My Movie List

Overview of Challenge App

As an avid movie watcher, I find that I would love to be able to keep a record of the movies I have watched or intend to watch, along with my own personal notes and ratings, so that I can refer back to my record when looking for movies to watch again or recommend to a friend.?For this challenge you should use the OMDB API (http://www.omdbapi.com) to supply your application with the data needed to allow a user to search for movies, and view details about the movies they find.?

Guidance?

The work we do as engineers often goes beyond just writing code, and you should use this challenge to demonstrate your knowledge and understanding of such aspects as planning, execution and testing, which are critical in real world software development. Here are some examples of questions that may come up in conversation with the interview panel:

- Have you broken down the work into small pieces? What is your process for that like? Could certain features be delegated?
- Time is limited, which features were prioritized in order to deliver an MVP in the allotted time?
- Is the code tested, can it be changed easily without introducing bugs?
- Can you explain your decision making, and defend you choices?

Feature Requirements?

Note to the candidate: We have intentionally included more features than we expect to be able to be developed in the 3-5 hour timeframe. Please prioritize the features as you see fit and implement what you can. We will discuss this prioritization during the interview. We understand that candidates are busy and their time is valuable.

- The user should see a basic UI.
- The user should be able to search for any movie they would like by title.?The OMDB API should be used as the data source.
- The user should be presented with a list of movies to select from based on their search results.?
- Selecting a movie should allow the user to see additional details about that specific movie.?
- The user should be able to save a movie and they should be able to mark it as "watched" or "unwatched".?
- The user should be able to add a rating and/or comments about the movie that they saved.
- If a user has not watched a movie, they should not be able to rate it.?
- A user should be able to see a list of saved movies along with the details, watched status, and comments/ratings for each movie.
- A user should be able to sort their list by title.
- A user should be able to sort their list by rating.
- When a user rates a movie, that action should be recorded in a static file log with a timestamp, and a basic description of the action. For example "2020/01/01 9:00pm User saved 'The Mask' to list"
- A user should be able to access their list of saved movies via a REST API which returns JSON as defined by the provided swagger json spec.?

Assumptions/Constraints

- Assume that this user is the only one using this app, so there will be no need for authentication.
- Assume that you will not have a database and all "saving" will be done in memory.?
- Save as little as possible locally, rely on the OMDB APIs where you can.

Expectations

- The Admin team develops in a combination of NodeJS + Express + React and Ruby On Rails. While we'd prefer that the challenge was developed with this in mind, please feel free to code this challenge in the language of your choice.
- Why reinvent the wheel? If you have a boiler plate or code generator you would like to use, we are into it, but please be ready to discuss what you chose and why.
- We have intentionally included more features than we expect to be able to be developed in the 3-5 hour timeframe. Please prioritize the features as you see fit and implement what you can. We will discuss this process during the interview.
- We are looking to understand how you think about code organization, component APIs, asynchronous data handling, etc. To best facilitate that, we recommend focusing on quality of implementation, with fairly clean code, rather than focusing on quantity of features for this challenge.
- While we aren?t hiring you to be a designer, we are hoping to see solid instincts around user interfaces. In short: do take the time to make it look good, but don?t worry about spending multiple hours making it look amazing.

Submitting Your Code

Once you have completed the challenge, please zip the project folder up, and upload the zipped files to google drive. Your Autodesk recruiter will provide you with a link. Feel free to include any files you used to plan the work or take notes. We welcome anything that helps us gain insight to your process.