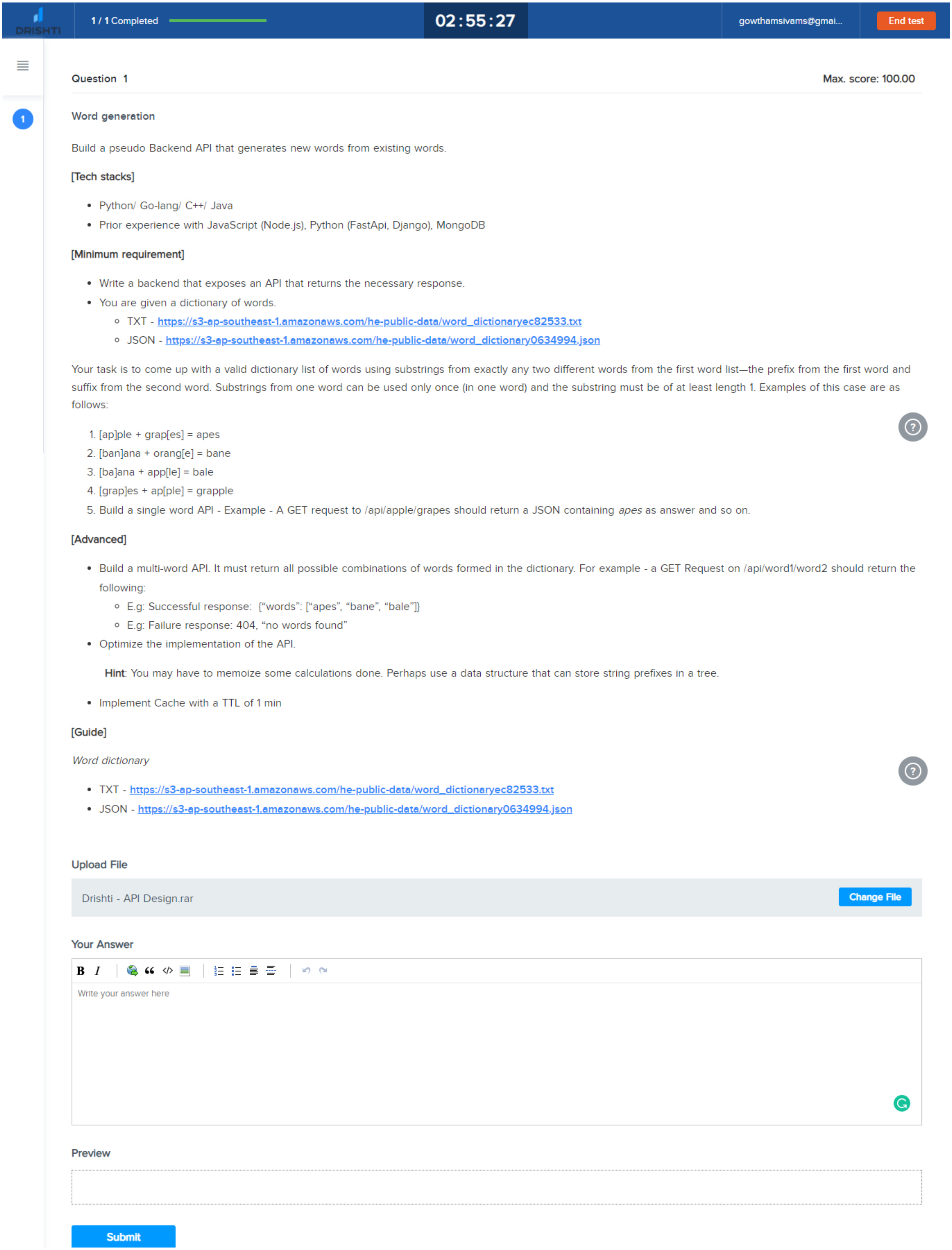

Question 1

Max. score: 100.00

Word generation

Build a pseudo Backend API that generates new words from existing words.

[Tech stacks]

- Python/ Go-lang/ C++/ Java
- Prior experience with JavaScript (Node.js), Python (FastApi, Django), MongoDB

[Minimum requirement]

- Write a backend that exposes an API that returns the necessary response.
- You are given a dictionary of words.
    - TXT - https://s3-ap-southeast-1.amazonaws.com/he-public-data/word_dictionaryec82533.txt
    - JSON - https://s3-ap-southeast-1.amazonaws.com/he-public-data/word_dictionary0634994.json

Your task is to come up with a valid dictionary list of words using substrings from exactly any two different words from the first word list—the prefix from the first word and suffix from the second word. Substrings from one word can be used only once (in one word) and the substring must be of at least length 1. Examples of this case are as follows:

1. [ap]ple + grap[es] = apes
2. [ban]ana + orang[e] = bane
3. [ba]ana + app[le] = bale
4. [grap]es + ap[ple] = grapple
5. Build a single word API - Example - A GET request to /api/apple/grapes should return a JSON containing *apes* as answer and so on.

[Advanced]

- Build a multi-word API. It must return all possible combinations of words formed in the dictionary. For example - a GET Request on /api/word1/word2 should return the following:
    - E.g: Successful response: {"words": ["apes", "bane", "bale"]}
    - E.g: Failure response: 404, "no words found"
- Optimize the implementation of the API.

  Hint: You may have to memoize some calculations done. Perhaps use a data structure that can store string prefixes in a tree.

- Implement Cache with a TTL of 1 min

[Guide]

Word dictionary

- TXT - https://s3-ap-southeast-1.amazonaws.com/he-public-data/word_dictionaryec82533.txt
- JSON - https://s3-ap-southeast-1.amazonaws.com/he-public-data/word_dictionary0634994.json

Upload File

Drishti - API Design.rar

Your Answer

Write your answer here

Preview