Michelle Yong Kai Wen: Project Portfolio for EzWatchList

1. Introduction

The purpose of this portfolio is to document my roles and contributions to the project, in terms of the code, the User Guide and the Developer Guide.

In a team of 5 software engineering students, we were tasked to enhance an existing desktop application, AddressBook, for our software engineering project. We chose to morph it into a watchlist for movie and television series called EzWatchlist.

1.1. About EzWatchlist

EzWatchList is a Command Line Interface desktop application, which means that it is designed for those who types fast and prefer to execute commands through typing. It helps users, especially those who are forgetful, to keep track of the shows they had watched or wish to watch, reducing the hassle of having to search for shows frequently.

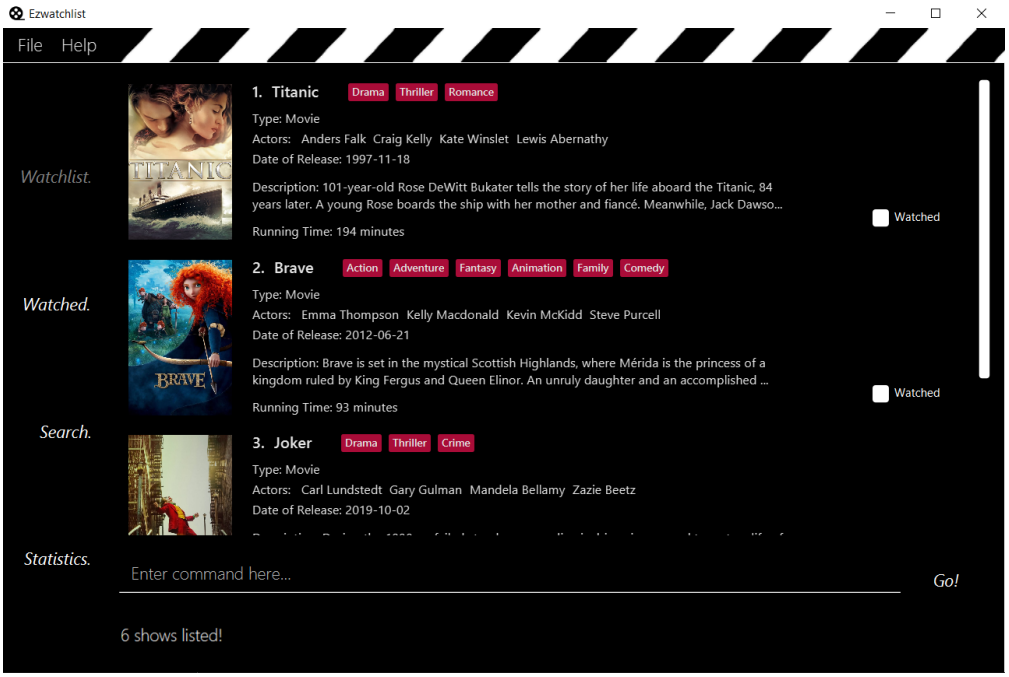

Figure 1. Graphical User Interface of EzWatchlist

1.2.1. Main Features

- Add, edit and delete shows
- Search online and offline
- Statistics

1.3. Interpreting Symbols

The table below shows a summary of the symbols used and their respective meanings

Symbol	Meaning
`Commands`	Commands input to the command line of EzWatchList
	Tips for the user
	Additional information
3	Keyboard input to EzWatchList

1.4. Role

My role was to implement the `search` feature and ensure proper documentation of the code. I had to implement the `search` feature such that users would be able to search when online and offline. The code had to be refactored and the `search` command had to be modified to allow users to search for different categories such as searching by show name, actor or genre.

2. Summary of Contributions

This section shows a summary of my coding, documentation, and other helpful contributions to the team project.

2.1. Major Contributions

Enhancement added: Implemented the `Search` command

- **What it does**: The `search` feature allows users to search for shows, from the online database, offline database, their watchlist or watched-list. Users can search for shows based on the show name, genres or actors and filter based on the type and where the show is found at.
- **Justification**:The `search` feature enables users to find new shows online easily or check if they had watched a certain show.
- **Highlights**: This enhancement requires a change on how `SearchCommand` process the search method and additional integration with the API.
- **Code contribution**: View my code contributions here.

2.2. Other contributions:

- **Project Management**: Assigned issues #39 #59 #171
- **Enhancements to existing features**: Implemented an internal storage #246
- **Documentation**: Updated the User Guide and Developer Guide #118
- **Community**: Reviewed team pull requests #30

3. Contributions to the User Guide

This section showcase my contributions to the EzWatchlist User Guide and my ability to write documentations to teach users on how use EzWatchlist.

Search for a show: `search`

Searches for shows whose names contain any of the given keywords from the online database, unless specified to be from the watchlist or watched list.

Format:

- by name: `search n/SHOW_NAME… [g/GENRE]… [a/ACTOR_NAME]… [o/FROM_ONLINE] [t/SHOW_TYPE] [w/HAS_WATCHED]`
- by genre: `search g/GENRE… [n/SHOW_NAME]… [a/ACTOR_NAME]… [o/FROM_ONLINE] [t/SHOW_TYPE] [w/HAS_WATCHED]`
- by actor (from watchlist only): `search a/ACTOR_NAME… [n/SHOW_NAME]… [g/GENRE]… [o/FROM_ONLINE] [t/SHOW_TYPE] [w/HAS_WATCHED]`
  - `SHOW_NAME`, `GENRE`, `ACTOR_NAME` can be any words, as long as it does not contain `/`.
  - `FROM_ONLINE` and `HAS_WATCHED` can only be `yes`, `true`, `no` or `false`.
  - `SHOW_TYPE` can only be `movie` or `tv`.

Special commands to take note of:
`o/no`: to search from watchlist or watched list
`o/no w/no`: to search from watchlist only
`o/no w/yes`: to search from watched list only

When searching based on genre online, only movies will be searched.

For `[o/FROM_ONLINE] [t/SHOW_TYPE] [w/HAS_WATCHED]`, if multiple entries are entered, only the last would be considered. e.g. "search n/Avengers t/movie t/tv" will be interpreted as "search n/Avengers t/tv".

Space between prefix and slash is not acceptable. e.g. "n /Avengers" will throw an error.
Space after prefix is acceptable. e.g. "n/ Avengers" will be interpreted as "n/Avengers".

The search is case insensitive. e.g "avengers" will match "Avengers".
The order of the keywords matter. e.g. "Chris Evans" will not match "Evans Chris".
Not only full words will be matched. e.g. "Joke" will also match with "Joker".

Example Usage:
You may want to search for movies named "Avengers" and also movies with an actor named "Tom". As shown below, assume that your watchlist only has a movie "Avengers: Endgame" that you have watched.

Figure 2. Current **Watched Page** *with a movie watched "Avengers: Endgame"*

1. Navigate to the **Search page** by one of the following ways:
    - clicking on the **search** tab
    - hitting the keyboard shortcut `3`
    - typing `3` in the command box and pressing `enter`
    - typing `search` in the command box and pressing `enter` as shown in the figure below

Figure 3. Entering of the `search` *command*

After you have entered the command, you will be led to the **Search page** as shown below.

Figure 4. Graphical Interface of the **Search Page**

You may skip Step 1 as you will be automatically be navigated to the **Search Page** when you key in any valid `search` command, such as the one in Step 2 below.

2. Enters `search n/Avengers a/Tom t/movie` into the command box in the **Search Page** as shown below.

Figure 5. Entering of the `search` command to search for shows

Press `enter` after entering the information shown above and wait for the information to load.

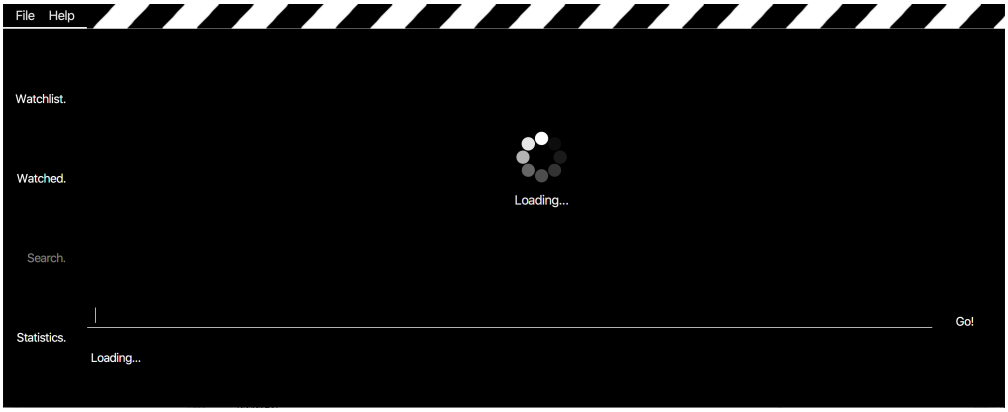

Figure 6. Loading Screen of the **Search page**

The loading page as shown above will appear while EzWatchlist searches for your shows.

You can go to the other pages in the mean time. Do allow some time for the search to load.

3. Search page shows the list of shows based on `search n/Avengers a/Tom t/movie`.

Scenario A: You are offline
Only movies from the watchlist or offline database with the movie name "Avengers" or actor name "Tom" will be shown, as seen in the figure below.

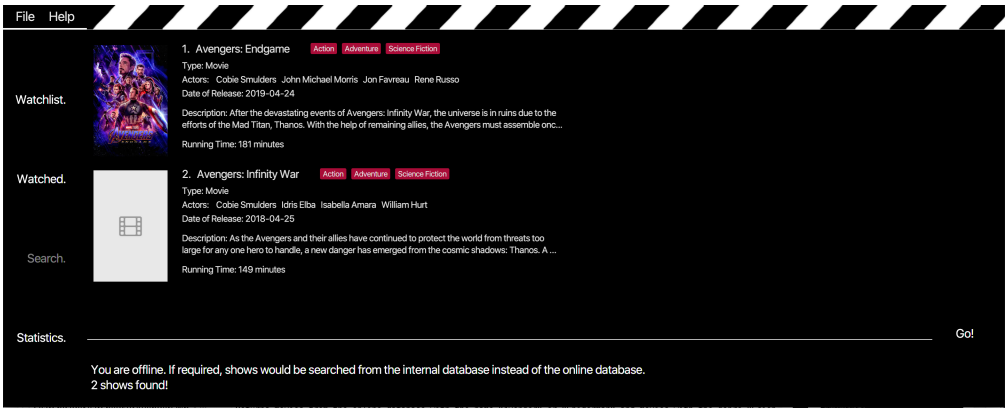

Figure 7. **Search Page** *showing the search results when offline*

Scenario B: You are online
Only movies from the online database with the movie name "Avengers" or actor name "Tom" will be shown, as seen in the figure below.

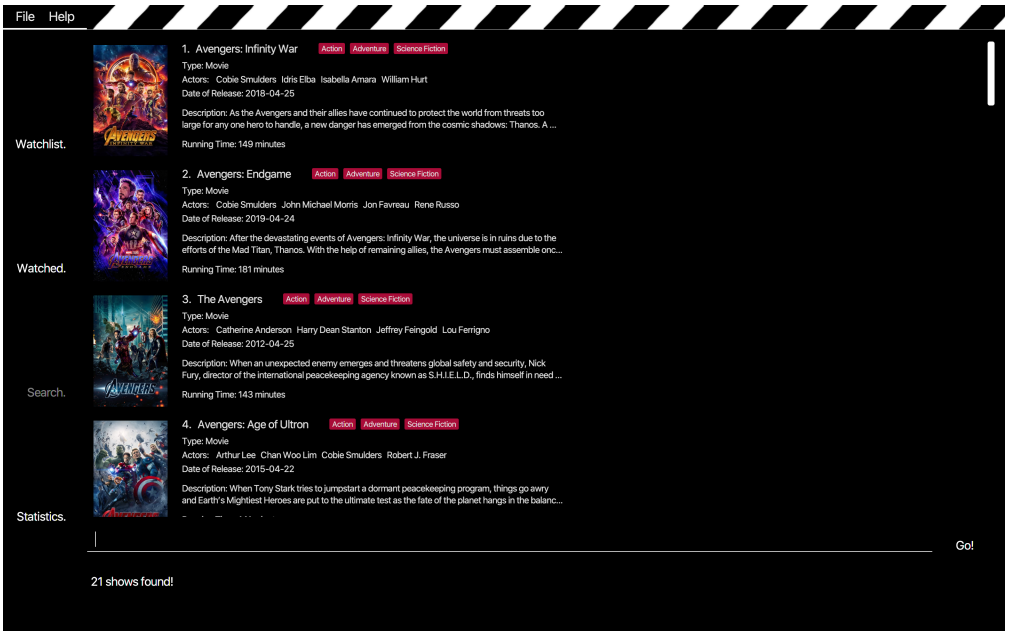

Figure 8. **Search Page** *showing the search results when online*

Examples:

- By name:
  - `search n/Joker o/no`
    Returns shows with the name "Joker" within the watchlist
  - `search n/Avengers g/Science Fiction t/movie n/Spiderman`
    (If online) Returns movies from the online database with the name "Avenger" or "Spiderman" and movies with the genre "Science Fiction"
    (If offline) Returns movies from the internal database, watchlist and watched-list with the name "Avenger" or "Spiderman" and movies with the genre "Science Fiction"
- By genre:
  - `search g/Action t/movie`
    (If online) Returns movies from the online database with the genre "Action"
    (If offline) Returns movies from the internal database, watchlist and watched-list with the genre "Action"
- By actor: (from watchlist only)
  - `search a/Tom o/no w/no`
    Returns shows within the watchlist with actor named "Tom"

4. Contributions to the Developer Guide

This section showcase my contributions to the EzWatchlist Developer Guide and my ability to use diagrams and technical terms to inform other developers on the features and implementation of EzWatchlist.

[Feature] Search Feature

The `Search` feature allows users to search for shows from the **online database**, the **internal database**, their **watchlist** or **watched-list**. It allows users to search for shows based on either "**name**", "**genre**" or "**actors**", or a combination of them.
Users can choose to search from the online database or their watchlist and watched-list, and also filter their search based on the show type.

The following *activity diagram* summarises the workflow of the `Search` feature:

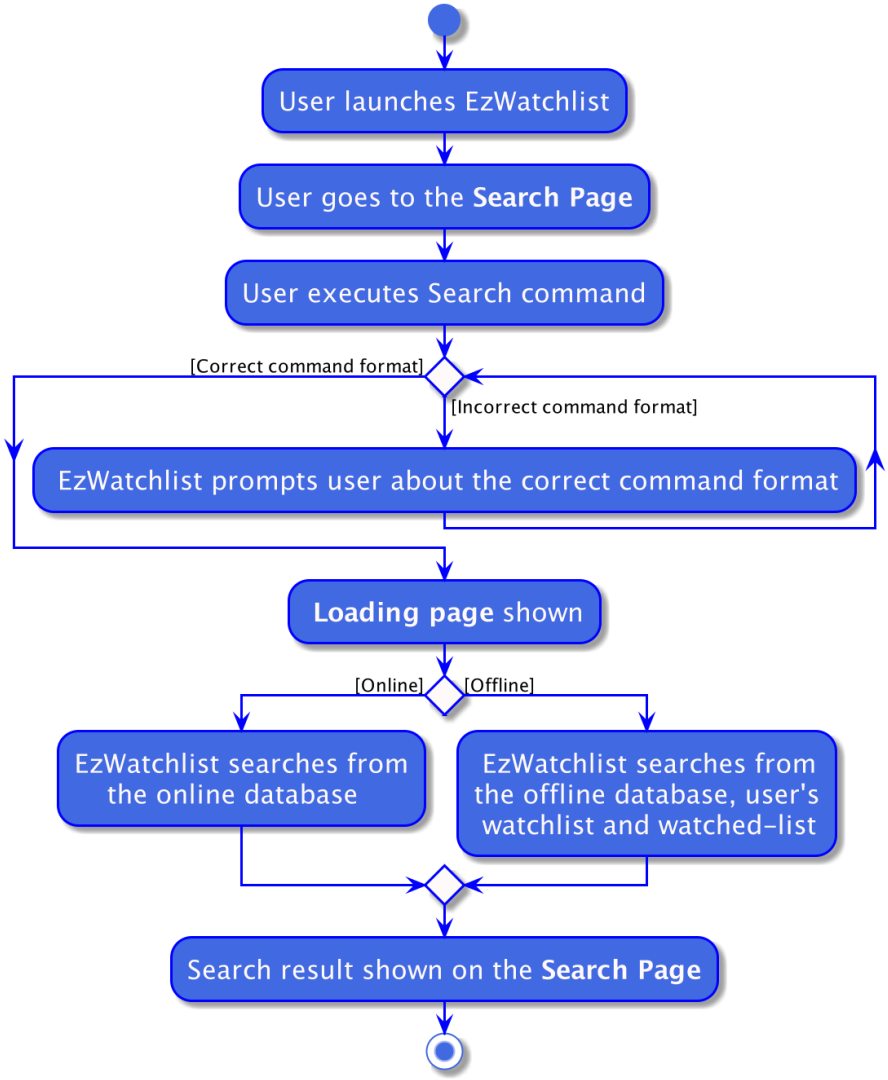

Figure 9. Activity Diagram for the `Search` Feature

Implementation

The search mechanism is facilitated by `SearchCommand` which can be found under the commands package. It extends `Command` and uses the `SearchCommandParser` to process the command entered by the user.

Given below is an example usage scenario and how the search mechanism behaves at each step.

Step 1. The user launches the application, goes to the **Search page** and executes `search n/Avengers o/no` command to search for shows named "Avengers" from the watchlist and watched-list.

Step 2. Entering the command calls `LogicManager#execute()`, which in turns calls the `WatchListParser#parseCommand()`.

Step 3. `WatchListParser#parseCommand()` returns a new `SearchCommandParser` and the `SearchCommandParser#parse()` command is called.

If the user enters a wrong command, such as incorrect prefixes or keywords, a new `ParseException` is thrown.

Step 4. A new `SearchCommand` is created, with the hash map containing the contents to be searched as a field of `SearchCommand`.

Step 5. The `SearchCommand#execute()` method is called, referencing the current model.

If the user is not connected online, a new `OnlineConnectionException` is caught and search would be done using the **internal database**, **watchlist** and **watched-list** instead of the **online database**.

Step 6. The `SearchCommand#searchByName()` method is called, referencing the current model.

Step 7. The `SearchCommand#addShowFromWatchListIfSameNameAs()` method is called, referencing the current model and name of the show to be searched from the list of shows.

Step 8. The `Model#getShowFromWatchlistIfHasName()` method is called, referencing the name of the show to be searched. A list of shows with the name "Avengers" is retrieved.

Step 9. The `SearchCommand#addShowToSearchResult()` method is called, referencing the list of the shows found in Step 7. Shows are filtered based on the possible filters and added the the `SearchCommand#searchResult`

Step 10. A new `CommandResult` is created, referencing the search message to be shown to user. This `CommandResult` is returned to the `LogicManager`.

The following *sequence diagram* summarizes how the search operation works based on the example above:

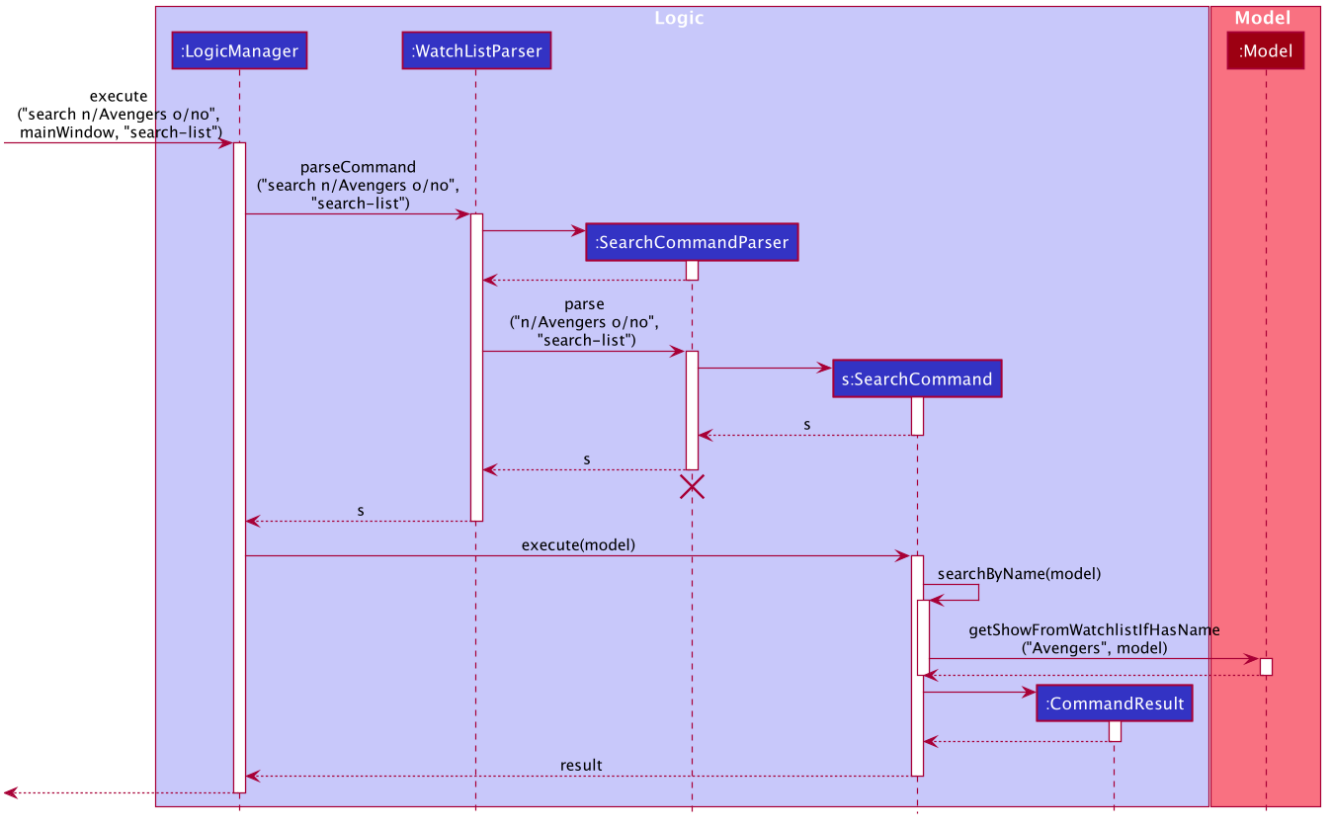

Figure 10. Sequence Diagram for the `Search` Operation

Design Considerations

Aspect: How `SearchCommand` reference the information to be searched for

- Alternative 1 (current choice): Takes reference to the hash map from `SearchCommandParser`, consisting of all the possible methods to search for shows
  - Pros:
    - Easy to implement
    - Can be easily made use of by retrieving the list of what the user want to search from the hash map
  - Cons:
    - Command may be longer and user would be required to be familiarize with the prefix, such as `n/` when searching by name
    - Certain list might be redundant as the user might not have requested to search by certain fields
    - Higher run time in checking through all the lists

- Alternative 2: Takes reference to a string from `SearchCommandParser` and by **name**, **genre**, and **actor** using that string
    - Pros:
        - Easy to parse the information from `SearchCommandParser` to `SearchCommand`
    - Cons:
        - Does not allow the user to have the freedom to choose what they would like to search by
        - May return irrelevant results back to the user as the user did not specify what they would like to search based on
        - Higher run time as there would be a need to search based on all 3 methods

Aspect: How `SearchCommand` is executed

- Current choice: Search by name, followed by genre then actor, when the user chooses to search with any combinations of the 3 methods
    - Pros:
        - Easy to implement and make use of.
        - Shows all the shows from the combination of the search results
    - Cons:
        - The user has to input either a name, genre or actor in order for the search to work
        - Logic is repetitive when searching based on name, actor and genre
        - Does not allow the user to search for shows that has a combination of the names, genres and actors
          e.g. `search n/Avengers g/Comedy` would show a search result with shows that either have the name "Avengers" or the genre "Comedy", instead of shows that have both the name "Avengers" and the genre "Comedy"