Frontend Assignment

The product:

The product will be a Star Wars dashboard where users can view information about the different content types in the Star Wars universe, such as planets, starships, and characters. The dashboard will be divided into 6 categories:

- Films: This category will list all of the Star Wars films. The films will be displayed in a grid view by default, but users will be able to toggle to a list view. Each film will have a title, a poster image, and a release date.

- People: This category will list all of the important people in the Star Wars universe. The people will be displayed in a list view by default, but users will be able to toggle to a grid view. Each person will have a name, a birthdate, and a species.

- Planets: This category will list all of the planets in the Star Wars universe. The planets will be displayed in a list view by default, but users will be able to toggle to a grid view. Each planet will have a name, a climate, and a gravity.

- Species: This category will list all of the species in the Star Wars universe. The species will be displayed in a list view by default, but users will be able to toggle to a grid view. Each species will have a name, a homeworld, and a lifespan.

- Starships: This category will list all of the starships in the Star Wars universe. The starships will be displayed in a list view by default, but users will be able to toggle to a grid view. Each starship will have a name, a model, and a hyperdrive rating.

- Vehicles: This category will list all of the vehicles in the Star Wars universe. The vehicles will be displayed in a list view by default, but users will be able to toggle to a grid view. Each vehicle will have a name, a model, and a top speed.

Each category will open as a page, and each page will contain a list of individual resource elements with an image. There is a menu that opens a dropdown located on each resource element.

For example, the Films page will have a list of all the films in the Star Wars universe, each with a title and image.

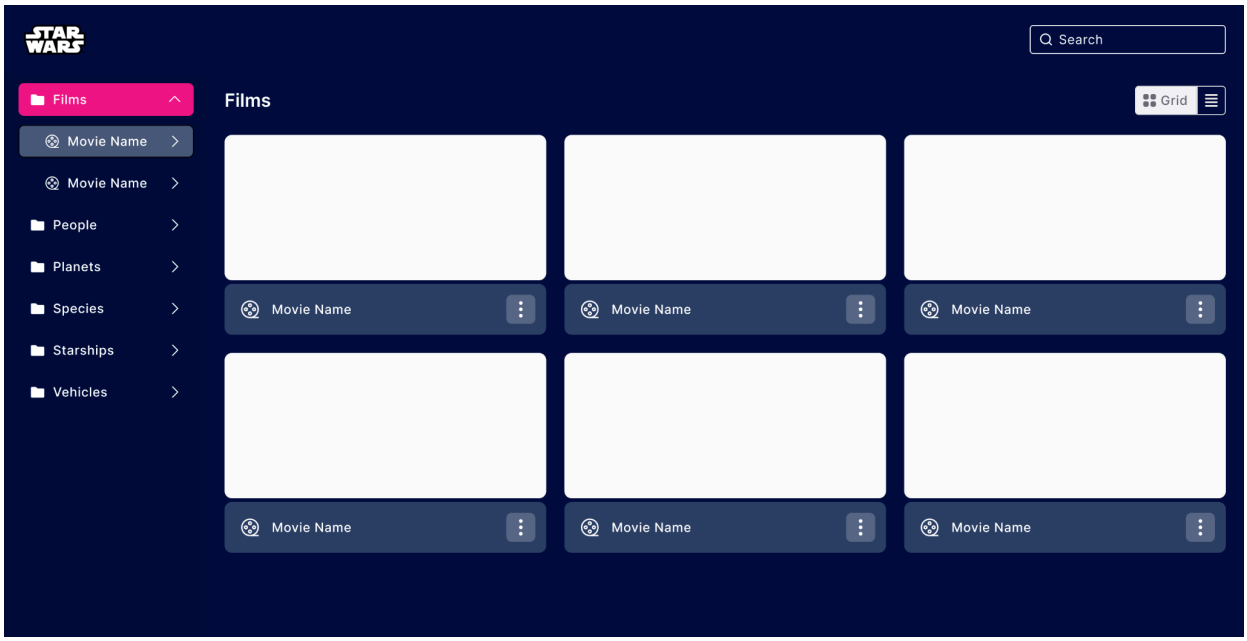
When a user clicks on a resource element, a sidebar will open to display all the details of the resource. For example, if a user clicks on a film, the sidebar will display the film's title, release

date, director, actors, and plot summary. Same will also apply to other resources as well. Here we will encourage you to get all details from the resource api here.

The design:

The design of the dashboard will be responsive, with all the elements scaling properly on different screen sizes. The design will also include micro-interactions and hovers, such as a film poster that animates when the user hovers over it. The images for the dashboard will be sourced from the website https://picsum.photos/, or users can add their own images. Refer to the Figma below for full design in 1366px

https://www.figma.com/file/of5rQlXrQ2nb1blflVNjzM/FRONTEND-ASSIGNMENT?type=design&node-id=14-1800&mode=dev



The assignment:

The assignment will be to create a React JS app that implements the design of the Star Wars dashboard. The app should use the SWAPI API to fetch the data for the different content types. The app should also be fully responsive and include micro-interactions and hovers.

API: https://swapi.dev/
Placeholder Images: https://picsum.photos/

The following are some specific requirements for the assignment:

- The dashboard must be fully responsive. (Mobile, Tablet, Web-1920)
- The design must be pixel perfect.

- The micro-interactions and hovers must be implemented correctly.
- The SWAPI API must be used correctly to populate the sidebar with data.
- The grid and list views must be implemented correctly.
- The source code must be well-organized and commented.

The deliverables:

The deliverables for the assignment will be the following:

- The source code for the dashboard. (GitHub link)
- A link to a live demo of the dashboard. (Optional)
- A brief write-up about the design decisions that were made.

The evaluation:

The assignment will be evaluated on the following criteria:

- The app's adherence to the design
- The app's pixel perfection to the design
- The app's use of the SWAPI API
- The app's responsiveness
- The app's micro-interactions and hovers