**Introduction**

The application allows users to explore the characters of the Star Wars universe as well as the details related to each one of them such as movies where they appear, home world, race, etc. Users can search characters by name in the given search bar.

The backend act as middleware between the frontend and the data sources by consuming the SWAPI and local JSON file and providing its own API to the frontend.

For the search function, it will query the SWAPI data source first. If no result, for example “Grogu”, it will query the local JSON files. If it is still no result, the frontend will display “Result Not Found”. Otherwise, it will display the character details as a grid.

**Installation guide**

The application requires node.js 18.12.0, which can be installed by:

apt install nodejs   
apt install npm   
nvm install 18.12.0

To install dependency, in your terminal, please go to the “test” folder:

nvm use 18.12.0;   
npm install

cd client   
npm install

The application includes frontend (/test/client/) and server (/test/server/). To start the application, in your terminal, please go to the “test” folder, and run the following commands:

//start server   
cd server  
nvm use 18.12.0; npm start

//start client   
cd client   
nvm use 18.12.0; npm start

Open browser and go to the homepage by following the link <http://localhost:3000> :

A screenshot of a computer

Description automatically generated

If you are not using “localhost”, please update the “proxy” value in /test/client/package.json

**QA/Test plan**

To test the search by SWAPI in the server:

1. Get to the homepage <http://localhost:3000>
2. Input “r2”
3. Click “Search”. It should show the results as follows:

A screenshot of a computer

Description automatically generated

To test the search by SWAPI in the server (vehicles and starships case):

1. Get to the homepage <http://localhost:3000>
2. Input “Luke Skywalker”
3. Click “Search”. It should show the results as follows:

A screenshot of a computer

Description automatically generated

To test the search by local JSON file in the server:

1. Get to the homepage <http://localhost:3000>
2. Input “Grogu”
3. Click “Search”. It should show the results as follows:

A screenshot of a computer

Description automatically generated

To test result not found from the server:

1. Get to the homepage <http://localhost:3000>
2. Input “abc”
3. Click “Search”. It should show the results as follows:

A screenshot of a computer

Description automatically generated

**Rationale behind decisions made**

To increase response time, the application queries the data sources (API and local JSON files) asynchronously if possible.

Because of the layout, the application only shows the nest elements (e.g. vehicles, films) title or name even the full data is prepared at the server.

Because of the security, the react client is using proxy to call the server api. For more information, please visit:   
<https://blog.logrocket.com/why-you-should-use-proxy-server-create-react-app/>