



# TOP-TED

## COMPITO 3

BARCELLA GABRIELE (1058426)

BORDOGNA FILIPPO (1058427)

CASSINA ANDREA (1057831)

MORÉ GABRIELE (1058401)

# LAMBDA FUNCTION – WATCH NEXT BY ID

- Partendo dal codice visto durante la lezione abbiamo sostituito l'attributo tag con l'attributo id in modo da poter fare delle chiamate API puntando ad un preciso ID
- La risposta di questa chiamata è l'oggetto che contiene le informazioni di tutti i video correlati a quello dato come parametro

```
connect_to_db().then(() => {  
  console.log('=> get the id');  
  talk.find({_id: body.id})  
    .select("watch_next")  
    .then(talks => {  
      callback(null, {  
        statusCode: 200,  
        body: JSON.stringify(talks)  
      })  
    })  
})  
)
```

# LAMBDA FUNCTION – WATCH NEXT BY ID

- Potremmo utilizzare questa API per:
  - Ottenere i talk che trattano di un determinato argomento inerente al video appena visto
  - Ottenere una classifica dei video con più views e correlati al video appena visto
  - Mostrare altri talk dello stesso speaker
  - Mostrare le informazioni del relatore quando il video viene messo in pausa

```
[{
  "watch_next": [
    {
      "url": "https://www.ted.com/talks/michele_wucker_why_we_ignore_obvious_problems_and_how_to_act_on_them",
      "watch_next_idx": "c75f4b38653ffbd1823509ef2ed20928",
      "main_speaker": "Michele Wucker",
      "title": "Why we ignore obvious problems? and how to act on them",
      "details": "Why do we often neglect big problems, like the financial crisis and climate change, until it's too late? Policy strategist Michele Wucker urges us to replace the myth of the \"black swan\" -- that rare, unforeseeable, unavoidable catastrophe -- with the reality of the \"gray rhino,\" the preventable danger that we choose to ignore. She shows why predictable crises catch us by surprise -- and lays out some signs that there may be a charging rhino in your life right now.",
      "posted": "2019-05",
      "n_views": 68556651,
      "durate_sec": 5109,
      "tags": [
        "TED",
        "talks",
        ...
      ]
    },
    ...
  ]
},
...
]
```

# IMPLEMENTAZIONE RICHIESTA DEI TALK IN BASE ALLA DURATA

- La richiesta:
  - min\_durate: durata minima del talk (opz.)
  - max\_durate: durata massima del talk
- Funzionamento:
  - Restituisco i task con durata compresa tra «max\_durate» e «min\_durate»
  - Se non viene specificata «min\_durate» verrà impostata a zero

```
{  
  "min_durate": 1000,  
  "max_durate": 2000,  
  "doc_per_page": 10,  
  "page": 1  
}
```

```
connect_to_db().then(() => {  
  console.log('=> get_most_viewed talks');  
  talk.find({durate_sec : {$lt: body.max_durate, $gt: body.min_durate}},  
    {_id: 1, title: 1, main_speaker: 1, details: 1, posted: 1,  
     url: 1, n_views: 1, durate_sec: 1, tags: 1})  
    .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)  
      })  
    })  
  })  
})
```

# IMPLEMENTAZIONE DELLA RICERCA PER MAGGIOR NUMERO DI VIEWS PER MESE E ANNO

- Richiesta:
  - year: anno del talk (opz.)
  - month: mese del talk (opz.)
- Funzionamento:
  - Se nessun parametro viene specificato restituisce i video più visti di sempre
  - Se viene specificato solo «year» vengono restituiti i video più visti dell'anno
  - Se vengono specificati entrambi i parametri restituisce i video più visti nell'intervallo specificato

```
{
  "doc_per_page": 10,
  "page": 1,
  "year": 2020,
  "month": "02"
}
```

```
talk.find({posted: {$eq: body.year+"-"+body.month}}, {_id: 1, main_speaker: 1,
  title: 1, details: 1, posted: 1, url: 1,
  n_views: 1, durate_sec: 1, tags: 1})
.sort({n_views: -1})
.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)
  })
})
}
```



## CRITICITÀ E POSSIBILI EVOLUZIONI

- Non sono fornite API direttamente da TEDx quindi è necessario fare uno scraping
- Ricavare la trascrizione dei talk da API fornite da TEDx
- Tradurre i talk in più lingue di quelle fornite da TEDx



## LINK UTILI

- [Trello](#)
- [GitHub](#)