

# Modern Interface Design for Film Search Engine

## **Design Document**

Team EndFrame

Nadeem Nasimi

Elliot Radsliff

Anmol Singh

Devon Wasson

1 May 2017

## **Problem Statement**

Research in cinema is difficult as it requires the researcher to watch and annotate entire movies to extract meaningful information. This can be both time-consuming and tedious. The goal of this project is to automate the process of “watching” movies and allow users to extract information from hundreds of movies in seconds. The ability to index words and phrases used in movies throughout history will allow researchers to easily identify trends in societal changes and identify how cinema adapted to handle the constantly changing landscape of humanity.

The goal of this project is to design and implement an intuitive user interface coupled with a scalable and extendable server architecture. With such a large and powerful dataset, there are many different types of search options and means of displaying the results in ways that would be valuable to scholars.

## **Background and Motivation**

Through this search engine, we plan to automate the process of “watching” movies and allow users to extract information from hundreds of movies in seconds. The ability to index words and phrases used in movies within the database will allow researchers to easily identify trends in societal changes and identify how cinema has adapted to handle the constantly changing landscape of humanity.

The system we built already had a well defined back end structure. There exists a database filled with metadata taken from 90 films. This database includes phrases of dialogue in films, stored with screenshots and timestamps. The goal of this project was to design and implement an intuitive user interface coupled with a scalable and extendable server architecture. With such a large and powerful dataset, there are many different types of search options and means of displaying the results in ways that would be valuable to scholars.

Our client left the design of the website very open ended for us. Over the next few weeks, we discussed among ourselves and with some CS faculty the potential of the website and some aspects we could improve upon. After discussing those ideas with our client, we came up with the following major requirements for our project:

- Streamline the output of relevant films
- Improve the output of individual films
- Improve navigation between searches
- Add more features

Looking at the original website design, we realized how cumbersome it was for the user to go through the screenshots that occupied an entire page and required scrolling down to get to the next screenshot. In addition, it was hard for the user to distinguish which screenshots belonged to what movie. This was another aspect we wanted to improve in the website.

Another aspect that we were able to check out early in the process that needed improvement was navigating between various search results. Improving navigation would prompt the user to use the website more often. Lastly, our client wanted us to add a few interacting features that would help the user in understanding and getting more information out of the screenshots than they usually did.

## **Design Process**

We began our design process by creating three separate designs for the user interface, drawing from the designs of popular websites like Netflix and Google. We then created simple paper prototypes of each of these designs and conducting user interviews, leading us to choose the design that tested the best amongst the most users. After receiving this feedback from users, we updated our design and created a more detailed digital wireframe using online tools, which can be found here [https://marvelapp.com/19bfcci/screen/16769862]. Using this digital wireframe, we conducted our second round of user interviews and used that feedback to further refine our design into our final design. This user interviewing process allowed us refine our design by revealing both the usefulness of certain features as well as overall flaws in the user experience. In order to implement our design in an aesthetically pleasing way, we followed Google's Material Design principles.

## **Results**

Below are some of the specific changes that we made to the website's user interface.

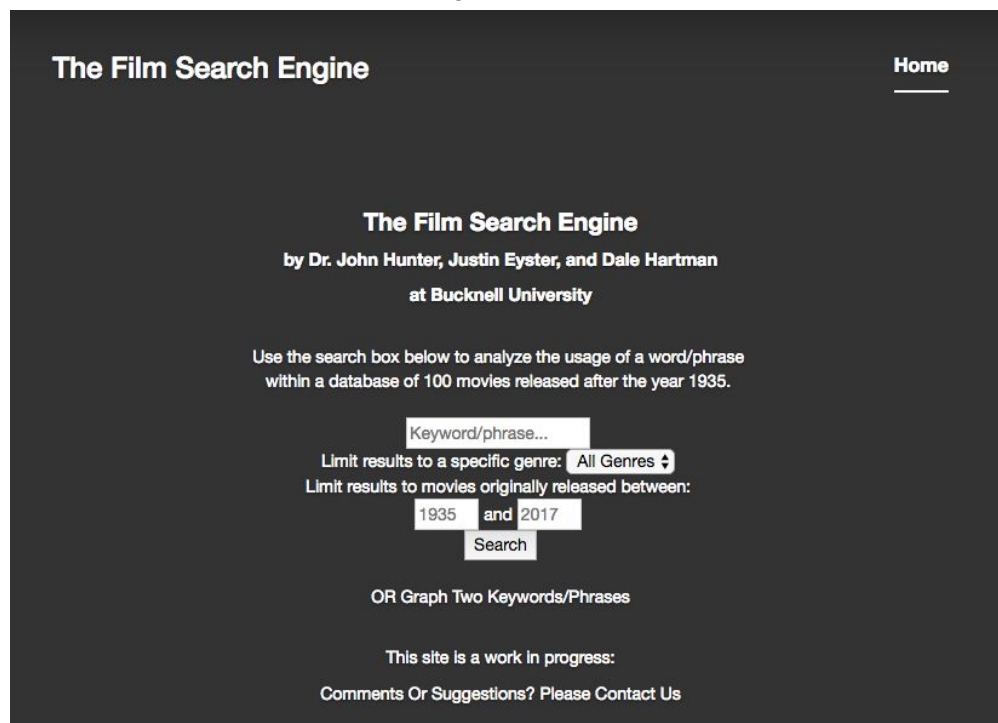
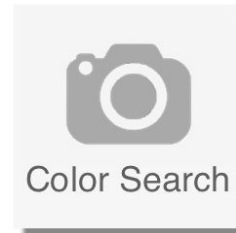
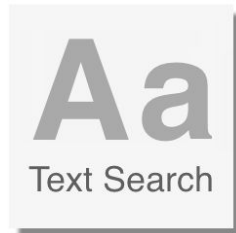


Figure 1: The original Homepage

## Research films like never before.

Find out what was happening in a film when a specific phrase was said using the text search.  
Use the color search to find scenes in films with similar colors to images you upload.  
Click on one of the options below to get started.



[ABOUT THE PROJECT](#)

[ABOUT US](#)

[GITHUB](#)

[CONTACT US!](#)

Figure 2: The new Homepage

As can be seen in the figure 1 and 2 above, we removed a lot of the clutter on the original homepage that came in the form of alternating text and text boxes and we made the main functionality of the page (to perform a text or color search) far more obvious.

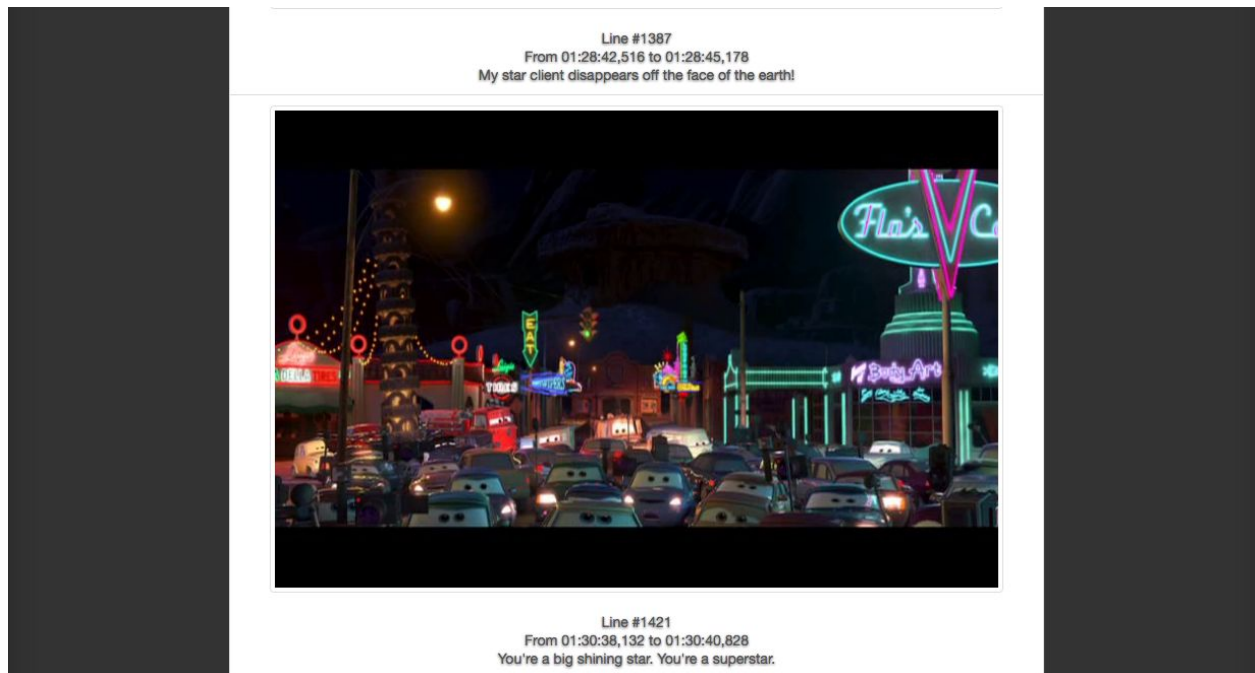


Figure 3: The original Results Page

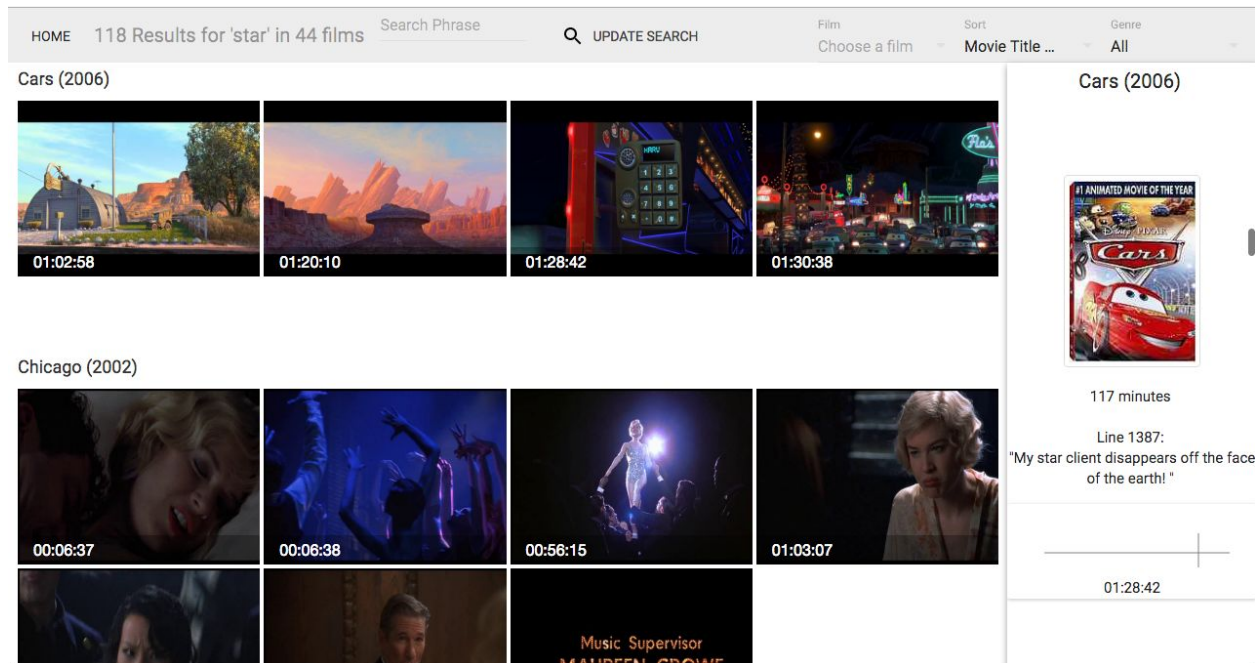


Figure 4: The new Results Page

One of the main changes we made to the results page (as can be seen in figures 3 and 4) are that we provided the user with a broader view of all the results in each film, meaning that they can now see more of the screenshots of interest in the film side by side. This allows the user to get a better feel visually of what is happening in the film when the search term is spoken. In addition, if the user is interested in learning more about a film, hovering over any screenshot causes the drawer on the right side of the screen to populate with metadata specific to that film and screenshot, such as: title of the film, movie cover image, film length, the spoken dialogue when that screenshot was taken, the line number of dialogue, and a timeline representing where in the film the screenshot occurs. We also added the toolbar above that allows users to quickly perform another text search without having to return to the homepage. The toolbar also allows the user to sort the results by genre, release year, relevance, or movie title (alphabetically).

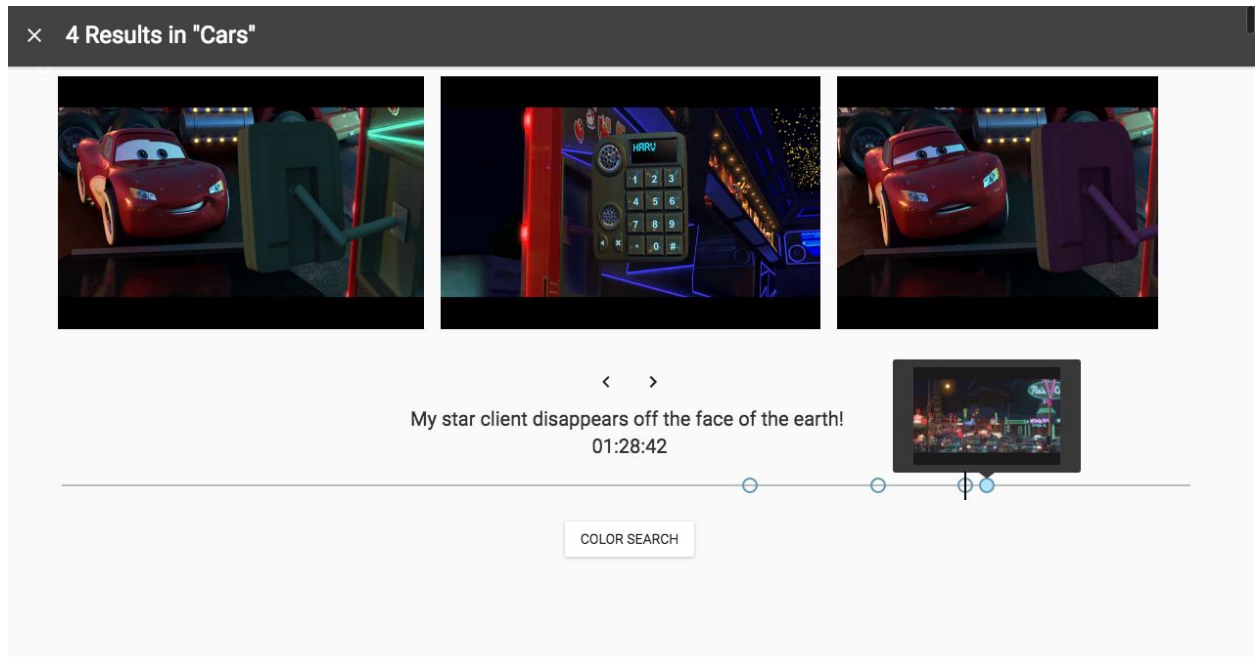


Figure 5: The new Context Page

Lastly as can be seen in figure 5, we updated the context page. Originally, the context page was just a page of the 10-20 screenshots that came before and after the screenshot of interest. Unfortunately, there was no good way to be able to look through the context page without losing sight of the other times the word of interest was mentioned in the film. We redesigned the context page to provide the screenshots before and after the screenshot of interest (all of the screenshots before and after are now available) and added a interactable timeline that displays the temporal location of all the times the word of interest is mentioned in the film and allows users to preview the screenshot at those times by hovering over one of the timeline's blue circles, and allows users to directly navigate to the screenshot of interest by clicking on one of those blue circles. Finally, we also added the capability to perform a color search on any of the screenshots within the film via the color search button at the bottom of the page.

## **Methods**

We started off our project with a good amount of planning. Our goal was to make a variety of low-fidelity prototypes, choose the best idea, and refine it from there into a medium-fidelity prototype. At that point, we conducted user interviews to refine the design even more until we knew exactly what we wanted to produce for the final project. Our process went as follows:

- Brainstorm three good ideas
- Draw out wireframes on paper
- Conduct user interviews to find the best designs from paper prototypes
- Refine design and create a medium-fidelity prototype on marvelapp.com
- Conduct user interviews with medium-fidelity prototype to find flaws in user experience
- Refine designs in design document and begin coding the actual website

This method allowed us to know what our plans were when we started coding. This saves a lot of time in the long run as we do not do any work with code that we might not use in the future. With this method, everything we coded ended up being a part of our final design in one way or another. If we did not do this much planning, we might have used too much of our time on one design, only to find out later that a different design would have been better.

When testing, we conducted interviews with a variety of users. We tried to interview people with and without technical backgrounds, with and without cinematography backgrounds. By doing so, we covered a sample of what our user base might be and were able to find out which aspects of our design were intuitive to some groups, but confusing to others. For the paper prototype, we had the users complete tasks of varying difficulty. Their success levels determined how well designed our prototype was. We were able to figure out which aspects of our design did not make sense to our users and refine our designs from there. Once we had a better version, we used a website to make a more interactive wireframe to allow users more control when interacting with the design. This round of interviews was more concerned with seeing how well the users could interact with the website given little information. Their adoption rate of different features told us where we needed to refine our design even further. Finally, we had enough information to go forward and make the high-fidelity prototype using the technologies we planned on using for our final product.

We decided to use Facebook React as our primary JavaScript library when developing our interface [1]. React is a high performance library for designing user interfaces. We chose this library for several reasons:

- The ability to write readable and maintainable code

- Fast performance through the use of the Virtual-DOM and only re-rendering components that have been modified
- One-way data flow allows for efficient management of data in the application

In order to help manage the client-side data we store in the webapp, we used a plugin called React-Redux [2]. This plugin connects the individual components we develop to a global store that keeps track of the state of the application. This store keeps track of the user's interactions with the interface and the data received from API calls.

Once we completed our medium fidelity prototype we began developing the interface by following Google's Material Design principles [3]. We did this by using the Material-UI ReactJS library [4], which contains components we could use to implement our prototype. These are the reasons we decided to use Material Design:

- Its slick, simple components encourage user engagement
- The layering aspect gives depth to the app, making for a more intuitive experience
- Combines classic principles of design with innovation and modern practices
- All aspects are well-documented, providing consistency

Our development workflow was made with productivity and long-term maintainability in mind. We use npm [5] to install JavaScript packages, along with a metadata file that contains a versioned list of the dependencies and a script to package them all together. As a result, a developer can easily get started by running "npm install" followed by "npm start" in their terminal window. The following are the other core aspects of our development environment:

- Webstorm as a JavaScript integrated development environment (IDE) [6]
- Browserify and Babel for transpiling ES6 (ECMAScript 6) and React code to JavaScript ES5, which is compatible with all modern browsers [7,8]
- Watchify for automatic bundling of our code to a single JavaScript file when any of the individual modules are updated [9]



## **Scrum**

Given below is the list of all user stories with their respective priorities that were accomplished we accomplished by our team:

<b>Priority</b>	<b>User story</b>
Business Critical	As a user, I should be able to easily see the films related to my search results.
Business Critical	As a user, I should be able to go back and forth between general and specific search results
Business Critical	As a user, I want to be able to search for a word or phrase and view the matched caption line and a screenshot of when the line was said, so that I can collect data about different films.
Business Critical	As a user, when I am viewing the results of the search for a specific movie, I want to be able to quickly navigate from one hit within the movie to the next hit within that movie
High Priority	As a user, I want to be able sort my search results based on time, genre, and hits.
High Priority	As a user, I want to be able to view a few screenshots before and after the relevant screenshot in order to be able to gain context about the relevant screenshot.
High Priority	As a user, I should be able to quickly see relevant information about the project on the main screen.
High Priority	As a user, I want to be able to hover over a film image and see meta-data from that film
High Priority	As a user, I should be able to save and share the current context of my search
Important	As a user, I should be able to give feedback on the website whenever I want.
Important	As a user, I should be able to specify if I want to search by color or by word
Low Priority	As a user, I should be able to see more detailed information about the project by clicking a link on the main page
Low Priority	As a user, I want to see a timeline of when a search term appears in a given film to see the distribution of search term hits throughout the film.

If we compare this to our major project goals, these user stories address those requirements fully.

Our burndown chart is given below.

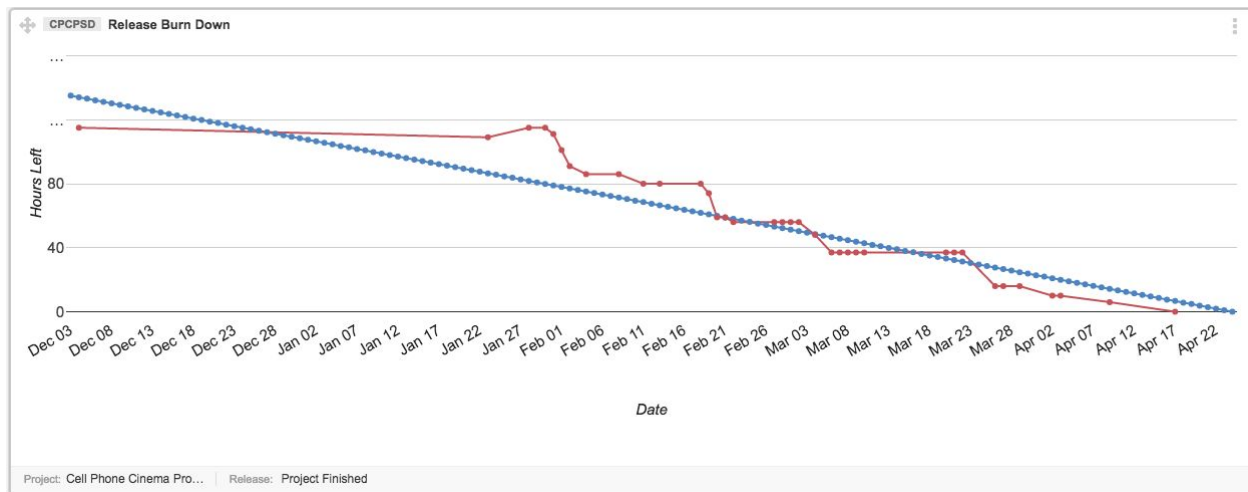


Figure 6 - Burndown chart for our project

As we can see, the biggest gap we had was from mid-December to mid-January where no work got done. This was due to winter break falling in that period. Coming into January, we were behind our expected work schedule as can be seen in the burndown chart, and we added in new user stories to meet changing requirements. The sudden drops at various points show the completion of the sprints. This burndown chart only updated when an entire user story was completed, and we completed most of our user stories at the end of each sprint. Going through nine major sprints and finishing up our work on time helped us finish our project a few days prior to the expected release date. The project velocity, calculated by dividing the total amount of hours by the days spent, was roughly 5 work-hours a day dedicated to coding. We had a conversion rate of about 1 hour on our scrum board being equal to 4 hours of work which accounts for the fact that our burn down chart only says that we worked for 120 hours the entire semester. Additionally, the burndown chart does not take into account all of the planning and meetings we had to attend, only coding and some prototyping hours.

## **Unimplemented Features**

There were many features we did not have time to complete or were beyond the scope of this project. These weren't part of our original user stories.

### **Search Page**

- Movie title bigger and darker color
- Have movie title float when scroll down
- Filter option of production studio
- Make genre filter scrollable
- Have more stats and graphical representations on the search results, and metadata, (X number of hits out of total possible hits in a movie)
- Selection of multiple genres
- Have arrows next to film poster in metadata drawer
- Graph to show # of hits per year
- Expand and collapse search results
- White space for single hit within a movie
- Have # of screenshots appear in the genres section drop down
- Make the update search part more intuitive
- Make timeline in metadata more intuitive

### **Context Page**

- Navigate through screenshots using arrows
- Have a way to magnify the timeline hits in areas that are cluttered

### **Home Page**

- Want to be able to search by phrase, not just multiple individual words

### **New Feature**

- Add other new and exciting data visualizations
- When music is played during a screenshot, be able to listen to that music

## **Works Cited**

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