BuzzBeeper

Anusha Venkapally Santosh, Riddima Tiwari, Sayali Patkar, Tanya Gupta,Wiliams Babajide Samuel

Introduction

The BuzzBeeper is a MEAN stack web application built by a team of five to fulfill the requirement of the course “Advanced Software Programming”. BuzzBeeper is a television show tracker which gives information to users regarding their desired shows, all in one place. This web application makes it easier for the users to follow their favorite television shows. This application gives a detailed information of the show when it is subscribed by the user.

Abstract

BuzzBeeper is a dynamic web application which has the information of all television programs that are broadcasted on various television channels. This information would be very useful to viewers who would like to know when and where the programs are being telecasted. There is also an option for the users to make subscriptions to their favorite channels along with customer support. This application is built on MEAN stack.

Design : MVC

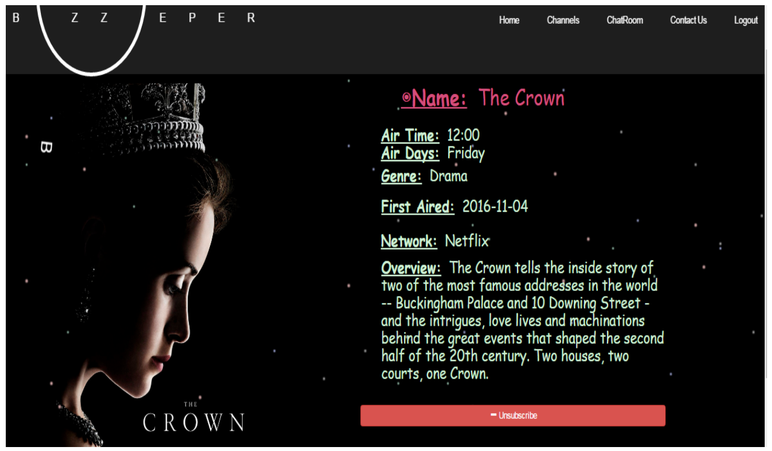
This Project is developed based on Model View Controller. Model View Controller or MVC as it is popularly called, is a software design pattern for developing web applications. It is made up of the following three parts:

The Model: The model is responsible for managing the data of the application.

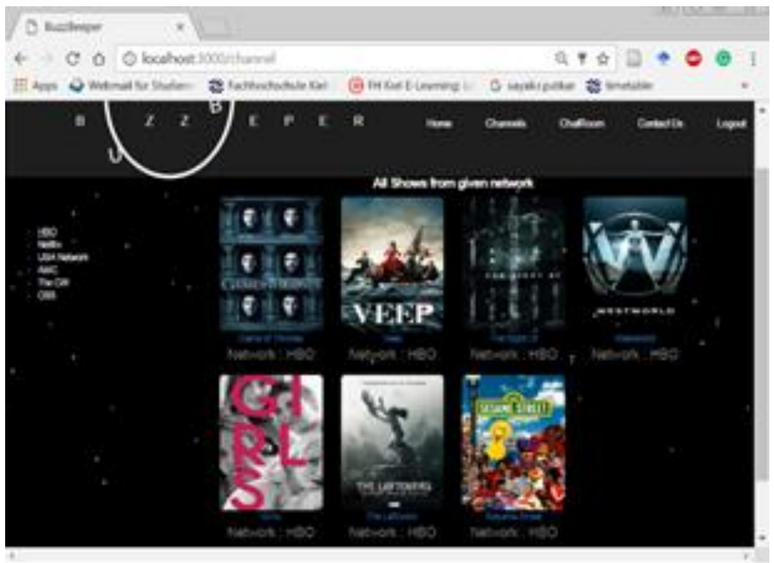
The View: A presentation of data in a particular format, triggered by a controller's decision to present the data.

The Controller: The controller is responsible for responding to user input and perform interactions on the data model objects.

Result



Show Page



Channels Page

Areas to be focused

Show Page: The channel page list of the channels and by default all HBO shows are displayed. The user is redirected to the show page when he clicks on the show from any channel. This page displays all the details which are associated with the show. These pages are visible to any user. Only the authenticated user has the capability to subscribe to the show and unsubscribe.

Querying & Parsing the TDVB API: One of the admin functionalities is to add shows to the website from the various television channels. For this purpose, an “add” page is created whose link is not added anywhere in the application. We have obtained the TVDB API key from the source which takes the query and fetches the required data. To normalize all the tags, we have used the “xml2js” parser. BuzzBeeper uses asynchronous waterfall model to handle the various operations involved in fetching the data.

Authentication: For Authentication tasks, i.e. Login, Logout & Signup, BuzzBeeper uses Firebase Authentication. Wherein Password matching & session keeping is basically taken care of by Firebase.

Conclusion

The Project was indeed a very interesting and enjoyable experience. The project is still in an experimental stage but introduces new promising concepts. This web application, if deployed in the future, would definitely serve the purpose of it being created. We are looking forward to make this as a mobile application along with enabling the video services, which we consider as our future work.