Flick Blender

Team 3

# Increment 1 Report

Rakan Alanzi

Bill Capps

Doug Hoskisson

# Introduction

## Project Goal and Objectives

Successful television shows often spawn additional shows and movies. While fans are delighted to see more from their favorite characters, it is easy for a person to become confused about the order each episode should be seen in. Many fans have created websites to help instruct other fans how to watch their show, but these sites often only cover a single franchise. For an individual attempting to create a list himself or herself, it can be a very tedious and time-consuming task. Our project solves this problem by using information from the internet to automatically create a list of episodes in the correct order for viewing.

As mentioned above, there are some websites that solve this problem for an extremely limited scope. Other websites, such as www.IMDB.com [1], provide information about shows without letting combine the information from other shows that are related. There are also several application programming interfaces (APIs) that give access to the same information as the websites. Our project would be unique by letting users create a viewing checklist using any combination of shows and movies that they want.

The objective is to allow people to spend their time enjoying their favorite television programs instead of frustratingly attempting to determine what episode to watch. The application should also include additional information about the shows and individual episodes to prevent the user from having to go to additional locations to find facts such as a cast list or episode synopsis. Finally, the application should be interactive, allowing the user to find information and make changes to the episode list.

The primary feature of the application is to merge television shows and movies together using the date they were released for viewing. Other features will allow the user to modify this sort order using the order the episodes appear on DVD or manually alteration. Additional information is available through APIs. This information could include cast, synopsis, reviews, and what streaming services offer the episode. The list will also let the user mark which episodes have been seen so that the user always knows what to watch next. It may also notify the user of new episodes that will be aired within the next few days.

We are considering also adding some social features to the application. One feature would be allowing the user to see what his or her friends have watched. Another option would be to automatically (at the user's discretion) post viewed episode on Facebook or Twitter. We may also include the ability for the user to provide his or her own ratings or reviews of watched videos.

This application would fulfill a need and provide useful features to the user. Like most of applications of this type, it would be supported by advertisement revenue or donations if put into production.

# Project Plan

## Schedule

### *Increment1*

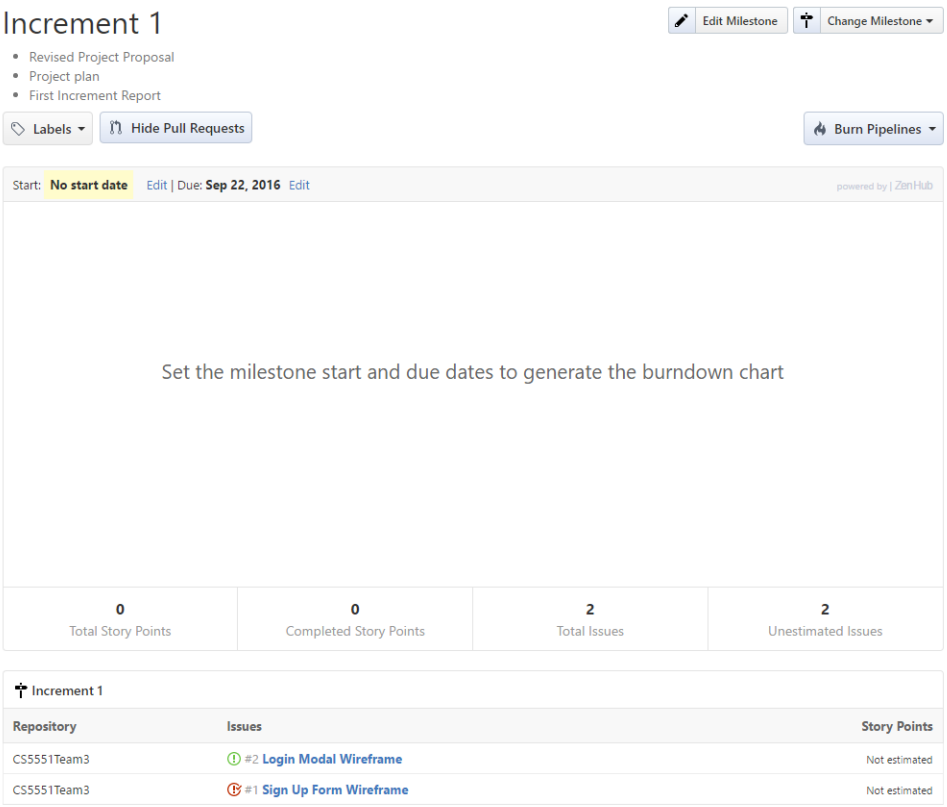
- Create skeleton of main page
- Create login screen
- Get access to APIs
    - Facebook oauth
    - Google oauth
    - TheMovieDB

## Timelines and Responsibilities

- Rakan
    - Create skeleton of main page
- Bill
    - Get access to APIs
- Doug
    - Create login screen

## Burndown Chart

During Increment 1 we have been learning how to use it as a team.  As a result, the burndown chart does not show correctly yet, but it will for Increment 2.



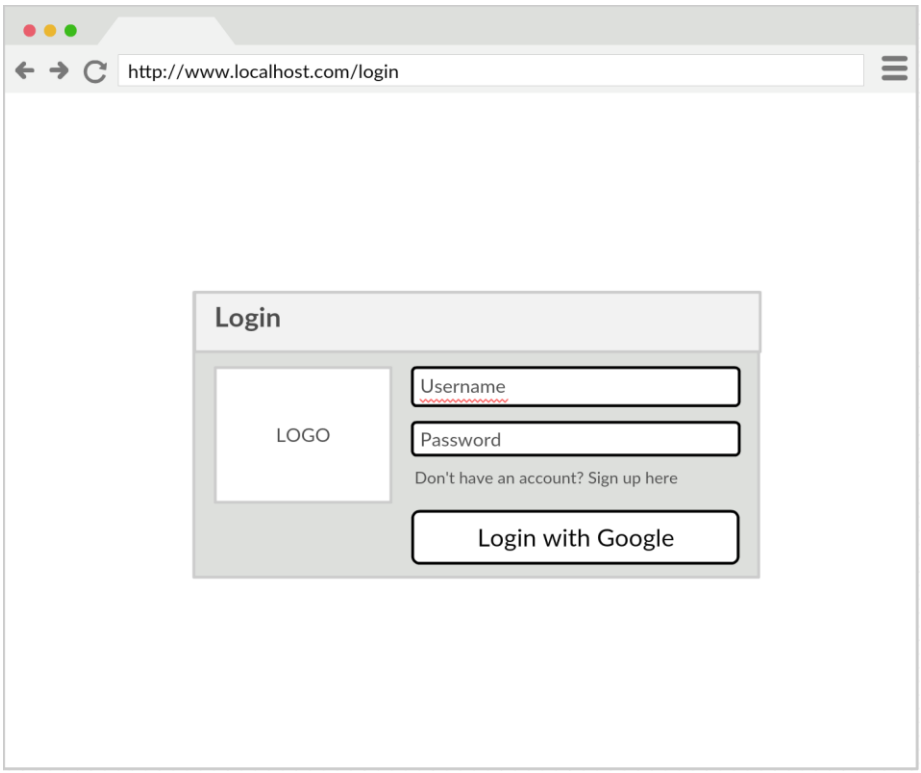
# First Increment Report

## Existing Services

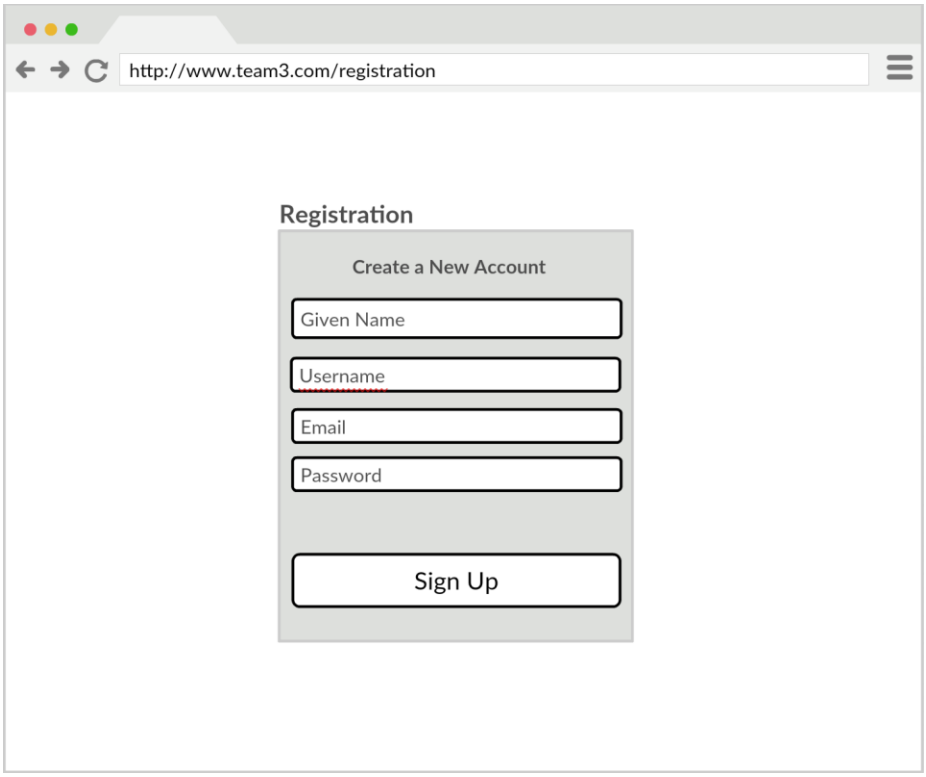
- Google oauth
    - Used to authenticate the user into the system
- TheMovieDB.org API
    - Used to access information about TV shows and movies

## Detail Design of Features

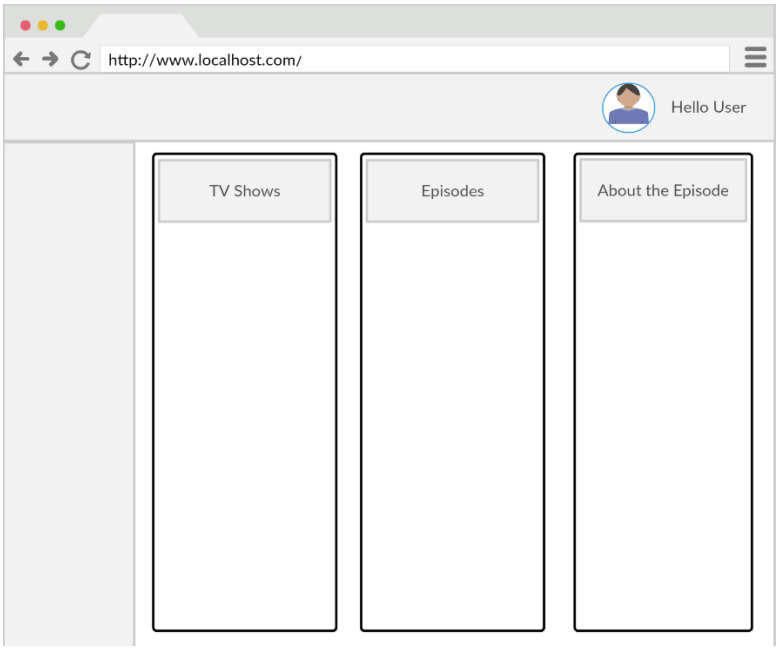
Login page



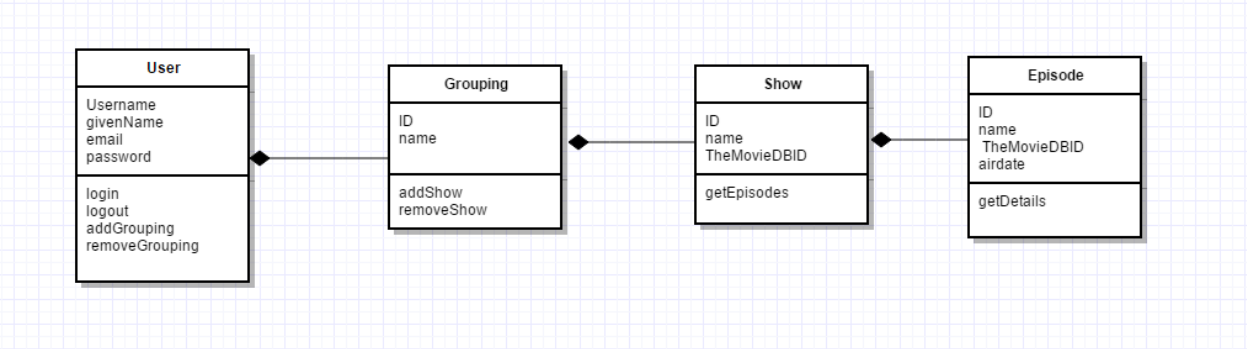
Registration Page




Home Page



Class Diagram




## Testing

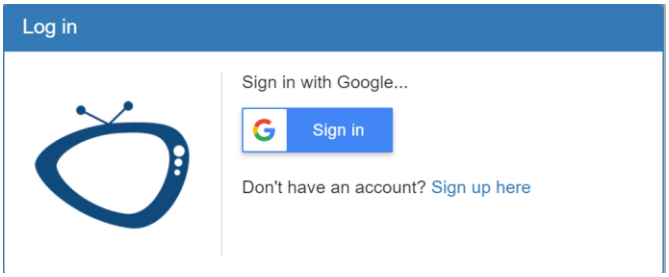
| # | Title | Description | Expected Outcome | Result |
|---|---|---|---|---|
| 1 | User with Google account | A user authenticates using Google credentials | The user should be taken to the home page | Pass |
| 2 | User without Google account | A user attempts authentication but does not have Google credentials | The user should be denied access to the system | Pass |

## Implementation/Deployment

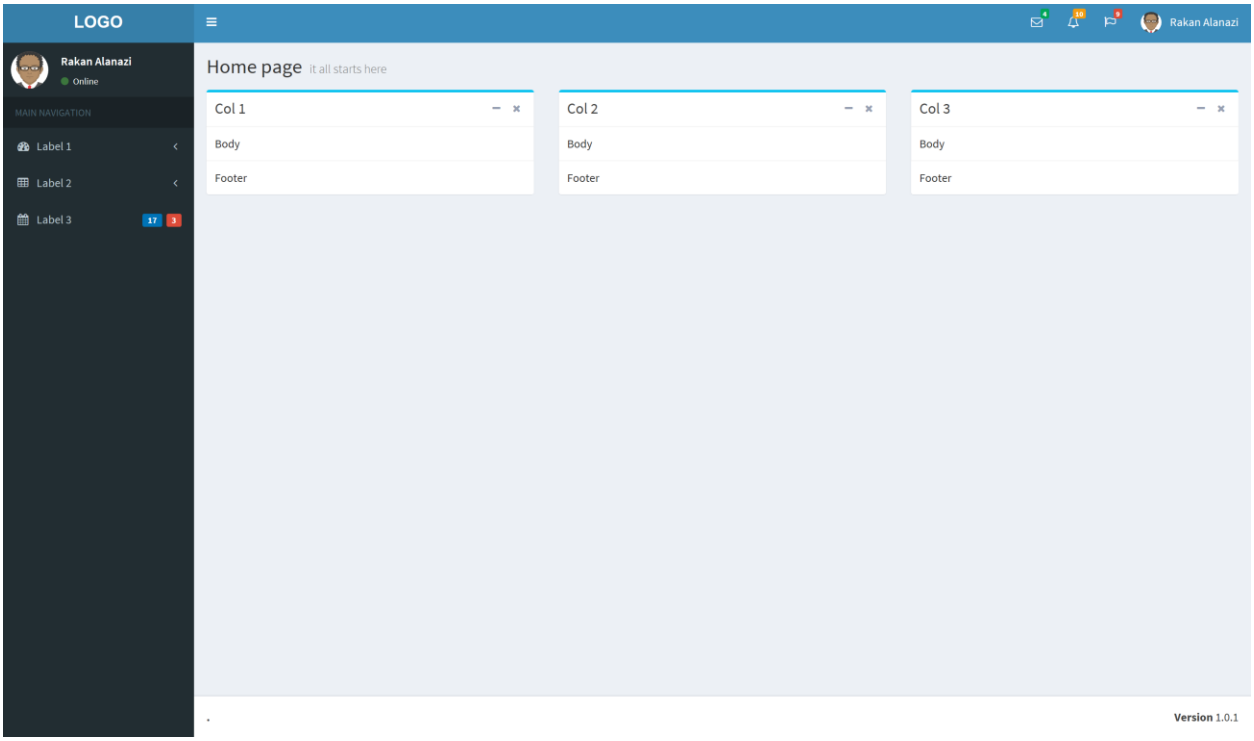
Technologies used:

- HTML
- REST
- Bootstrap
- CSS
- JavaScript

Login Page



Home Page


LOGO
Rakan Alanazi
Online
MAIN NAVIGATION
Label 1
Label 2
Label 3
17
3
Rakan Alanazi
Home page
it all starts here
Col 1
Body
Footer
Col 2
Body
Footer
Col 3
Body
Footer
Version 1.0.1

## Project Management

- Rakan
    - Create and document skeleton of main page
    - 33%
- Bill
    - Get access to APIs
    - Create report
    - 33%
- Doug
    - Create and document login screen
        - Using Google account only
    - 33%

## Bibliography


1. IMDB - http://www.imdb.com/

2. Episode Guides Android Application - https://play.google.com/store/apps/details?id=com.lordmarty.epguides

3. TVmaze.com - http://www.tvmaze.com/

4. Yahoo! Answers - https://answers.yahoo.com/question/index?qid=20090713153353AAw4mTd

5. GateWorld - http://www.gateworld.net/news/2009/05/stargate-recommended-viewing-order/

6. Arvin Johansson Arbab - http://arvtard.com/stargate-sg-1-and-atlantis-viewing-order-python-code

7. StackExchange - http://scifi.stackexchange.com/questions/4877/what-order-should-i-watch-the-various-stargate-television-series-in

8. Tumblr - http://nowwearealltom.tumblr.com/whovieworder

9. Rothwell - http://rothwell.im/etc/doctor-who/

10. Blogspot - http://wibblywobbly-timeywimeystuff.blogspot.com/2012/12/doctor-who-torchwood-viewing-order.html

11. Daniel Goleman - http://www.danielgoleman.info/