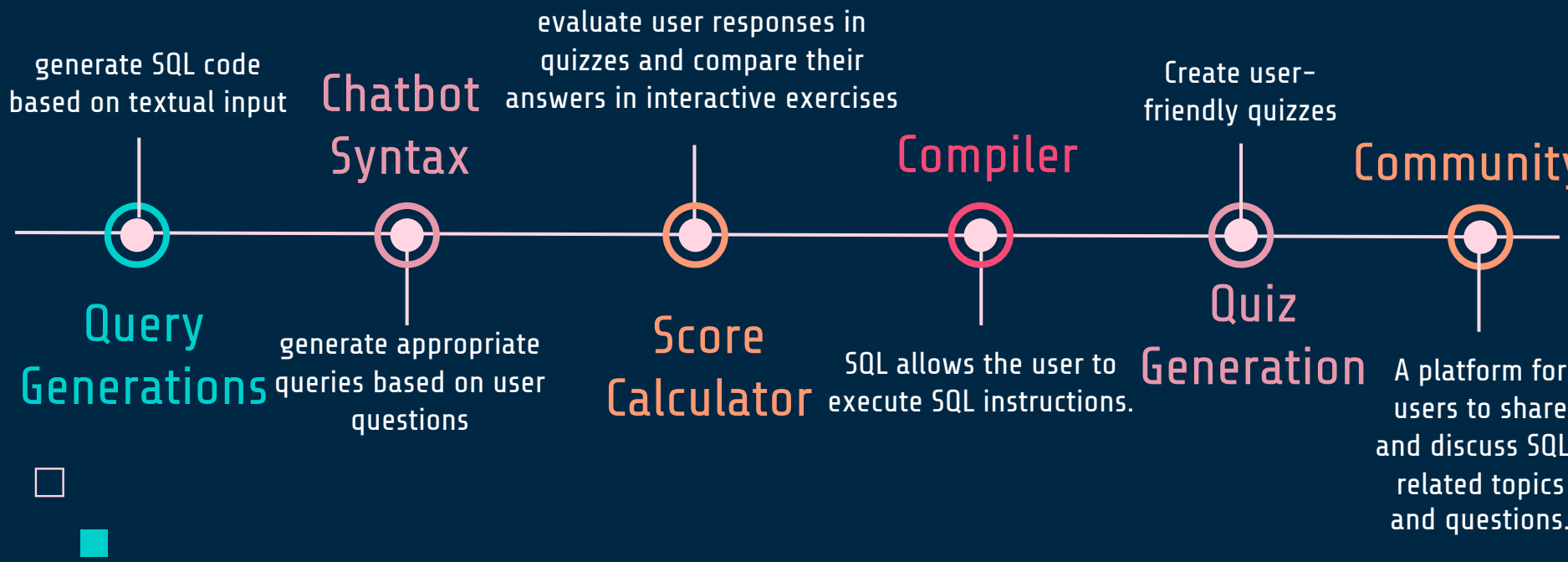
Backend
Next js API Routes
Next js Server Action
Flask API

Python
Jupyter
Colab

DevOps
Docker
GitHub
Supabase

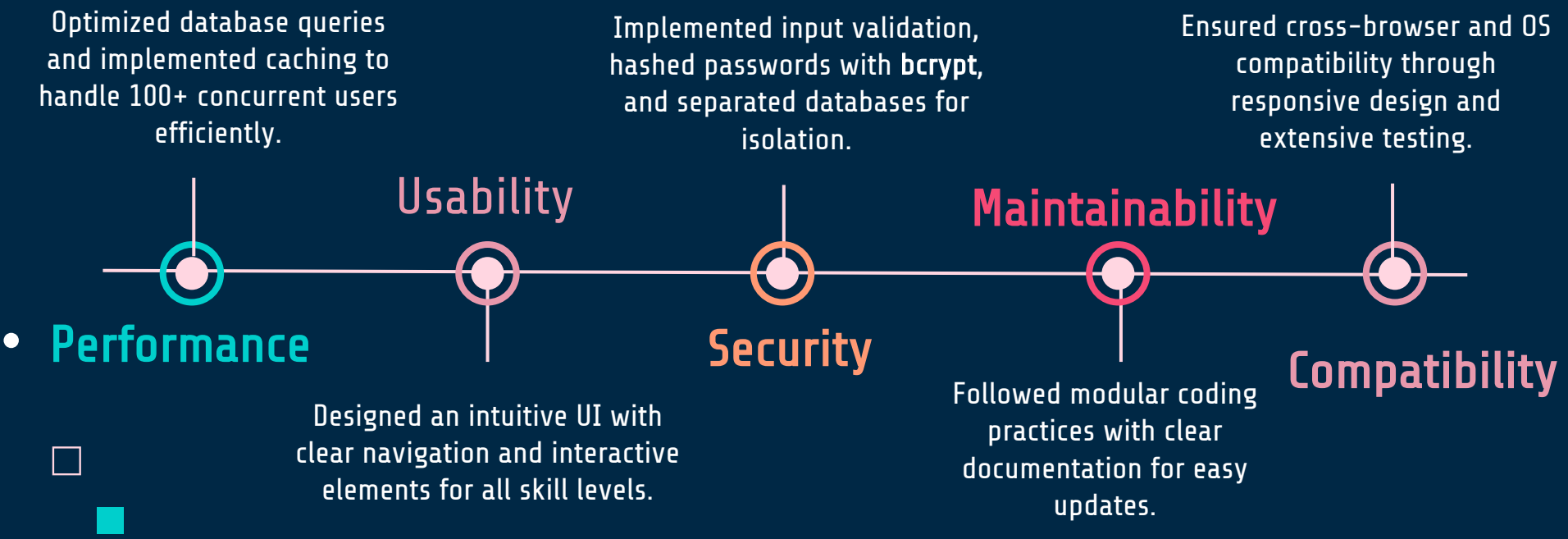
Result

We were able to build an educational platform that includes the following features:



Result

We achieved non-functional requirements by:



Metrics

Visual Appeal

Ease of Use

Performance Speed

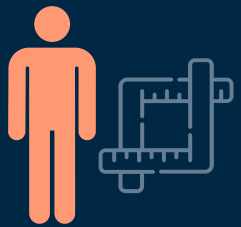
Device Compatibility

Content

Overall Experience

0-9

On scale

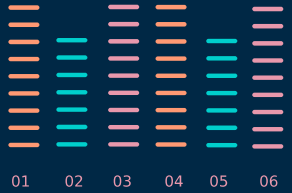


12

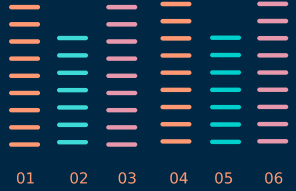
samples

Reviews

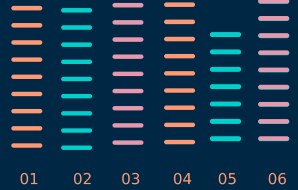
— user 1



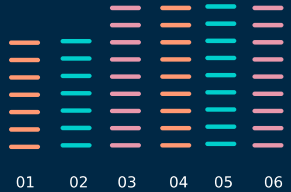
—user 2



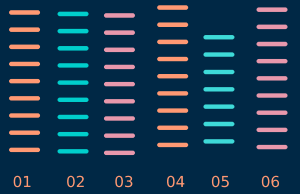
—user 3



—user 4

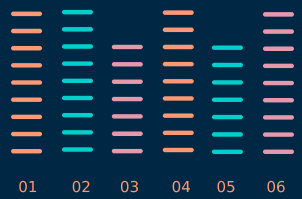


—user 5

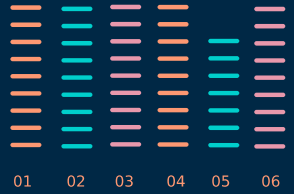


—user 6

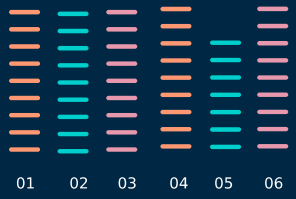
Reviews



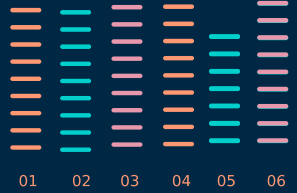
— user 7



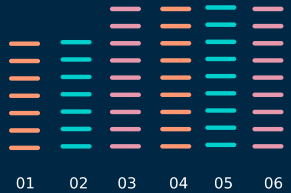
—user 8



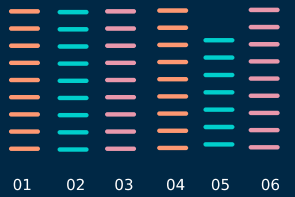
—user 9



—user 10

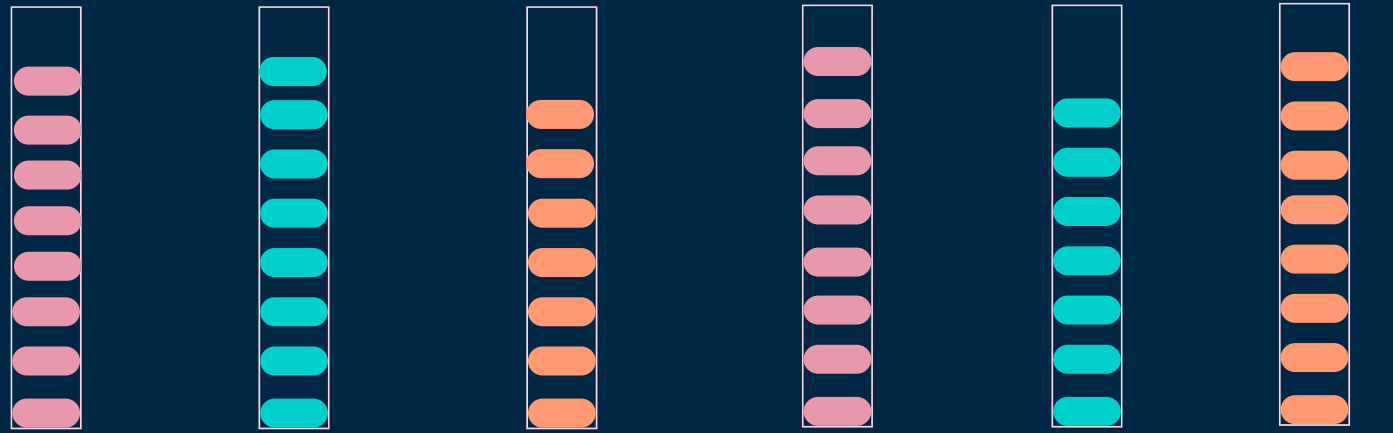


—user 11



—user 12

ANALYSIS



95%
Visual

85%
Easiness

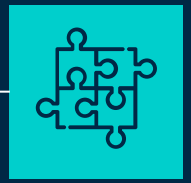
85%
Performance

90%
Compatibility

80%
Content

95%
Overall

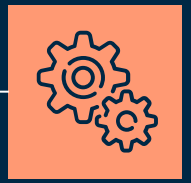
DIFFICULTY



01

USER DATABASE

CREATE NEW DB FOR EACH USER



02

AI Models

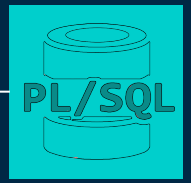
Large pre-trained models require high computational power.

03

Complex Queries

Handling complex SQL queries efficiently

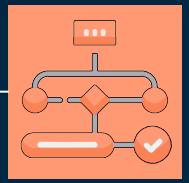
Future work



01

PL/Sql

Support PL/Sql in the platform



02

ERD-to-Code

erd convert to code using artificial intelligence

03

Auto Complete

Auto-complete instructions as you type using AI

Conclusion

SQL Mentor transforms SQL learning with AI-driven tools, interactive practice, and personalized guidance, making mastery efficient and engaging.

THANKS FOR Listening