



Rex Software PTY. LTD
29b Helen Street
Newstead QLD 4006

rexsoftware.com



Backend Developer Test

The “Movie API” Test

Build an API which achieves the following:

1. Provides a user / password authentication mechanism which exchanges correct user / password for a secure token of some kind.
2. Is secured with a token for any request to an endpoint other than the authentication endpoint and root url (via middleware).
3. Provides a read / write API for the following domain:
 - MOVIE
 - Name
 - GENRE
 - Associated ACTORS (and the name of the character they play)
 - Rating
 - Description
 - Image
 - ACTOR
 - Name
 - Birth Date
 - Age
 - Bio
 - Image
 - GENRE
 - Name
 - MOVIES in this genre
 - ACTORS in this genre
4. Provides a method for uploading imagery of movies and actors.
5. Ships with migrations and seeders.
6. Provides documentation via API Blueprint or Swagger (you can either include this in your project, or publish it via a free account on <https://apiary.io/>). If you use swagger, please include instructions for rendering the doc.
7. Implement a testing mechanism.

Bonus points (required for senior applicants, optional for others):

- Implement a system that allows logged in users to "favourite" movies.
- Implement a rate limiting mechanism.
- Consider how thumbnails may operate for imagery - either implement a method for this or write a small doc on how you might achieve this.
- Implement a test suite to test the API implementation against the swagger or blueprint specification.
- Implement unit tests.
- Make it possible to launch the entire project, create and seed database etc. with no external dependencies other than docker / docker swarm by executing a single command line call (aka dockerise everything including the db, auto migrate, auto seed) and provide instructions to launch.

Items to pay attention to:

- HTTP Status Codes
- HTTP Verbs
- Response transformation (e.g. <http://fractal.thephpleague.com/>)
- Validation and errors (You probably shouldn't be able to save an author without a name)
- Code quality and documentation
- Response types should be in json
- Developer Experience (e.g. listing endpoints when you go to api root, providing errors if unacceptable content types are given etc)
- The experience for the person reviewing your code - include read me, commands to speed up activation etc.