Modern Interface Design for Film Search Engine

Nadeem Nasimi, Elliot Radsliff, Anmol Singh, Devon Wasson

Department of Computer Science, Bucknell University, Lewisburg, Pa.

Abstract

The Film Search Engine website lets the user enter a search term or phrase and returns the screenshots of all scenes from the movies present in the database where the search phase was used in the dialogue. In doing so, it aids scholars in film and media studies by showing the relevant screenshots from movies, thereby helping them analyze and study films. The website's design makes it more efficient and manageable for the user to navigate through the search results.

Introduction

Our project aims to make a well designed web interface to aid in the research being conducted by our users. By employing good design practices, we are able to narrow down exactly what users need to be successful and increase user retention. We use React to build the website and follow Google's Material-UI style guide, both of which are decisions that will make it easier for people to continue development in the future. The result of our work is an intuitive application that grants users the ability to easily conduct their research and view their results in a well-formatted environment.

Methods

Our process was as follows:

- Brainstorm three good ideas
- Draw out wireframes on paper
- Conduct user interviews to find the best designs
- Refine design and create a medium-fidelity prototype
- Conduct user interviews with medium-fidelity prototype
- Refine designs in design document and begin coding

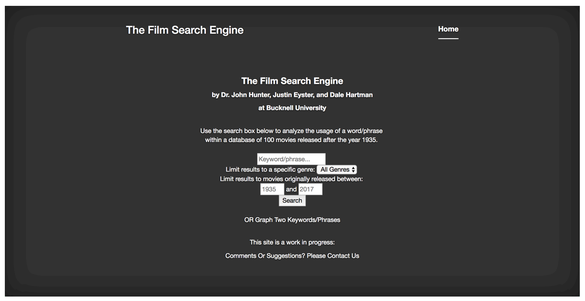
We use Facebook React as our primary JavaScript library for the interface. React is a high performance library for designing user interfaces. We use this library for several reasons:

- Readable and maintainable code
- Fast performance
- One-way data flow for efficient management of data

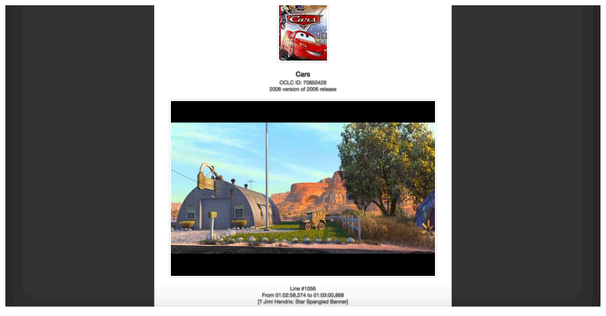
The following are the core features of our development environment:

- npm (Node Package Manager) to install dependencies with a single command
- Browserify and Babel for transpiling ES6 and React code
- Watchify for automatic bundling of our code to a single JavaScript file when any individual module is modified

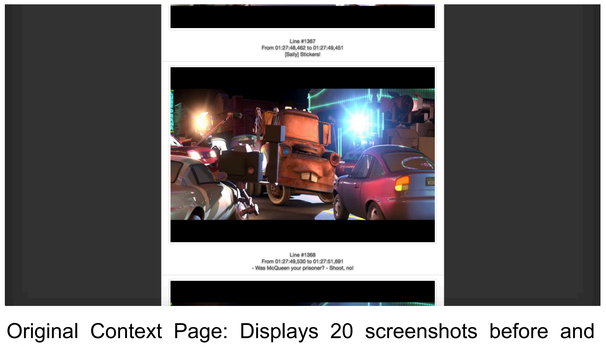
Results



Original Homepage: Perform text search from here

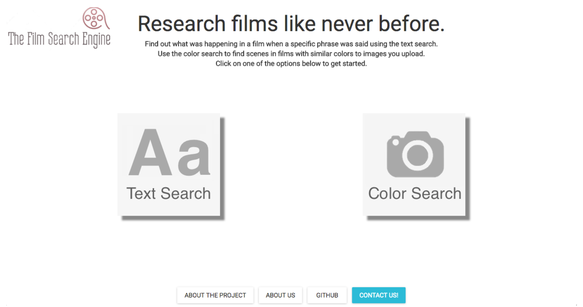


Original Search Results Page: Displays 1 screenshot/page

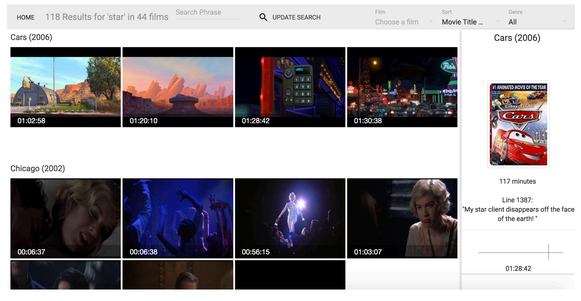


Original Context Page: Displays 20 screenshots before and after

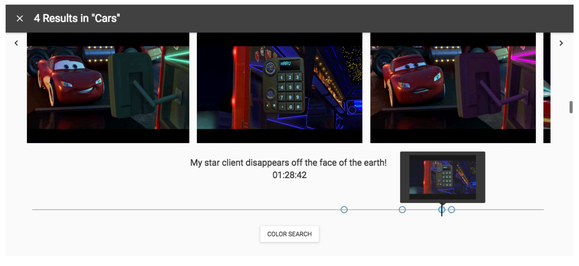
Results



Improved Homepage: Removes clutter and presents clear functionality of the web site



Improved Results Page: Adds a broad view, metadata drawer, and toolbar



Improved Context Page: Starts on the user's image of interest and adds a timeline

Discussion

The biggest aspects we would like to add to this project if we had time are as follows:

- Provide statistical details for users based on their search
- Minor usability improvements, such as floating film labels
- More filter and sort options
- Better integration with Color Search
- Keyboard shortcuts for navigation through the website

We were unable to implement all of the requested changes to the design as a result of receiving feedback from film and media studies scholars late in the process. Consequentially, one of the things we could have improved upon was getting feedback from our target users multiple times earlier in the process.

Conclusion

Our current user interface for the Film Search Engine website offers the users a clean, simplified, and user-friendly interface with additional features to ease and enhance the use of the website. With the use of best practices and extensive documentation on the new functionality, we hope to have a new team extend its functionality in the future.

References

1. http://www.material-ui.com/
2. https://facebook.github.io/react/
3. http://redux.js.org/
4. https://material.io/guidelines/
5. https://www.jetbrains.com/webstorm/
6. https://www.npmjs.com/
7. http://browserify.org/
8. https://github.com/substack/watchify

Acknowledgements

Professor John Hunter, Digital Humanities
Professor Brian King, Computer Science
Dale Hartman '18, Justin Eyster '17
College of Engineering
Department of Computer Science