Project Exploration and Requirements

David Swift & Andrew Hyland

# Architecture Overview

For this assignment a MVVM / MVC / 3-Tier approach has been taken as the foundation of the web application. MVVM is mention due to angular using this pattern as its core design, and we say MVC simply because angular is designed in a way to split each page into its own view with their own controllers (and more) along side data models if that particular page requires it. Lastly it happens to be 3-tiered because the flow of control is divided, mongoDB controls the data storage in the form of a JSON (specifically a variation called BSON) database, nodeJS being the server of choice that handles the requests and routing with the help of expressJS, while angular is the front-end of the application that handles the views.

Various different technologies come together to make this assignment possible and as noted above the core is made up of **M**ongoDB, **E**xpressJS, **A**ngularJS, **N**odeJS or simply put MEAN, 3 types of mean project structures we’re tested, the first being a do it yourself approach which was a lot more work than initially expected, so mean.io and meanjs offered a lot of potential and after some research into experiences by other developers meanjs was chosen. It turned out to be a chaotic project structure that did not match the intended direction for development. Angular-Fullstack was then discovered by accident that happen to line up closely to the do it yourself approach with much less bloatware as well.

It should of been made clear by now MEAN is a framework, since the the combination of the above four are so common many have implemented their own full stack framework solutions.

# User Stories

## Search

From any page a search should be available.

From the home page the search will be conducted across all data.

From a platform page the search will be narrowed by default to the platform specific data.

Results are narrow with each character typed

e.g. search: call of duty

call of duty

call of duty: modern warfare

call of duty: black ops

call of duty: ghosts

call of duty: advanced warfare

-> continue typing into search

search: call of duty: a

call of duty: advanced warfare

### Success:

Item(s) populate the page with the results.

### Failure:

page appears unpopulated.

## Create Review Article

From navigation (maybe) select ‘create review’.

Enter review details.

Submit/add review.

### Success:

Review successfully posted to database.

### Failure

Empty fields.

Already exists

## Create Comment / Review

From review article page select comment/review to article/comment.

Enter comment/review details.

Submit comment/review.

### Success

Comment/review successfully posted to database.

### Failure

Top level comment already exists (only one top level comment per user).

Empty comment field.

No rating on top level comment.

# Release Plan

The intended approach of releasing is to first establish a functional routing protocol for the web app navigation. That is assuming the index container page has been created first.

The next step is to create a basic structure of the page views.

Followed by creating templates for the articles.

Pull data to and from database.

Display data form database.

Adding create review article functionality.

Adding comment/review functionality to an article.

Built the search functionality.

Finalise the stylings.

Add any additional features if time and need permit.