

System Software Design

Project Draft

202355550

Kang-Kyung Min

Quiz generator

[Purpose]

- To support students' learning by generating quizzes and provide instant feedback
- To make learning more fun by adding game-like elements

[Functional Requirements]

- Use OpenAI API to generate quizzes
- Get user text for the topic of the quiz
- Allow users to specify quiz types and difficulty
- The AI response should be formatted so it can be parsed:

example)

Question: ?

Choices: (for the multiple choice quiz type)

1. ?

2. ?

3. ?

4. ?

Answer: 1

- Generate the UI from the AI response and get the user input
- Compare the user input and the answer, then show the result
- Store the result in a database for future reviews

[Non-Functional Requirements]

- Scalability: Suitable for small number of students (not for big companies)
- Accuracy: Wrong answers/choices by the AI should be minimized
- Response time: waiting time between each part of the program shouldn't be too long

[Components]

Load Balancer

- Handle the traffic efficiently
- Nginx

Servers

- Implement the functional/non-functional requirements
- Node.js

API

- Generate quizzes based on the user input
- OpenAI GPT

Database

- Store quiz results for future results
- PostgreSQL

[High-Level Design]

