

LinguaLearnX

Tomasoni Francesco - 1080980

Homework 3

Lambda Function - Watch Next

Viene sposta come API, dato un id come parametro di input , viene restituito l'id, il titolo e l'url dei watch next corrispondenti

```
connect_to_db().then(() => {
  console.log('=> get_all talks');
  talk.find({ id_ref: body.id }, { related_id : 1})
    .then(talks => {
      if (!talks) {
        throw new Error('nessun talk trovato');
      }
      callback(null, {
        statusCode: 200,
        body: JSON.stringify(talks)
      })
    })
    .catch(err =>
      callback(null, {
        statusCode: err.statusCode || 500,
        headers: { 'Content-Type': 'text/plain' },
        body: 'impossibile caricare i talk.'
      })
    );
});
```

Chiamata API

Risposta

```
{
  "id": "526880"
}
```

Cookies Headers (7) Test Results

Raw

Preview

Visualize

JSON 



```
[
  {
    "_id": "664ee61980fd8a6898e6848a",
    "related_id": [
      "109914",
      "76541",
      "100294"
    ],
    "titles": [
      "Whatever happened to the hole in the ozone layer?",
      "What's in the air you breathe?",
      "Why plague doctors wore beaked masks"
    ]
  },

```

Lambda Function - Generate_Quiz

esposta tramite API gateway.

Riceve in input l'id del talk e controlla che il talk abbia una trascrizione e almeno un tag.

Accede al dataset mongo e recupera il talk tramite id.

Con la libreria "Axios", messa come layer della funzione, viene effettuata una richiesta all'API di PrepAI, passando: il client e il

- token necessario,
- il contenuto, ovvero la trascrizione
- il topic, ovvero i tag
- tipo di domande (categoria 1 sono le crocette), ma ce ne sono altre
- numero di domande massime da generare

Il quiz viene generato e salvato nella collection "tedx_quiz".

```
const transcript = talk.transcript;
const tags = talk.tags;
const id_ref = talk.id_ref;

// Prepare payload for the API request
const payload = {
  topic: tags,
  content: transcript,
  quesType: '1',
  quesCount: '20'
};

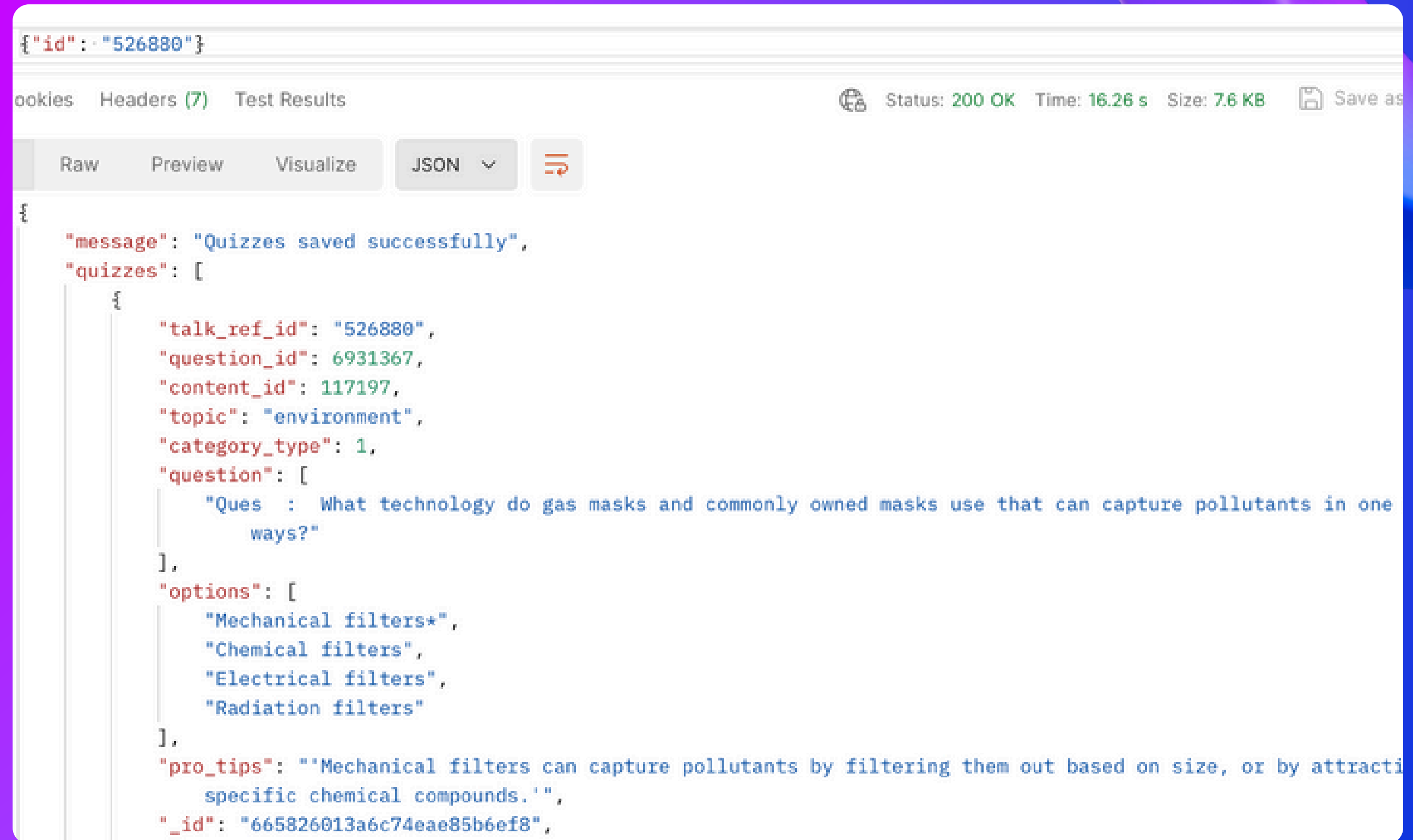
// Prepare headers for the API request
const headers = {
  clientId: '741195bbd0324c1327e31a05c71a252d522c3038',
  clientSecret: 'ZjKidlQqHWQugiqYhxBm4rCI0ybkuhlS_jg8Hwxd'
};

// Make the API request
const querystring = require('querystring');
const formData = querystring.stringify(payload);

axios.post('https://api.prepai.io/generateQuestionsApi', formData, { headers
  .then(response => {
    const questions = response.data.response;
```


Chiamata API

Risposta



The screenshot shows a web browser window displaying an API response. The address bar shows a URL with a query parameter `{ "id": "526880" }`. The browser's developer tools are open, showing the 'Test Results' tab. The response is a JSON object with a status of 200 OK, a time of 16.26 s, and a size of 7.6 KB. The response is displayed in the 'JSON' tab, showing a successful message and a list of quizzes.

```
{
  "message": "Quizzes saved successfully",
  "quizzes": [
    {
      "talk_ref_id": "526880",
      "question_id": 6931367,
      "content_id": 117197,
      "topic": "environment",
      "category_type": 1,
      "question": [
        "Ques : What technology do gas masks and commonly owned masks use that can capture pollutants in one ways?"
      ],
      "options": [
        "Mechanical filters*",
        "Chemical filters",
        "Electrical filters",
        "Radiation filters"
      ],
      "pro_tips": "'Mechanical filters can capture pollutants by filtering them out based on size, or by attracting specific chemical compounds.'",
      "_id": "665826013a6c74eae85b6ef8",
    }
  ]
}
```

Generate_Quiz - Salavatggio su Mongo

i Quiz vengono salvati domanda per domanda (identificata tramite id) con l'id del talk a cui appartengono.

```
_id: ObjectId('6658234af0fd526b4342bef6')
talk_ref_id: "526880"
question_id: 6931306
content_id: 117195
topic: "environment"
category_type: 1
▼ question: Array (1)
  0: "Ques : What is the minimum size of the majority of
▼ options: Array (4)
  0: "1 micron"
  1: "2.5 microns*"
  2: "5 microns"
  3: "10 microns"
pro_tips: "'The majority of wildfire smoke particles are sm
__v: 0
```

Criticità

Tempistiche

La generazione dei quiz potrebbe richiedere abbastanza tempo per testi lunghi

API di PrepAI

L'API di PrepAI offre una prova gratuita di 15 giorni, raggiungibile creando nuovi account, ma non all'infinito.

Questa API genera quiz con domande e risposte, ma al posto di fornire la risposta corretta viene restituito il campo "pro tips" che sarebbe il pezzo di testo da cui è stata estrapolata la domanda, richiedendo quindi una verifica umana.

Possibili evoluzioni

Tempistiche

Si potrebbe pensare di separare la generazione dei quiz con il salvataggio su mongo, in modo da poter gestire le cose in modo asincrono.



API di PrepAI

Per quanto riguarda la verifica, potrebbe essere automatizzata, tramite l'utilizzo di AI come ChatGPT, che fornisce delle API, che renderebbe poi il controllo umano più veloce.

Oppure trovare un'altra API che generi i quiz e fornisca la domanda corretta.

Lambda Function

L'obiettivo principale di questa applicazione è insegnare le lingue tramite i video TEDx. Una volta generati i quiz, bisogna creare una funzione che recuperi le domande e le risposte e generi la traduzione e il significato di ogni parola nella lingua madre dell'utente.



Esplora, Impara, con
TEDx e le lingue!

LinguaLearnX

