TalkWithTed

Mehmood Zakria - Tecnologie Cloud & Mobile 2024

Homework 3

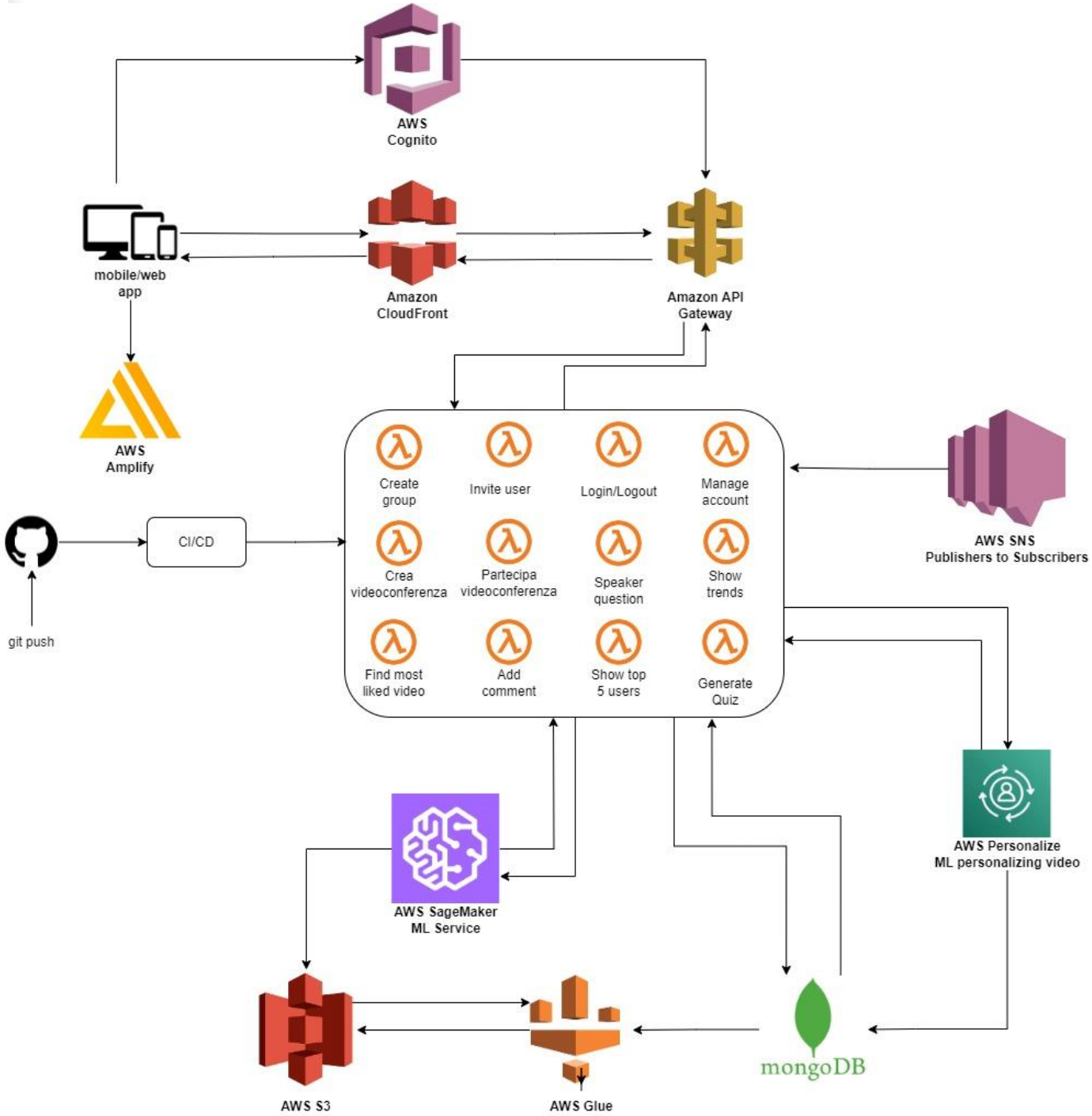
Architettura

Prima:

- ✓ Caricamento dati su S3 + mongoDB Atlas
- ✓ Implementazione Job PySpark mediante AWS Glue

Ora:

- ➢ Implementazione lambda function
- ➢ Esposizione API mediante Postman



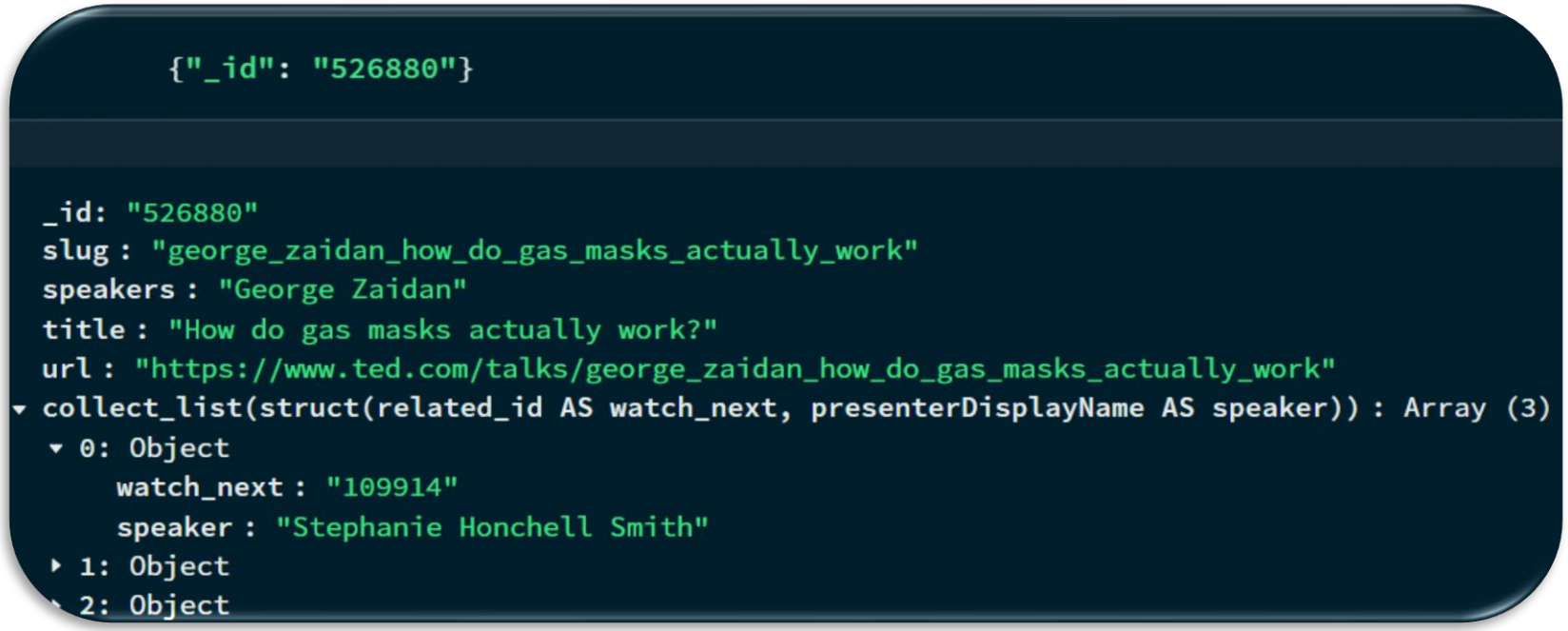
Handler.js WatchNext

Operazioni:

- Controllo se _id esiste
- Ricerca video relativi per id
- Formattazione Json con relativi campi per output

```
connect_to_db().then(() => {
    console.log('=> get_all talks');
    talk.find({_id: body.id})
        .skip((body.doc_per_page * body.page) - body.doc_per_page)
        .limit(body.doc_per_page)
        .then(talks => {
                talk.find({_id: talks[0].watch_next})
                .then(t => {
                    callback(null, {
                    statusCode: 200,
                    body: JSON.stringify({id: t[0]._id, title: t[0].title, details: t[0].details, url: t[0].url, watch_next: t[0].watch_next, speaker: t[0].speaker})
                })
            })
        })
        .catch(err =>
            callback(null, {
                statusCode: err.statusCode || 500,
                headers: { 'Content-Type': 'text/plain' },
                body: 'Could not fetch the talks.'
            })
        );
});
```

Esempio visualizzazione dati:



Output WatchNext

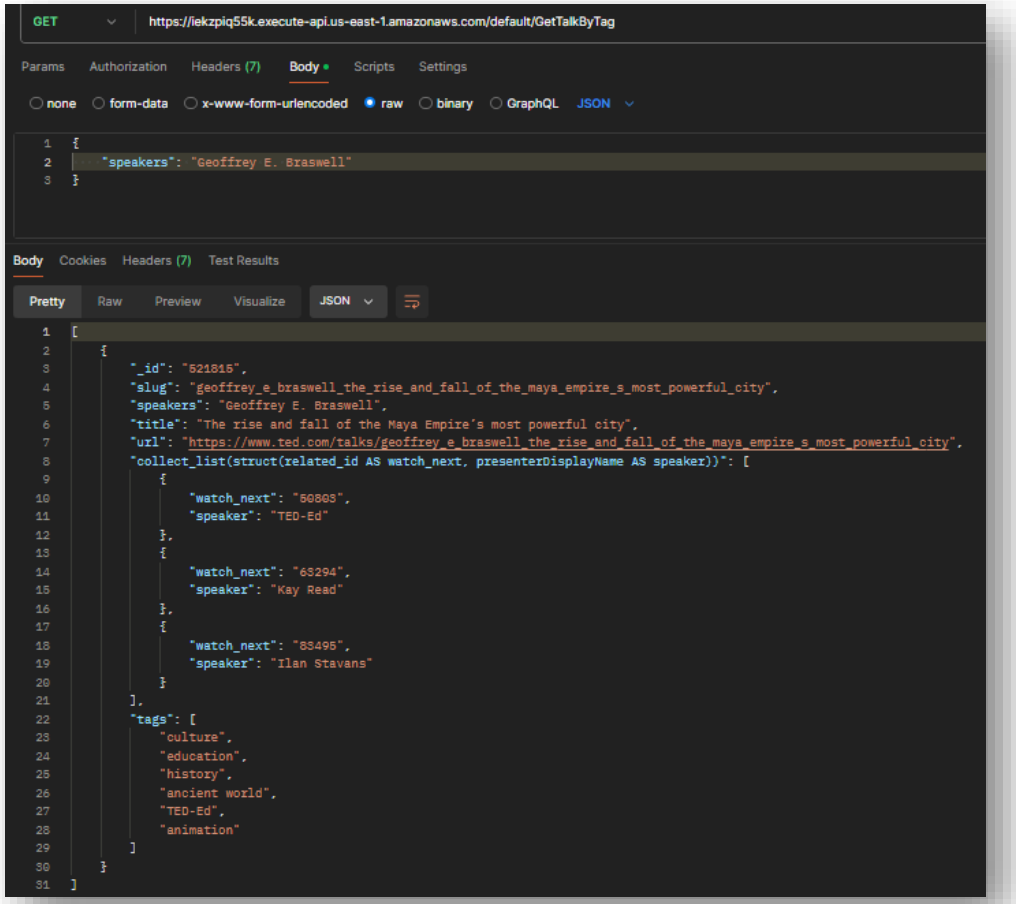
TedxSpeaker

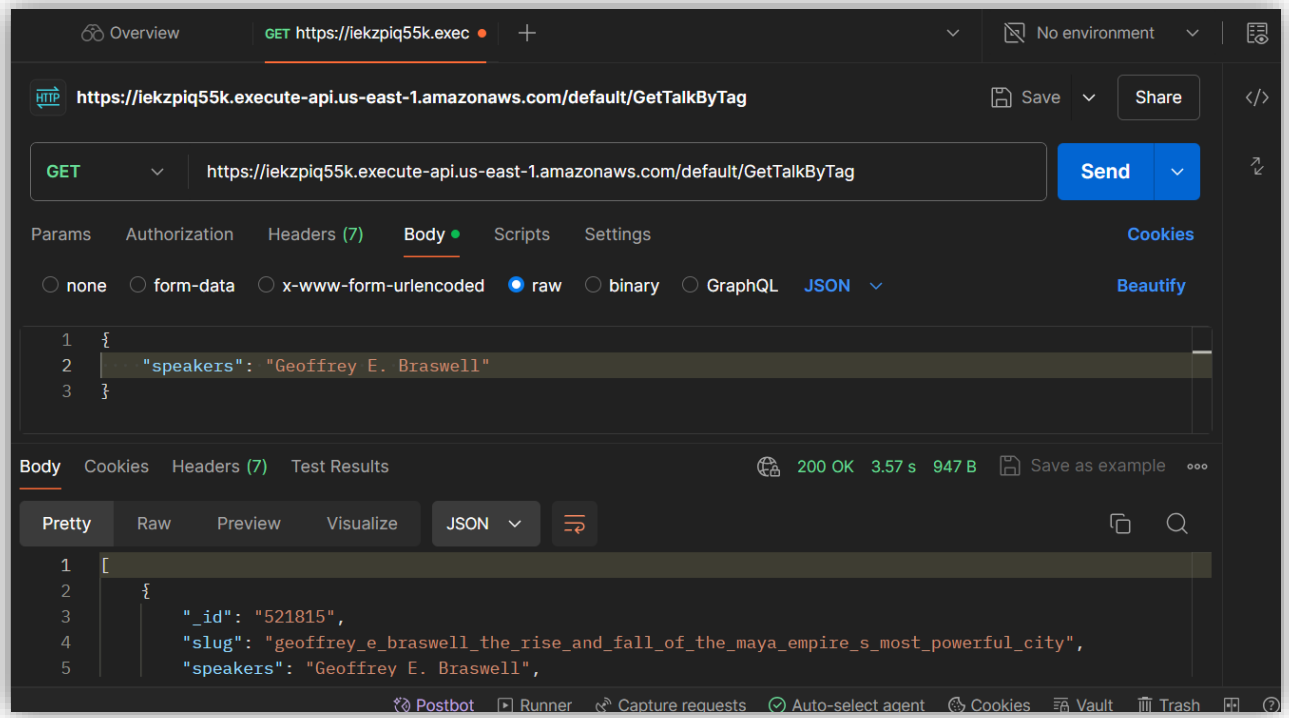
- Connessione a mongodbAtlas
- Controllo se speaker esiste
- Ricerca talk per speaker
- Formattazione output in JSON

```
const talk_schema = new mongoose.Schema({
    _id: String,
    title: String,
    url: String,
    description: String,
    speakers: String,
    watch_next: String
}, { collection: 'tedx_data' });
```

```
    connect_to_db().then(() => {
        console.log('=> get_all talks');
        talk.find({speakers: body.speakers})
            .skip((body.doc_per_page * body.page) - body.doc_per_page)
            .limit(body.doc_per_page)
            .then(talks => {
                    callback(null, {
                        statusCode: 200,
                        body: JSON.stringify(talks)
                    })
                }
            )
            .catch(err =>
                callback(null, {
                    statusCode: err.statusCode || 500,
                    headers: { 'Content-Type': 'text/plain' },
                    body: 'Could not fetch the talks.'
                })
            );
    });
};
```

Output Speaker



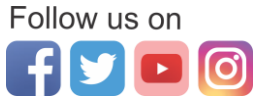


Criticità

- Debugging
- Poca esperienza con linguaggio node js
- Molti problemi di connessione a database, causati anche da piccoli errori di codice

Evolizioni

- Lambda function per l'ordinamento speaker per numero di talk



- Lambda function per la ricerca dei contatti per speaker

- Implementazione lambda function per generazione quiz

Quiz

Grazie

Mehmood Zakria

https://github.com/zakria-mehmood/unibg_cloud_e_mobile_2024

z.mehmood@unibg.it

https://trello.com/invite/b/PhlMCIv5/ATTI53506od83afb7092dd5d9ce5fa9e7331B166BE6F/talkwithted

Trello