CO4015 MComp Computer Science Project

Interim Report

Gaming Centre

Simon Fisoye-Kings November 2020

School of Informatics, University of Leicester

Contents

[Aim and Objectives 3](#_Toc57062269)

[Aim 3](#_Toc57062270)

[Objectives 3](#_Toc57062271)

[Survey of Literature/Information Sources 4](#_Toc57062272)

[Similar Applications 4](#_Toc57062273)

[Recommendation Algorithms 4](#_Toc57062274)

[Requirements 5](#_Toc57062275)

[Mandatory 5](#_Toc57062276)

[Recommended 5](#_Toc57062277)

[Optional 5](#_Toc57062278)

[Outline of Specification and Design 5](#_Toc57062279)

[React 5](#_Toc57062280)

[Redux 7](#_Toc57062281)

[Bootstrap 8](#_Toc57062282)

[MySQL 8](#_Toc57062283)

[Auth0 8](#_Toc57062284)

[Planning and Timescales 8](#_Toc57062285)

[Bibliography 10](#_Toc57062286)

# Aim and Objectives

## Aim

The main of this project is to create an application to allow gamers to have a centralised hub where they can get the relevant information for their gaming interests. The application should retrieve the latest information about video games that the users are interested in. Users will be able to post their own content for either their friend or the world to see. A review section will allow users to get a range of user reviews in addition to reviews from critics. A recommendation algorithm will be used to recommend games based on their previous application usage, which shows the games that they are interested in. This application will give gamers a single platform to meet all their gaming needs.

## Objectives

1. Prepare the technologies and platforms for where the application will be developed.
   1. Set up the web application framework
   2. Connect the framework to a database where data can be stored securely
2. Set up the authentication section of the application
   1. Create a feature that will allow users to log into the application
   2. Create a feature that will allow users to register for access to the application.
3. Set up the content feed section of the application and view game details
   1. Create a feature to allow users to upload content
   2. Develop a feature where users can like and comment on content
   3. Develop a feature where users can search for a video game
   4. Develop a feature to allow users to view details of a video game
4. Set up a review section of the application
   1. Allow users to view both user and critics reviews
   2. Allow users to submit reviews on video games
5. Set a recommendation feature
   1. Allow users to specify their games of interest
   2. Allow users to enter their details that may affect their interest in video games
   3. Show users the games that are recommended to them.

# Survey of Literature/Information Sources

## Similar Applications

The official IGN website is a good source of information for this project [1]. The website has features that would be useful for the application that is being developed in this project. These features include being able to search for a video game, viewing details on a video game and their review system. However, the review section only consists of a single review from IGN themselves, which I believe this application can improve on. This is an example of a source allowing me to view possible missing features that could be incorporated in this project’s application.

Another website that was used as a good source was IMDB [2]. This website was used in a similar way to IGN as it included several features that could be useful in this project’s application. This was mainly with its review system that had both reviews from critics and users. This website does also allow you to view details about video games that you search for. However, the range of features for video games is limited as the website focuses more on films. Missed features such as recommendations for users is something this project’s application will be able to do.

## Recommendation Algorithms

An engaging resource is a paper that looks in detail about the recommendation system that is used by the popular video streaming platform, Netflix [3]. The report not only looks into Netflix’s method of recommendation but also the most popular recommendations that are known. This consists of Content-based Filtering, Popularity, Collaborative Filtering and Hybrid Approaches. The paper provides a detailed view of each algorithm, conducts data analysis and discusses the implementation of Netflix’s recommendation. This paper is an impactful source that will allow me to get more information on the different choices of recommendation algorithms and the benefit & drawbacks that comes with them. It also allows me to investigate Netflix’s own algorithm to understand how they have been so successful.

In a similar case, another paper looks in detail at how the popular video game digital distribution service, Steam, recommends the next video game for its users to purchase in its recommendation algorithm [4]. This platform is an improvement on the previous paper as this one is in the same market in terms of video games. The report completes a comprehensive test of three different recommender models. These include DeepFM, Factorization Machines and deep neural networks. The paper also makes a significant and exciting review of the effect that reviews on games have on recommending video games to users.

Another interesting source is a paper that looks at one of the most popular applications in YouTube. The paper investigates the Deep Neural Networks for YouTube Recommendations [5]. The paper looks in detail on one of the most sophisticated recommendation systems for a leading video sharing platform. YouTubes recommendation algorithm uses deep neural networks. Neural networks are a series of algorithms that actions to know the underlying relations in a set of data through a process that copycats the way the human brain operates. This source provides an in-depth investigation on arguably the best recommendation algorithm out there. Users discover 70% of videos watched through this recommendation system.

To get a better understanding of the different options that are available when choosing a recommendation algorithm, a paper on an improved version of Collaborative Filtering Recommendation Algorithm and Recommendation Strategy [6] was reviewed. The paper looks at an enhanced collaborative filtering procedure, a study into community detection process and dual overlapping community detection systems. “To solve the problems existing in the traditional and recommended technologies, this paper proposes a collaborative recommendation method based on community detection based on community discovery technology and collaborative recommendation technology in social networks” [6].

# Requirements

## Mandatory

* A user should be able to login using their email and password.
* A user should be able to register themselves.
* A user should be able to view the details of a video game.
* A user should be able to view their recommended games.
* A user should be able to edit their personal information.

## Recommended

* A user should be able to post their game-related content.
* A user should be able to search for a video game.
* A user should be able to leave a review on a game.
* A user should be able to view reviews on a game.

## Optional

* A user should be able to view a critics review on a game.
* A user should be able to view a recommendation percentage on a game.
* A user should be able to like & comment on content that is posted.

# Outline of Specification and Design

## React

This application will be developed using React. “React is a JavaScript library that is used for building user interfaces” [7]. React is declarative, which makes your code predictable and more comfortable to debug. When data changes, React efficiently updates the views and renders the right components. Being component-based allows React components to take care of their state and then conduct them to create high-level user interfaces. Component logic is written in JavaScript, pass data through the application and save the desired state of the DOM. React also makes it very easy to reuse code, making the general program code reusable.

One of the main features of React is the composition of its components. It allows components written by different developers to work together. It also enables the local state to be modified into a component without the need to change any other components using it. Components essentially state any composable behaviour, which includes rendering, lifecycle and state.

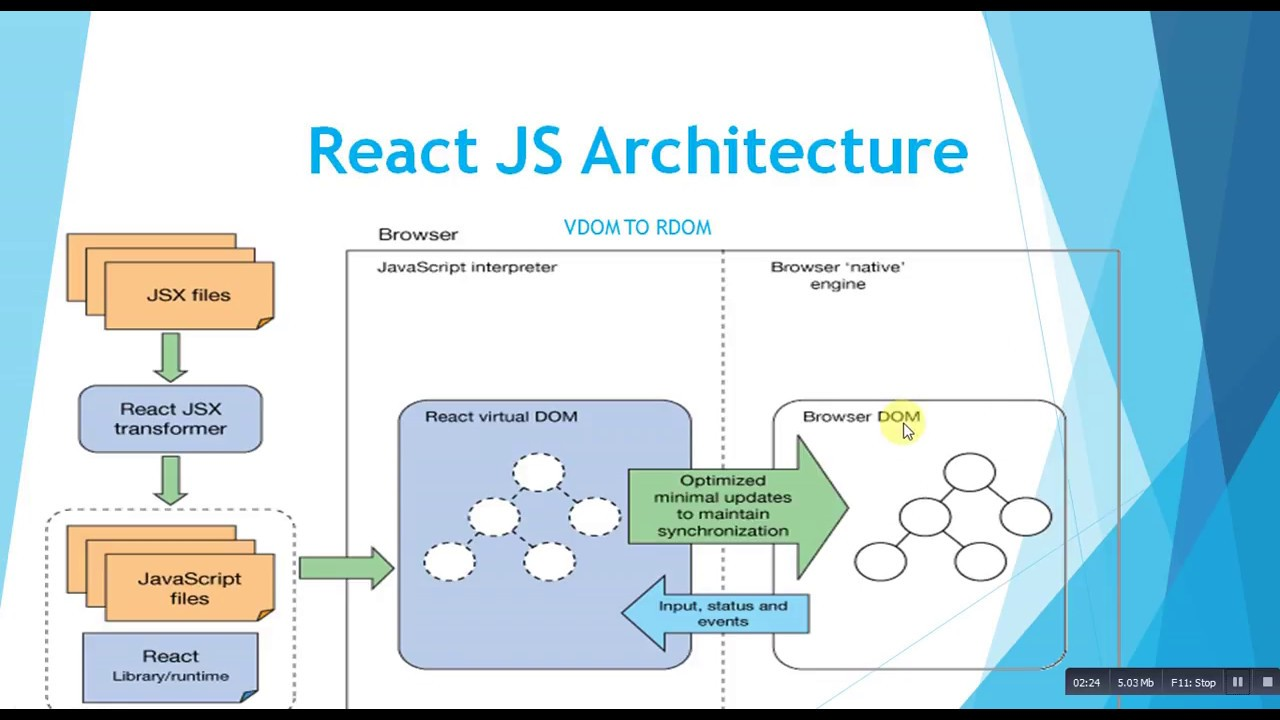


Figure 1: React JS Architecture

The code for the application is initially written in a JSX format. The React JSX transformer transform the files into Javascript files and the React library. We then go onto the virtual DOM, which is a representation of the UI is stored in memory to contain the ideal state of components. Once the React knows the state of a component it makes sure to Browser DOM is matching. This means minimal updates are needed as only changing components are affected.

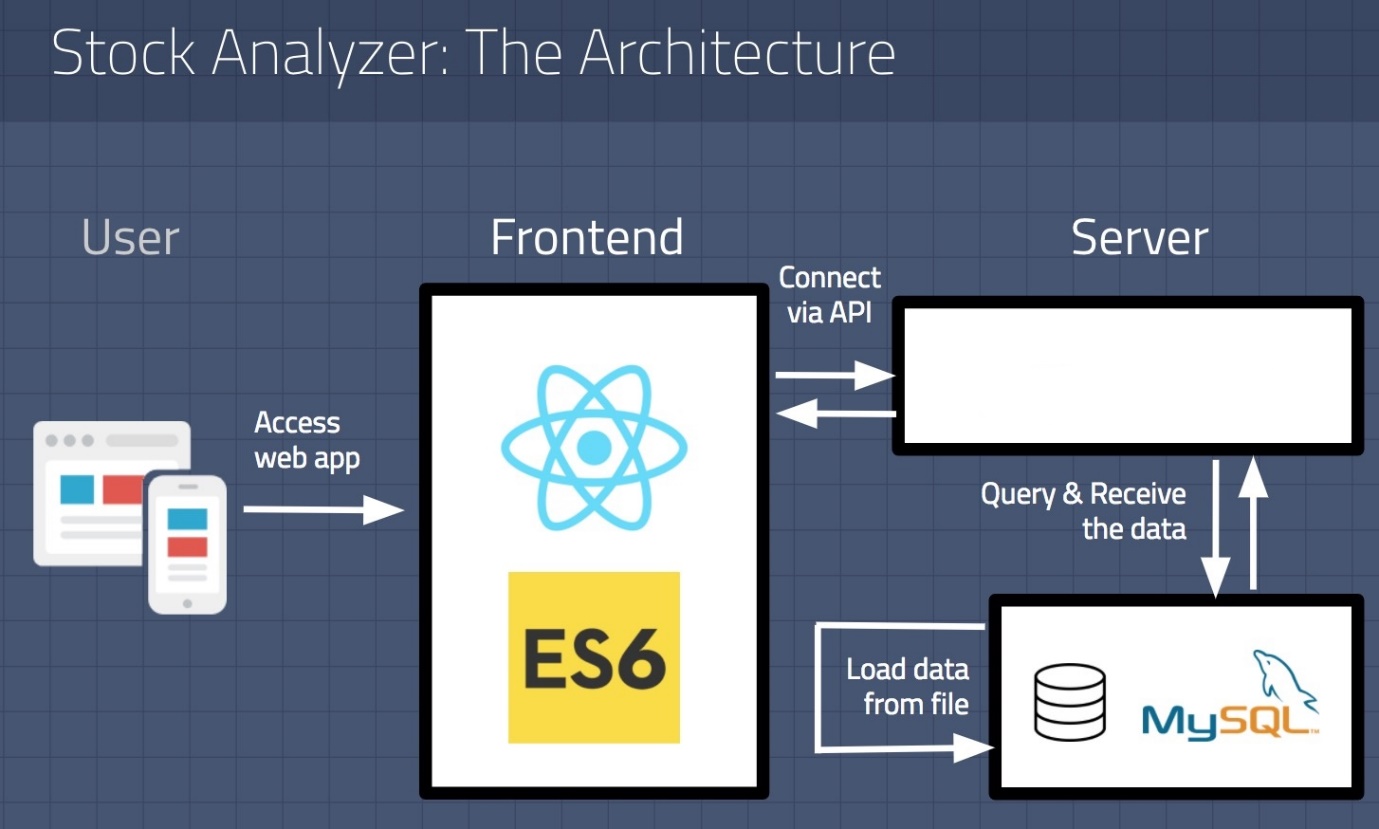
Figure 1 is a visual representation of the potential architecture of the react web application. However, unlike most user interfaces and frameworks, React does not enforce an architecture pattern on its users. It is essentially just a view that caters to user interfaces. Under the user interface is a tree of components. All components in React can hold a state which is the data that needs to be tracked to make the application function.

Figure : How React communicates to DB

Figure 2 is an image that describes how React communicates with the database. Firstly once the application is up and running, and the user accesses the web application, React will send requests to the server. This will be through an API call and most likely be through HTTPS or possibly other protocols. The server will then process this request and conducts a database operation based on the request. The database will then returns the data that was requested by the server or throws an error if anything goes wrong. Finally, the server sends back the requested data to the react application for it to be displayed or accessed by the user.

## Redux

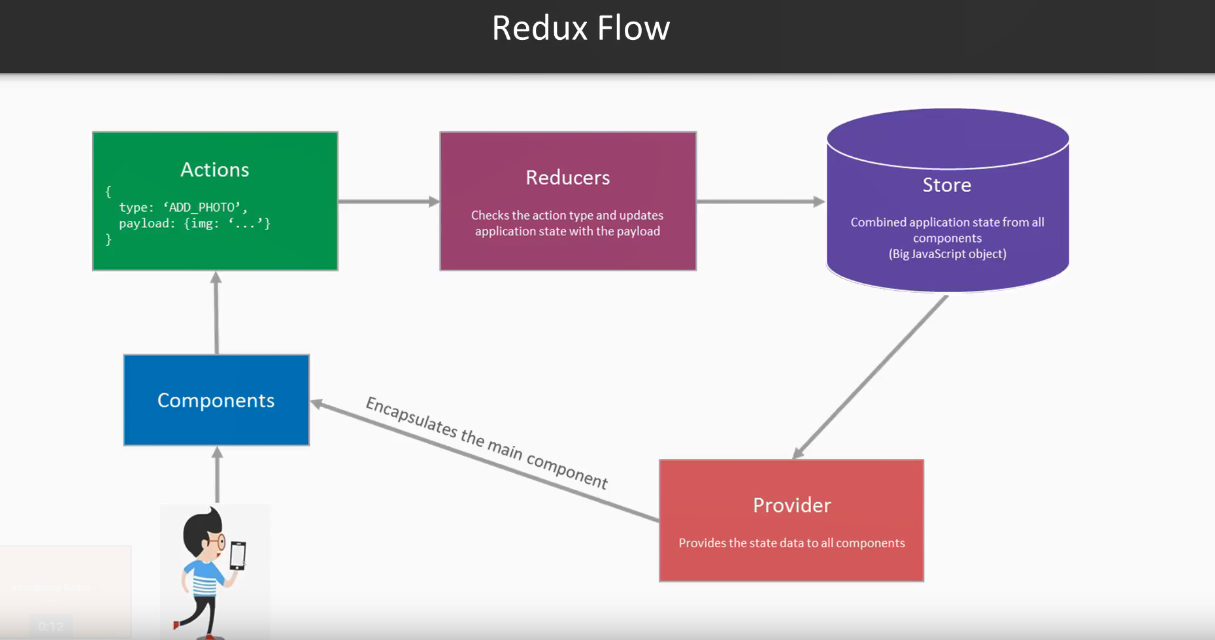
“Redux is a predictable state container that is used for JavaScript applications” [8]. Redux allows you to create applications that behave consistently, run on a range of different environments and makes it very easy to test. Redux is centralised meaning that the applications logic and state centralising enables influential abilities like undo/redo and state persistence. Redux has a great debug system called Redux DevTools. The tool allows you to trace when, where, why and how the application changes state. Redux’s architecture lets you log changes and sends complete error reports back to the server. Redux is flexible as it can work with any user interface layer and has a broad range of ecosystem of addons. “React Redux is designed to work with the React component model” [9]. Once how to extract the values for components that are needed by Redux is obtained, components store the values as props. React Redux also makes wrapper components that achieve the store interaction logic automatically. React Redux also has a complicated performance optimisation which ensures that components only re-renders when data has changed.

Figure : Why is React so important for State Management in React

Figure 3 is a visual representation of how Redux works for React to ensure sufficient state management. When an operator interacts through the web application, and a state change occurs in a component, it will trigger an action. These are direct javascript objects that send data from the application to the store. The Reducers stipulate how the application’s state was modified, based on the response to actions sent to the store. This depends on the action type. The store is what brings everything together. It allows access and modifications to the state. The provider is what distributes the data from the store to the individual components.

## Bootstrap

“Bootstrap is an open-source CSS framework that helps create responsive web applications” [10]. It contains design, navigation, forms, templates and typography for buttons and other interface components that are expected in a web application. React-Bootstrap replaces the Bootstrap JavaScript to create comprehensive looking designs on React web applications.

## MySQL

MySQL is a popular and well designed open-source relational database system that has a range of different features. There is a range of database system choices, but several aspects of MySQL make it stand out. One of the main benefits is that it comes with a high level of security. The DBMS comes with installed with a script that helps improve safety by removing unidentified accounts, defining a password for the root user, setting the installation’s password security level, and eliminating test records that are, by default, available to all operators. “MySQL supports user management and allows you to grant access privileges on a user-by-user basis” [11] . “MySQL’s scalability is unmatched, MySQL offers unmatched scalability to facilitate the management of deeply embedded apps using a smaller footprint even in massive warehouses that stack terabytes of data” [12]. Another option could have been to use Oracle, which has a high level of versatility and performance. However, “Oracle is a complex database to use and would require significant time and effort to use efficiently” [13].

## Auth0

Auth0 is a service that enables you to add both authentication and authorisation to your application. It helps you not worry about the security risk that comes from developing these security aspects manually. Each time a user tries to authenticate themselves, Auth0 will verify this user and send back the required information to the application. The Auth0 configuration within the application that comes with this project allows users to log in and authenticate themselves. “Auth0 will also enable users to register, logout, help retrieve user information, protect application routes and call protected endpoints from an API” [14].

# Planning and Timescales

In this project, I will adopt an Agile Methodology. This methodology mainly consists of the solutions and demands in the project being able to evolve and adapt to what the desired output of the project is. This also means that the project will be working in 2-week sprints.

|  |  |  |  |
| --- | --- | --- | --- |
| Objective | Description | Due Date | Sprint |
| Research into technologies that are to be used in the project. | Research into the frameworks, database systems, IDE’s that will be used to develop the application | 11th October | Sprint 0 |
| Research into similar applications | Looking at different applications and websites to learn about the benefits and drawbacks | 25th October | Sprint 0 |
| Research into recommendation algorithms. | Looking into the benefits and drawbacks of different recommendation systems. | 1st November | Sprint 0 |
| Develop a login feature | Create a feature to allow users to log into the application | 8th November | Sprint 1 |
| Develop register feature | Create a feature that allows users to register a user to the application | 22nd November | Sprint 2 |
| Interim Report |  | 24th November | Sprint 2 |
| Create a feature to allow users to edit their details | Users need to be able to keep their details up to date. | 6th December | Sprint 3 |
| Develop a feature to allow users to view a list of video games | A user needs to be able to view a list of video games to allow them to find out more about their ones of interest. | 20th December | Sprint 4 |
| Develop a feature where users can view details on a video game | A user should be able to view a range of specific information of a video game of their choice | 20th December | Sprint 4 |
| Develop a feature to allow users to upload content | A user should be able to upload their content. This could include videos and photos. | 3rd January | Sprint 5 |
| Develop a feature to allow users to like posts. | A user should able to like a post that has been uploaded by users. | 3rd January | Sprint 5 |
| Develop a feature where users and leave comments on a post. | A user should able to leave a comment on a post that has been uploaded by users | 3rd January | Sprint 5 |
| Develop a feature where users can leave a review on a game | A user should be able to leave a written review on a range of video games | 17th January | Spring 6 |
| Develop a feature where users can view both critic and user reviews. | A user should be able to view user reviews and critic reviews. | 17th January | Sprint 6 |
| Interview |  | 26th February |  |
| Dissertation & Software System |  | 29th April |  |
| Mini Viva |  | 21st May 2021 |  |

# Bibliography

|  |  |
| --- | --- |
| [1] | “IGN,” [Online]. Available: https://www.ign.com/uk. [Accessed 13th November 2020]. |
| [2] | “IMDb,” [Online]. Available: https://www.imdb.com/. [Accessed 13th November 2020]. |
| [3] | L. E. Molina, “Recommendation System for Netflix,” Vrije Universiteit Amsterdam Faculty of Science Business Analytics, 2018. |
| [4] | J. G. D. P. Germán Cheuque, “Recommender Systems for Online Video Game Platforms: the Case of STEAM,” Companion The 2019 World Wide Web Conference, 2019. |
| [5] | J. A. E. S. Paul Covington, “Deep Neural Networks for YouTube Recommendations,” ACM, 2016. |
| [6] | X. L. a. D. Li, “An Improved Collaborative Filtering Recommendation Algorithm and Recommendation Strategy,” Mobile Information Systems, 2019. |
| [7] | “React - A JavaScript library for building user interfaces,” React, [Online]. Available: https://reactjs.org/. [Accessed 14th November 2020]. |
| [8] | “Redux - A Predictable State Container for JS Apps,” Redux, [Online]. Available: https://redux.js.org/. [Accessed 14th November 2020]. |
| [9] | “React Redux - Official React bindings for Redux,” React Redux, [Online]. Available: https://react-redux.js.org/. [Accessed 14th November 2020]. |
| [10] | “Build fast, responsive sites with Bootstrap,” Bootstrap, [Online]. Available: https://getbootstrap.com/. [Accessed 14th November 2020]. |
| [11] | O. a. M. Drake, “SQLite vs MySQL vs PostgreSQL: A Comparison Of Relational Database Management Systems,” Digital Ocean, 19 March 2019. [Online]. Available: https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems. [Accessed 14th November 2020]. |
| [12] | “8 Major Advantages of Using MySQL,” Datamation, 16 November 2016. [Online]. Available: https://www.datamation.com/storage/8-major-advantages-of-using-mysql.html. [Accessed 18th November 2020]. |
| [13] | D. Sullivan, “Advantages & Disadvantages of Oracle SQL,” Techwalla, 9 February 2017. [Online]. Available: https://www.techwalla.com/articles/advantages-disadvantages-of-oracle-sql. [Accessed 14th November 2020]. |
| [14] | D. Arias, “The Complete Guide to React User Authentication with Auth0,” Auth0, [Online]. Available: https://auth0.com/blog/complete-guide-to-react-user-authentication/. [Accessed 14th November 2020]. |