**Project 2 Documentation**

**Current Endpoints:**

/login

Supported Method: GET, POST

Query Params: username (account username), password (account password)

Description: This endpoint takes the user to a page that allows the user to enter their account information to log in

Supported Return Types: JSON, MongoDB data

/signup

Supported Method: POST

Query Params: username (account username), password (account password), pass2 (retype password)

Description: This endpoint allows the user to sign up and create an account

Supported Return Types: JSON, MongoDB data

/logout

Supported Method: GET

Query Params: N/A

Description: This endpoint allows the user to exit their account and go back to the login page

/maker

Supported Method: GET, POST

Query Params: title (move/show/video title), type (movie/title/other), link (optional URL)

Description: This endpoint lets users create and view their watch lists and details associated with the titles they create

Supported Return Types: JSON, MongoDB data

/getWatchlist

Supported Method: GET

QueryParams: N/A

Description: Shows all the titles created by the user and their details respectively

Supported Return Types: JSON, MongoDB data

**Purpose:**

The site’s purpose is to let users create lists of movies, tv shows, and other videos such as documentaries, clips, YouTube videos, etc. to act as a checklist or tracker for users to keep track of. For example, if the user wants to watch a movie and is afraid of forgetting to watch it, they could use this application to check which movies they still need to watch. Another example could be with YouTube videos and they can provide a link as well that they can click on to take them to the website and instantly watch the video.

**React:**

React is used in the login page where the user can instantly switch pages from login to signup and vice versa without reloading the page. This is also used in the maker page where the user can switch tabs to movies tab, shows tab, others tab, or all tab to take them to their respective pages without reloading, as well as when creating new titles, it will instantly add to the watch list.

**Mongo:**

Currently, the type of data being stored on Mongo is just a list of strings which are the title, type, and link of each entry the user makes. The title being what the entry is called, the type being whether it is a movie, show, or other, and the link being optional and to redirect the user to where the URL takes them.

**Handlebars:**

I am using two handlebars which are the login page and the app page. The login page is for letting the user login or signup and enter their credentials to take them to the app page. The app page is for letting the user create entries for their watch list as well as viewing their watch lists.

**Next Steps:**

The main features that I want to add/improve for the final versions are filtering the entries each user creates, for example, the movies tab only containing movies, and the shows tab only containing shows, etc. based on what ‘type’ the user enters, instead of all of them being mixed up in one page. Another feature I want to include is a delete and edit feature where the user can delete individual entries or edit the title/type/link. I will also improve the styling for the final version to make it look more professional and clean as the current prototype is not styled properly.