AWS Lambda Function

Michele Dentella - 1063244

Get_Watch_Next_by_Idx: dati

Talk.js: oggetto nodeJs dove “mappare” il contenuto ricevuto dalla query su mongoDB

campi importanti:

- _id : String
- watch_next_ids : [String]
  array di stringhe

```
 1 const mongoose = require('mongoose');
 2
 3 const talk_schema = new mongoose.Schema({
 4     _id:String,
 5     title: String,
 6     url: String,
 7     watch_next_ids:[String]
 8 }, { collection: 'tedz_data' });
 9
10 module.exports = mongoose.model('talk', talk_schema);
~
"TCM-lab/laboratorio-TCM/compito3/Talk.js" 10 lines --10%--      1,1           All
```

Get_Watch_Next_by_Idx: handler

essenziale lambda function, basata sulla versione presentata a lezione.

- controllo che il campo non sia nullo
- query per id ricevuto dalla GET restituisco solo l'array dei watch_next

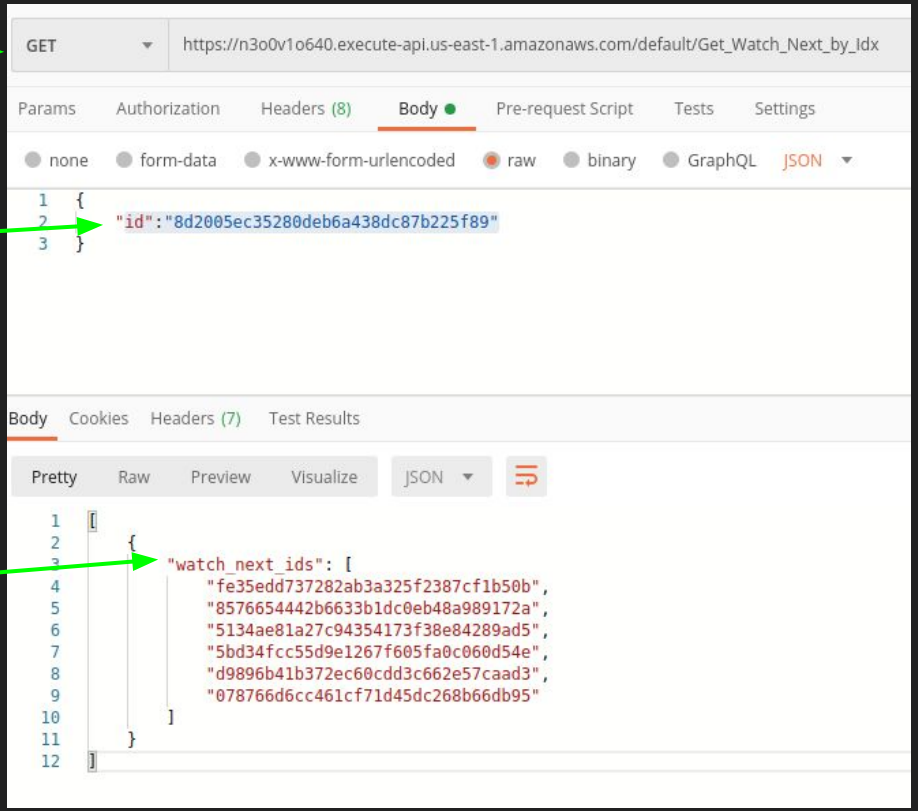
```
1  const connect_to_db = require('./db');
2
3  // GET BY TALK HANDLER
4
5  const talk = require('./Talk');
6
7  module.exports.get_next = (event, context, callback) => {
8      context.callbackWaitsForEmptyEventLoop = false;
9      console.log('Received event:', JSON.stringify(event, null, 2));
10     let body = {}
11     if (event.body) {
12         body = JSON.parse(event.body)
13     }
14     // set default
15     if(!body.id) {
16         callback(null, {
17                     statusCode: 500,
18                     headers: { 'Content-Type': 'text/plain' },
19                     body: 'Could not fetch the talks. Watch_next_ids is null.'
20         })
21     }
22
23     connect_to_db().then(() => {
24         console.log('=> get_all talks');
25         talk.find({_id: body.id}, { _id: 0, watch_next_ids:1})
26             .then(talks => {
27                     callback(null, {
28                         statusCode: 200,
29                         body: JSON.stringify(talks)
30                     })
31                 }
32             )
33             .catch(err =>
34                 callback(null, {
35                     statusCode: err.statusCode || 500,
36                     headers: { 'Content-Type': 'text/plain' },
37                     body: 'Could not fetch the talks.'
38                 })
39             );
40     });
41 };
42
~
~
<b/laboratorio-TCM/compito3/Get_Watch_Next_by_Idx.js" 42 lines --2%-- 1,1          All
```

Get_Watch_Next_by_Idx: risultati

GET fatta all'API gateway pubblico

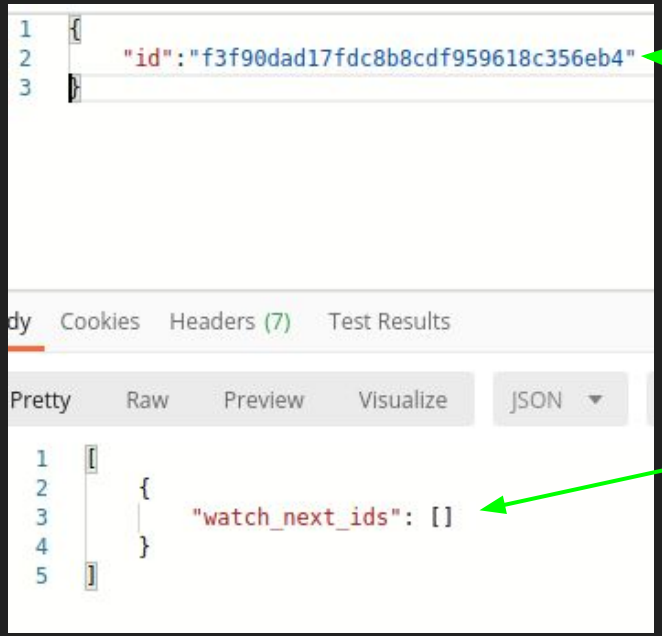
chiedo un id

ricevo un elenco di watch_next



caso particolare: watch_next vuoto

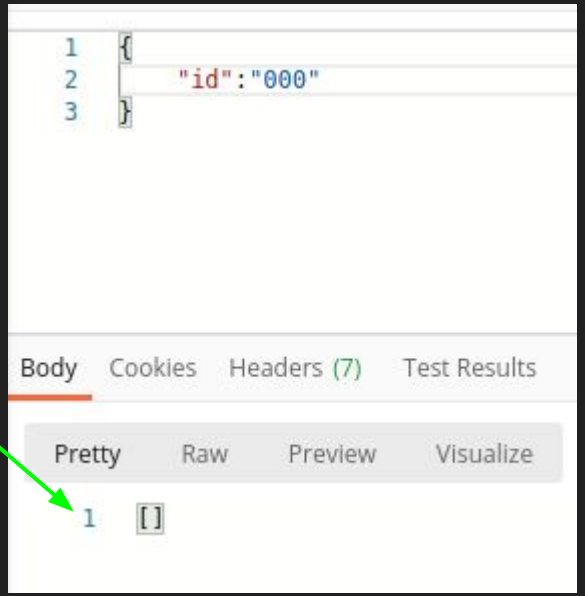
chiedo l'id di un talk che non ha watch_next



risposta corretta: campo watch_next_ids vuoto

caso particolare: id non valido

riesco a distinguere il caso con id non valido da quello con watch_next vuoto perché la risposta è diversa



esperienza utente/possibili evoluzioni

l’utente finale non utilizzerá direttamente questa funzione.

A seconda di come è pensata l’app finale si possono fare leggere modifiche per ottenere direttamente l’URL del video o il titolo o altre informazioni utili per consigliare l’utente nella visione di altri contenuti

criticità tecniche

- difficoltà nel fare debug
- nomi simili tra dati/variabili su servizi diversi