

# WUD Film Mobile

## Iteration Two Progress Report

*WUD you like to see a film?*



**Arup Arcalgud, Erin Goldberg, Stephanie  
Goldberg, Charles Johnson, Davud Porter, Francis  
Thompson, Susan Yang**

Fall 2016  
CS 506 - Software Engineering

# Summary

## Difficulties

The most prominent source of difficulty we've faced so far was dealing with the limitations presented by our extremely constricted access to WUD Film's back end components. Although our client James LaPierre, the director of WUD Film, is excited about the app we are developing and was very cooperative with our efforts, he does not have the authority or permissions to give us access to the back end of WUD's website, servers, or resources. As a result, we had no other option than to work around these obstacles using multiple web scrapes across varying data stores. WUD Film did permit us access to their Google Spreadsheet where the initial web scrape began; utilizing the Google Sheets API we acquired the title, showtime, and runtime of each planned film from the given spreadsheet. Using the acquired titles, we then assembled unique URLs hosted by WUD film for a secondary web scrape triggered upon a film listing (each individual scrape is done only when requested to minimize overhead). To run this web scrape we turned to the JSoup library to obtain descriptions, posters, youtube trailers, etc. from each individually assembled URL. This workaround ultimately functions properly, though we quickly learned that access to a structured backend is the optimal path when designing an app based on a website. Having to work without direct access to the information our app requires was a major difficulty, but ultimately we were able to obtain the necessary data.

During the implementation of the web scrape, it was operational on only one machine; the Google API documentation did not provide sufficient documentation towards its use across multiple machines. Through some research, the solution was found by copying the debug.keystore file into each machine's .android folder. This limitation of access delayed development further than anticipated.

Similarly, we ran into an issue with storing the names and email addresses

that are manually typed into the email sign-up page. Storing locally will not be sufficient, so we need to rethink the API we are using and figure out a way to pass the data that we collect to WUD Film, either in the form of a spreadsheet or even a text file (since we do not have the ability to take the names and emails and subscribe them to WUD Film's emailing list ourselves).

Another issue we came across during development had to do with screen orientation. When switching between portrait and landscape orientations multiple times, the pages would display on top of each other simultaneously. This issue was ultimately fixed by screen locking our app.

## Feature Implementation Changes and Omissions

We did not implement the "Text Size" option on our Settings Page. Throughout working on this iteration, we came to the conclusion that this option would be redundant. It is unnecessary given what Android settings already can do, in that users already have the ability to set up preferences, such as re-adjusting text to a larger size.

Another feature that was removed in this iteration was the email submission pop up that would have signaled to the user that their name and address were submitted successfully. Rather than having a separate text box pop up in order to let the user know that their submission was successfully received, the submit button flashes and changes text from "Submit" to "Submitted", while the text boxes are cleared out, in order to more cleanly convey to the user that something happened.

A feature that has not come to fruition for iteration 2 is the pushing of email addresses to a shareable platform for our client to access. This has been pushed back for iteration 3 because of the difficulties we faced coordinating with an appropriate depository for the email addresses and names that minimizes maintenance requirements for the user (as well as most efficiently mitigates a lack of maintenance).

## Testing

The majority of tests we did for this iteration were manual, and predominantly focused on the assured functionality and accuracy of the web scrape. The nature of our app, lacking significant user interaction but focusing significantly on displaying information, as well as our app's dependency on a complicated webscrape procedure, does not lend itself to automated testing at this point in time. Exhaustive manual testing has functionally covered the web scrape (save for the possibility of the client changing their website, which we have no control over and can guarantee will break functionality). For iteration 3, we left the requirements sparse in order to devote most of our time and effort to bringing everything together and perfecting what we already have. Thus, the brunt of our testing will take place during iteration 3 when functionality is complete and tests can be more meaningful.

For this iteration we used functional testing to make sure that all of the features we desire for the app are working and stable. The webscrape has been completed and is able to pull the information we require from the Google Spreadsheet and WUD Film website. Even though the web scrape was happening correctly, there were cases where the spreadsheet had an incomplete movie title ("Willy Wonka" instead of "Willy Wonka and the Chocolate Factory") or the title didn't match exactly how WUD Film's webpage named the movie (how it appears in the url), then the movie's expanded tab showed up blank in our app because the second webscrape did not find anything to gather data from. There was one particular instance where the film "Blindness" had a misspelling in its URL on WUD's site and showed up as **Blindess** without the second "n", so in our app when that movie's tab was selected to expand it had no data to show despite our code functioning properly and working as intended. Regardless of these (admittedly very serious) unforeseen complications, we still managed to clean up one discrepancy that occurs when a title has an apostrophe, but mostly this problem is a result of nonuniformity between the WUD Film spreadsheet and however the WUD Film individual web pages are set up. This is not issue that we can entirely fix without restructuring our entire webscrape logic and rewriting

our webscrape.

Other functional tests cover all activity screens and the individual features that each activity has. Tests were not automated because of the compact nature of this iteration and the brevity of the features of our application. Features that we have not yet tested are features that require further implementation, such as the email address upload and color scheme settings.

We are looking ahead and working to take care of particularly bad corner cases for the foreseeable future. The date field for a movie is read in from the spreadsheet as a string (“Jan”, “Feb”, “Mar”, etc.) and needs to be compared against the current date given by the Calendar’s month field, which is an integer.

Much of our webscrape functionality depends on the uniformity of the data stores we are pulling information from (the spreadsheet and individual movie pages). Thus far we have had to manually alter several fields on the spreadsheet to conform to our generic code conversion while also jumping through several hoops to generalize and convert the given text to the web page. Moving forward we must ensure the editor of the spreadsheet as well as the creator of each individual film page inputs the data correctly across the data stores in order for our app to continue functioning correctly.

## Instructions

If you want to run the app on your own machine, you will need to switch out Android Studio’s debug.keystore file with the one that can be found in our team’s gmail inbox. On Windows, the keystore.debug file is located in C:/Users/USERNAME/.android. On Linux or Mac, it is located at ~/.android and you may need to show hidden files for it to be visible. If we are demoing our app again, then we already have this done and it won’t be an issue.

Gmail: [wudfilmsapp@gmail.com](mailto:wudfilmsapp@gmail.com)

Password: WUDFILMSAPP1

Code: <https://github.com/spyang4/wudfilm>