MINI PROJECT

(2021-22)

Submitted By -

Abhyuday Chaturvedi(191500044)

Aryan Jaiswal (1915000158)

Divyam Goel (191500272)

Pratik Sharma(191500593)

Prattyush Srivastava (191500594)

Under the Supervision Of

Mr. Akash kumar Choudhary

Assistant Professor

Department of Computer Engineering & Applications

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

Declaration

I/we hereby declare that the work which is being presented in the Bachelor of technology. Project **"BucketList"**, in partial fulfillment of the requirements for the award of the ***Bachelor of Technology*** in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of my/our own work carried under the supervision of **Mr. Akash kumar choudhary Assistant Professor, Dept. of CEA,GLA University.**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

Sign: *Abhyuday Chaturvedi*

Name of Candidate: Abhyuday chaturvedi

University Roll No.:191500044

Sign: *Aryan Jaiswal*

Name of Candidate: Aryan Jaiswal

University Roll No.:191500158

Sign: *Divyam Goel*

Name of Candidate: Divyam Goel

University Roll No.:191500272

Sign: *Pratik Sharma*

Name of Candidate: Pratik Sharma

University Roll No.:191500593

Sign: *Prattyush Srivastava*

Name of Candidate: Prattyush Srivastava

University Roll No: 191500594

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

Certificate

This is to certify that the project entitled "BucketList", carried out in Mini Project – I Lab, is a bonafide work by Aryan Jaiswal, Abhyuday chaturvedi, Prattyush Srivastava, Pratik Sharma and Divyam Goel and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

Signature of Supervisor:

Name of Supervisor: Mr. Akash kumar choudhary

Date:

Training Certificates

- **Abhyuday Chaturvedi**

GOLD

GLA UNIVERSITY MATHURA

Accredited with A Grade by NAAC

Job Oriented Value-Added Course

on

Hybrid Mobile App Development with React-Native

Certificate of Participation

This certificate is awarded to **ABHYUDAY CHATURVEDI**, Roll No. **191500044** from **GLA University, Mathura**, for successfully completing the *Job Oriented Value-Added Course* on **Hybrid Mobile App Development with React-Native**, organized by Department of Computer Engineering & Applications (CEA) GLA University, Mathura (U.P.), India from **29th June, 2021** to **31st July, 2021**.

Dr. Rohit Agarwal
Head of Department

Mr. Pankaj Kapoor
Course Coordinator

- **Aryan Jaiswal**

GOLD

GLA UNIVERSITY MATHURA

Accredited with A Grade by NAAC

Job Oriented Value-Added Course

on

Hybrid Mobile App Development with React-Native

Certificate of Participation

This certificate is awarded to **ARYAN JAISWAL**, Roll No. **191500158** from **GLA University, Mathura**, for successfully completing the *Job Oriented Value-Added Course* on **Hybrid Mobile App Development with React-Native**, organized by Department of Computer Engineering & Applications (CEA) GLA University, Mathura (U.P.), India from **29th June, 2021** to **31st July, 2021**.

Dr. Rohit Agarwal
Head of Department

Mr. Pankaj Kapoor
Course Coordinator

- **Divyam Goel**

GOLD

GLA UNIVERSITY MATHURA

Accredited with A Grade by NAAC

Job Oriented Value-Added Course

on

Hybrid Mobile App Development with React-Native

Certificate of Participation

This certificate is awarded to **DIVYAM GOEL**, Roll No. **191500272** from **GLA University, Mathura**, for successfully completing the *Job Oriented Value-Added Course* on **Hybrid Mobile App Development with React-Native**, organized by Department of Computer Engineering & Applications (CEA) GLA University, Mathura (U.P.), India from **29th June, 2021** to **31st July, 2021**.

Dr. Rohit Agarwal
Head of Department

Mr. Pankaj Kapoor
Course Coordinator

- **Pratik Sharma**

udemy

Certificate no: UC-6cff4e03-5e7e-4b5f-b801-fa654e6b7aca
Certificate url: ude.my/UC-6cff4e03-5e7e-4b5f-b801-fa654e6b7aca
Reference Number: 0004

CERTIFICATE OF COMPLETION

The Complete React Native + Hooks Course

Instructors **Stephen Grider**

Pratik Sharma

Date **Nov. 15, 2021**
Length **38.5 total hours**

- **Prattyush Srivastava**

udemy

Certificate no: UC-c0161afd-c70c-4db5-9d73-cae1aa0f7e9c
Certificate url: ude.my/UC-c0161afd-c70c-4db5-9d73-cae1aa0f7e9c
Reference Number: 0004

CERTIFICATE OF COMPLETION

React Native - The Practical Guide [2022 Edition]

Instructors **Academind by Maximilian Schwarzmüller, Maximilian Schwarzmüller**

Prattyush Srivastava

Date **Nov. 15, 2021**
Length **33 total hours**

Department of Computer Engineering and Applications

GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,

Chaumuha, Mathura – 281406 U.P (India)

ACKNOWLEDGEMENT

Presenting the ascribed project paper report in this very simple and official form, we would like to place our deep gratitude to GLA University for providing us the instructor Mr Akash Kumar Choudhary, our Assistant Professor and supervisor.

He has been helping us since Day 1 in this project. He provided us with the roadmap, the basic guidelines explaining on how to work on the project. He has been conducting weekly meetings to check the progress of the project and providing us with the resources related to the project. Without his help, we wouldn't have been able to complete this project.

And at last but not the least we would like to thank our dear parents for helping us to grab this opportunity to get trained and also my colleagues who helped me find resources during the training.

Thanking You

Sign: *Abhyuday Chaturvedi*

Name of Candidate: Abhyuday Chaturvedi

University Roll No.:191500044

Sign: *Aryan Jaiswal*

Name of Candidate: Aryan jaiswal

University Roll No.:191500158

Sign: *Divyam Goel*

Name of Candidate: Divyam Goel

University Roll No.:191500272

Sign: *Pratik Sharma*

Name of Candidate: Pratik Sharma

University Roll No.:191500593

Sign: *Prattyush Srivastava*

Name of Candidate: Prattyush Srivastava
University Roll No: 191500594

ABSTRACT

In this project, we are creating an react application, which we have named BucketList. This application will provide us a platform to access the movies and webseries we want to watch at the ease of our fingertips. All the users will be having their separate accounts on this app which will be connected to their email id. Any content that the user wishes to watch will be entered by him in the search box which works on the basis of queries input. Apart from searching the content online, the user can also see the glimpse of the content by hovering over the content he/she visits. The app is suitable in the present scenario as the world is being digitalized and teenage group also likes these sites. The app will be completely efficient and transparent to the reviews of the people. To get more details about the movie one can click on the movie block and get further grave details. This app will be using API for providing all the names of movies. Further the FAQ section provides all the necessary details that the user may need about the app. The app also has a complete User Interface attached to the firebase a perfect login system with email id and password and a forget password too.

React App ecosystem is diverse and is changing people's life all over the world. Mobile users are expected to increase because of the advance changes of the operating system and the way it deals with issues and compatibility with other mobile devices. Furthermore designing solutions for the problems that we may face in future is essential. Like this application definitely stands the need of students at any time at their fingertips without any barrier of place.

INTRODUCTION

1.1 CONTEXT

This React Application "BucketList" has been submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering at GLA University, Mathura supervised by Mr.Akash Lumar Choudhary. This project has been completed approximately three months and has been executed in modules, meetings have been organised to check the progress of the work and for instructions and guidelines.

1.2 MOTIVATION

In the recent years, we have realized the craze of webseries & movies among youths which are available online free to watch .Mobile devices have been the greatest source of watching all the content while and having them at the reach of our fingertips would be an opportunity hardly any youngster would afford to miss.

In the century we are living the world is progressing at a really great pace, a lot number of technologies come up every single day. To keep up with the technology is also important to survive in this world of digitalization and learning. Along with this we need to have a place to keep the resources for areas of our interest so we thought of developing a app which could provide us with free virtual content to watch as well as a platform where we could keep these conetent we like marked.

Moreover this kind of application can be used in malls/cafes to show them content for free . This would be an excellent effort to provide free content without any boundaries to all.

1.3 OBJECTIVE

The main objective of this application is that it will provide recommendation from the choices fill by the user, it will prompt some genre of items which have to be filled by user and recommendation will be based on those choices

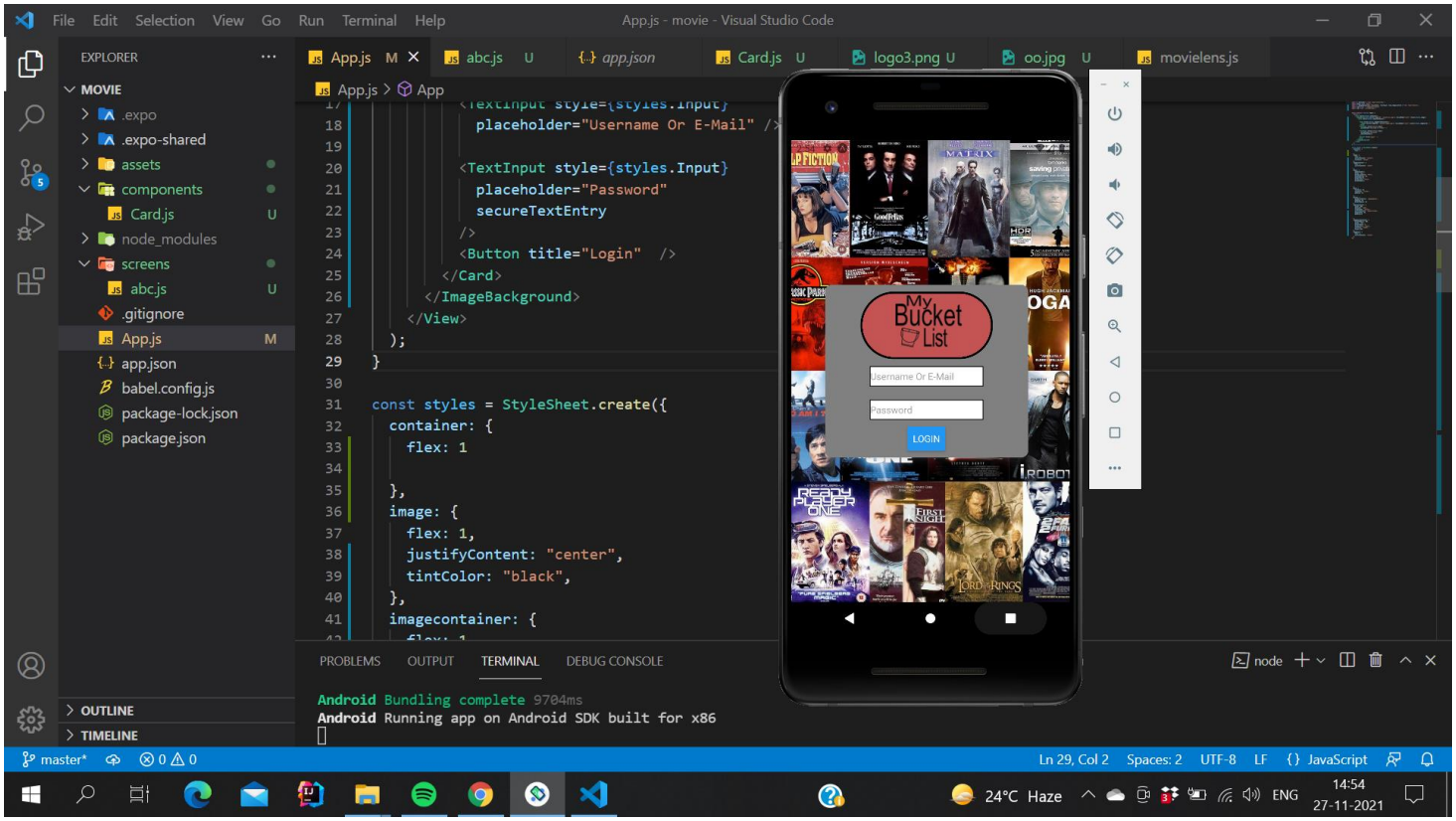
This application developed can be used at a variety of places, at cafes and have its significance. The goal of the app was to provide a free area where user can know what content he is watching and what to watch.

1.4 EXISTING SYSTEM

In the present scenario, we are dealing with the manual searching of movies from thousands of other places present in Google Movies API. With the help of this application we are able to find a place where we can easily find the Movies with the help of keywords. As this idea as already implemented here are the some snap how our application will look.

As soon as the user enters the app, there will be landing page containing the name of the app and then there will be a login /signup page .Initially there will be search bar as shown in the image below. Then on the basis of certain keywords the app will fetch the results and the movies will be displayed as shown in the second screenshot. In this we will add a feature to bookmarks the movie or add to favourites as this will be helpful when the user try to find the same movie again.

React Application - Bookopedia



SOFTWARE REQUIREMENT ANALYSIS

How Virtual Content ruling the world

One of the industries that have seen massive growth in the past few years is that of the OTT platforms and the global outbreak of the COVID-19 pandemic has been proved as a boon for it. While other industries have been adversely affected by the pandemic, the OTT platforms have walked the path otherwise. With social distancing being the new normal and the theatres and multiplexes being closed, people confined in their houses are left with very few options of entertainment. OTT platforms have played a major role in entertaining people amidst these critical times and have clearly taken over the other modes of entertainment. This blog will give you a detailed insight into this booming industry and the reasons for its popularity.

According to a survey conducted by the mobile marketing platform InMobi in April 2020, 46% of viewers are engrossed in watching online content. Added to this, experts also suggest that the popularity of these platforms could grow further in the future, as television channels may run out of content. According to Paritosh Joshi, Media Consultant and Principal, Provocateur Advisory, "As these channels have been unable to shoot due to the lockdown, they are running old content. In such a scenario, people will gravitate towards OTT to watch fresh content."

PROBLEM STATEMENT

The platform App "BucketList" is an React Application which will allow the users to search any launched movies & web series. Actually this app is connected to the API which shows movies collections. The user searches for the content.

Along the side, for the users a rating section is being provided for them to review the show they like it or not. As a help to the users, there will be FAQ Section containing the basic details and guidelines on how to use the app. The profile of each user is created and can be updated anytime as per the requirements. One another feature that our app also shows the current IMDB rating of

show by using an API .

HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirement

- Processor :intel
- Operating System :Any Operating System
- RAM : 8 GB (or higher)
- Hard disk : 256GB

Software Requirement

- Software used: Visual Studio
- Language used : React Native , JS
- Database: Firebase
- User Interface Design : Android Application and Website

TECHNOLOGY USED

REACT

React (also known as **React.js** or **ReactJS**) is a free and open-source front-end JavaScript library for building user interfaces or UI components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile

applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.:-

- **Native Apps:** An executable program coded in the machine language of the hardware platform it is running in. **Native applications** are compiled into the machine language of that CPU. For example, **Windows** and Mac executable **apps** are in x86 machine language, while **mobile apps** are ARM based. Native apps are the most common. They're coded in a specific language like Swift for **iOS** or Java for Android. A popular example is WhatsApp.
- **Web Apps:** are accessed via the internet browser and will adapt to whichever device you're viewing them on. They are not native to a particular system, and don't need to be downloaded or installed. Due to their responsive nature, they do indeed look and function a lot like mobile apps — and this is where the confusion arises.
- **Hybrid Apps: Hybrid apps are deployed in a native container that uses a mobile Web View object. When the app is used, this object displays web content thanks to the use of web technologies (CSS, JavaScript, HTML, HTML5).It is in fact displaying web pages from a desktop website that are adapted to a Web View display**

BASIC TERMINOLOGY

- **Layout:** Layout is the parent of view. It arranges all the views in a proper manner on the screen.
- **Activity**: An activity can be referred as your device's screen which you see. User can place UI elements in any order in the created window of user's choice.

- **View**: A view is an UI which occupies rectangular area on the screen to draw and handle user events.

- **Emulator**: An emulator is an Android virtual device through which you can select the target Android version or platform to run and test your developed application.

- **Manifest file**: Manifest file acts as a metadata for every application. This file contains all the essential information about the application like app icon, app name, launcher activity,

and required permissions etc.

- **API:** Short for Application Programming Interface. APIs are functions that developers can call on to access specific features by calling upon programs, code, and services that others have written. For example, if a developer wants to draw a button on the screen, she can insert a small bit of code that says "draw this kind of button, with this color and size and style, at this location" instead of dozens of lines of code that tells the graphics processor, in detail, exactly how to draw a button. If the application wants your location, it can use the location API to "get the device's location" and let Google's code handle the rest, instead of requiring the developer to build an entire location service from scratch just for her own app. There are thousands of APIs in Android, covering everything from drawing interface elements, to the cameras, to location access, to accessing storage, to 3D graphics (see: OpenGL ES) and much more.

- **Intent:** Intents are an essential part of the Android ecosystem. They are used to express an action to be performed. Intents allow you to interact with components from the same applications as well as with components contributed by other applications. It can be classified into implicit and explicit intents.

- **Implicit intent:** It does not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it.
- **Explicit Intent:** It specifies the component to start by name. You'll typically use an explicit intent to start a component in your own app, because you know the class name of the activity or service you want to start.

- **APK**: Short for "Android application package." The extension used in Android app installation files (e.g., app.apk). Similar in nature to an EXE file on Windows.

- **SDK:** Short for "Software Development Kit." As it pertains to Android, the SDK is a set of tools such as code libraries, a debugger, and a handset emulator that can be run on Windows, Mac, or Linux to facilitate the creation of Android apps by developers. While the SDK is generally intended for use by developers, end users can install the software on their home computer to execute ADB and Fast boot commands.

- **Action Bar**: The action bar is an important design element, usually at the top of each screen in an app that provides a consistent familiar look between Android apps. It is used to provide better user interaction and experience by supporting easy navigation through tabs and drop-down lists.

- **Navigation bar**: Android Navigation Drawer is a sliding left menu that is used to display the important links in the application. Navigation drawer makes it easy to navigate to and fro between those links. It's not visible by default and it needs to opened either by sliding from left or clicking its icon in the Action Bar.

- **Fragment**: A Fragment represents a behavior or a portion of user interface in a Fragment Activity. You can combine multiple fragments in a single activity to build a multi-pane UI and reuse a fragment in multiple activities.

- **Firebase** is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google's infrastructure. Firebase is categorized as a NoSQL database program,

which stores data in JSON-like documents. Firebase has three core services: a real-time database, user authentication and hosting. With the Firebase iOS SDK, you can use these services to create apps without writing any server code.

JSON stands for JavaScript Object Notation. It is an independent data exchange format and is the best alternative for XML. JSON is used for data interchange (posting and retrieving) from the server. Hence knowing the syntax and it's usability is important. JSON is the best alternative for XML and its more readable by huma

IMPLEMENTATION AND USER INTERFACE

Creating an app concept design with screen sketches and functional flow diagrams is the best way to communicate your vision to the mobile app developer. Making the concept clear to the developer is probably the most important factor in successful mobile app development. Yet it is one of the most common problems or obstacles in a mobile app development outsourcing project.

No matter what the marketing and profit goals are or if you are outsourcing an app for your personal use, you need to fully design and document the app concept if you expect a programmer to make your vision a reality. Developers are not mind readers and even descriptions given during conversations can be very fleeting or interpreted differently. Fully documenting your concept, therefore, leaves little to chance. The two most important things to do are: A) make a comprehensive description of how the app works and what it does (functionality) and B) create a comprehensive description of what the user sees and does (look and feel).

Implementation of the BucketList:

Implementation of BucketList is taken place in various phases. Firstly we build the login interface then Navigation drawer i.e. make fragment for each of the list item using the Navigation view and the make various layout for the supporting features and connect the app with the API for fetch the required book. And finally we parse the object to get the data in the required format and then display the result.

Step to be followed to develop the app:

1. Firstly we create the splash screen with animated text using JS and linked it with the main

Activity through React

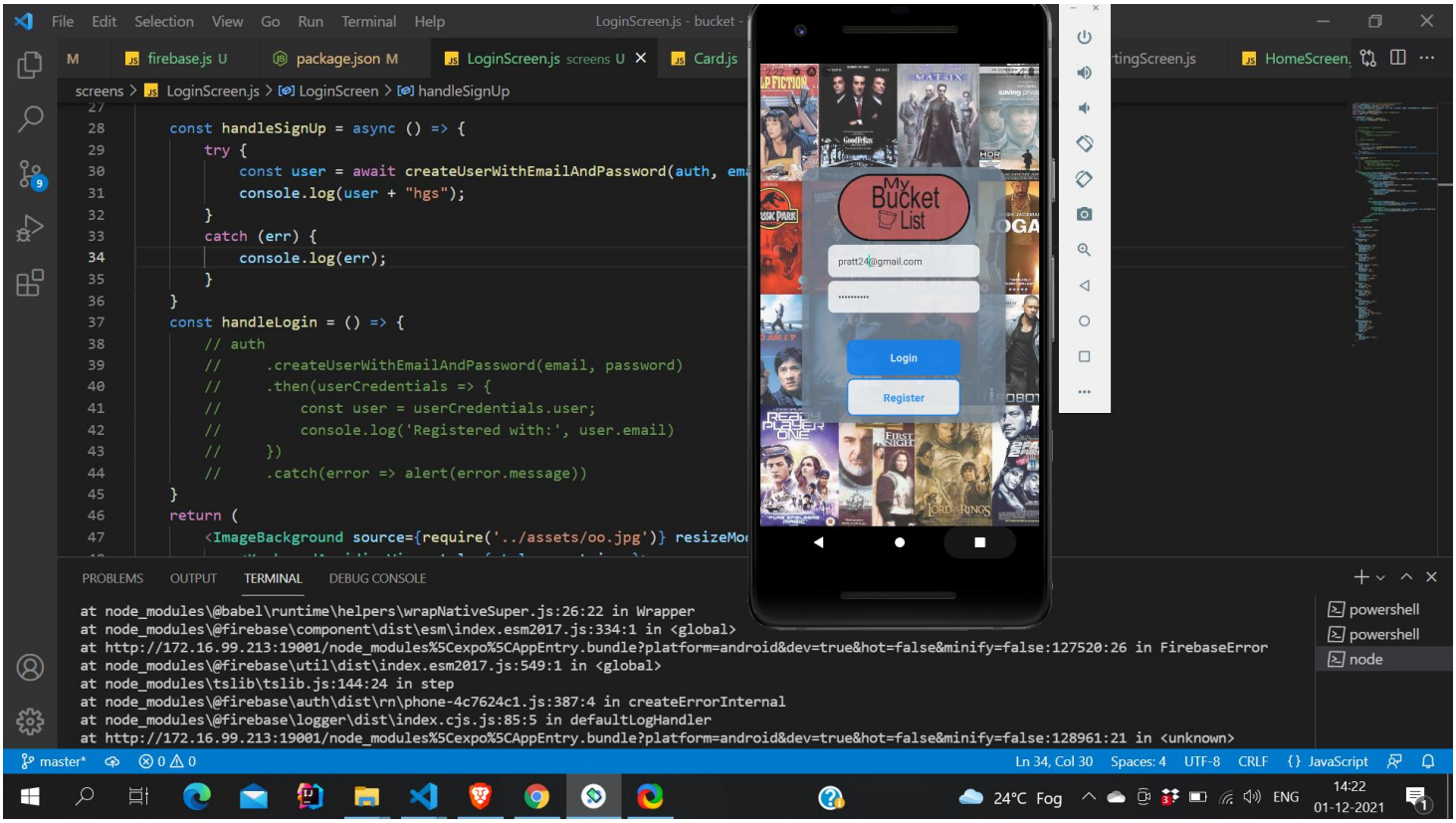
2. After that we create login phase which comprises of various phases that are mentioned below:
    - Login Page: allows user to login into the app if the user is existing one

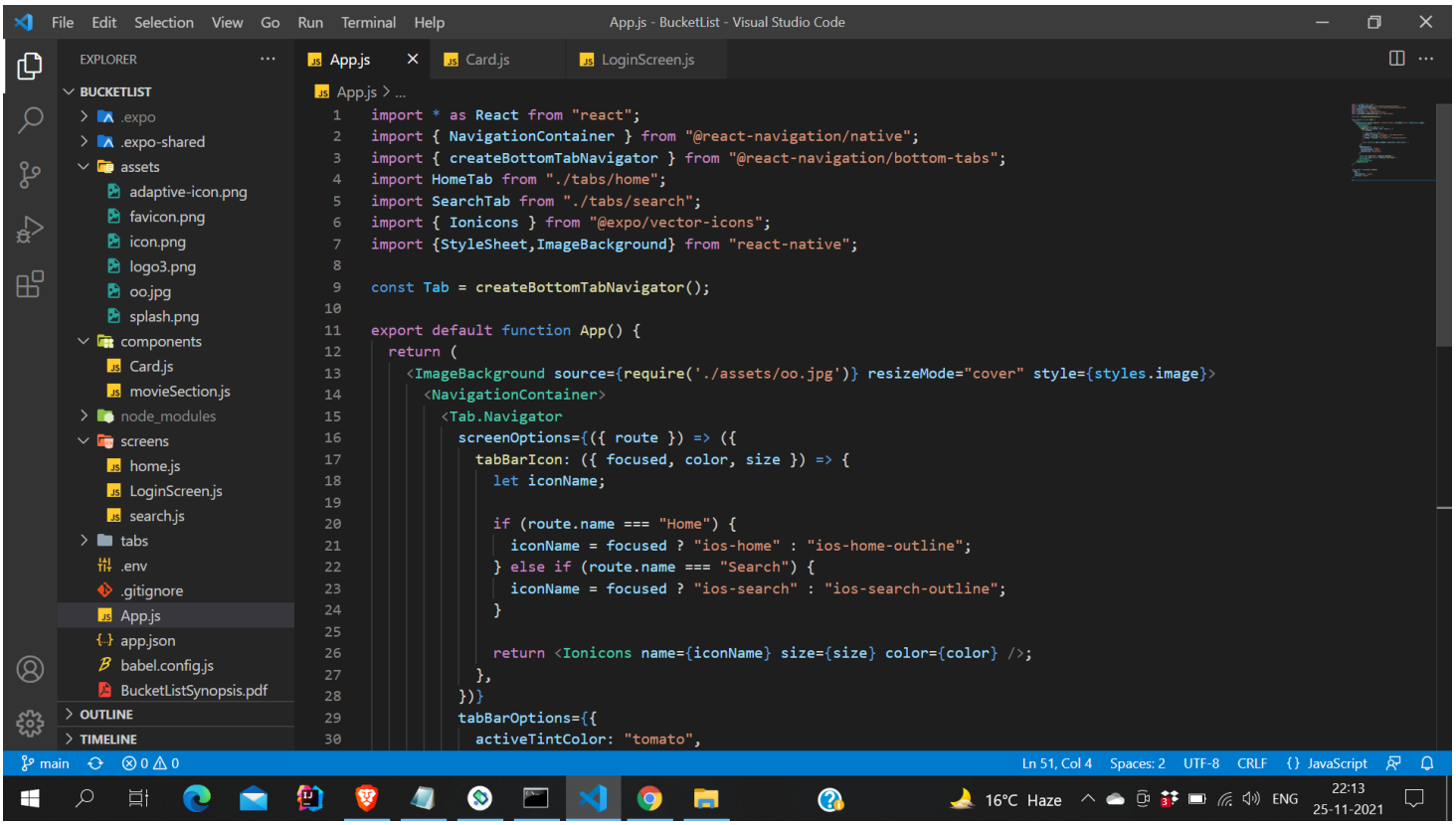
- **For authenticating the user we have used firebase authentication**.

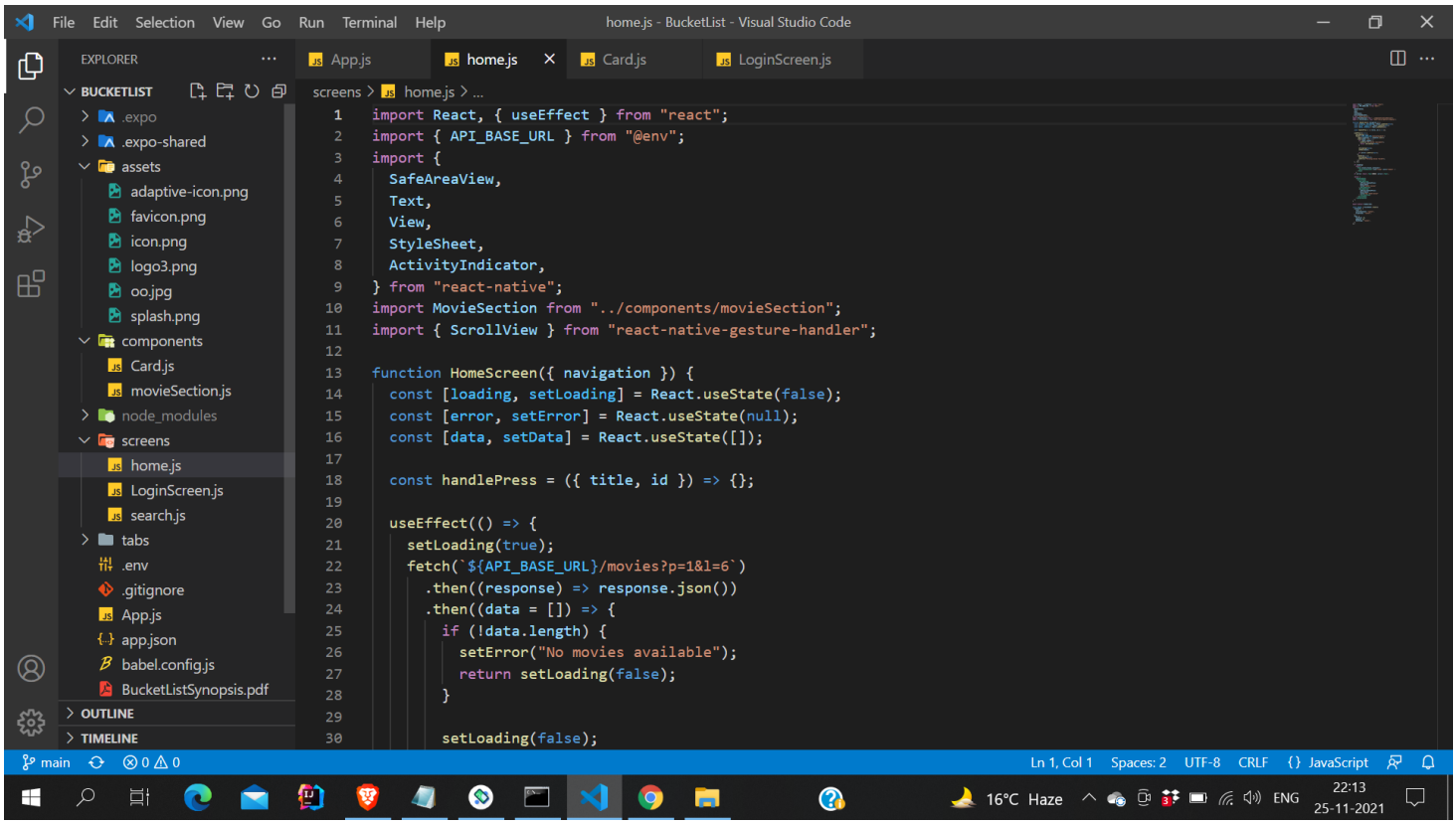
3. Now, we are going to create Navigation drawer for that purpose we have used following functionality of android:
    - Fragments(SupportFramentManager)
    - Menu – items
    - Drawer header
    - Hamburger icon
    - ActionBarDrawableToggle (help to create navigation Bar)
4. .Creating fragment for each of the menu item. Our Menu items are:
    - Dashboard
    - Profile
    - About App
    - FAQ
    - Favourites
    - Sign-Out
5. Now we have created various activities like Shows List, Shows Description and many more.
6. In this step we connect our app with the API .
7. Now we add data (that we have received from API) to the movie and series description activity .
8. In the description Activity there are various functionality. Some of them are mentioned below.
    - Watch Here: It will allow user to watch that show and redirect the user to the new page.
    - Favourites: it will add the shows to the favourites that you can watch to later.

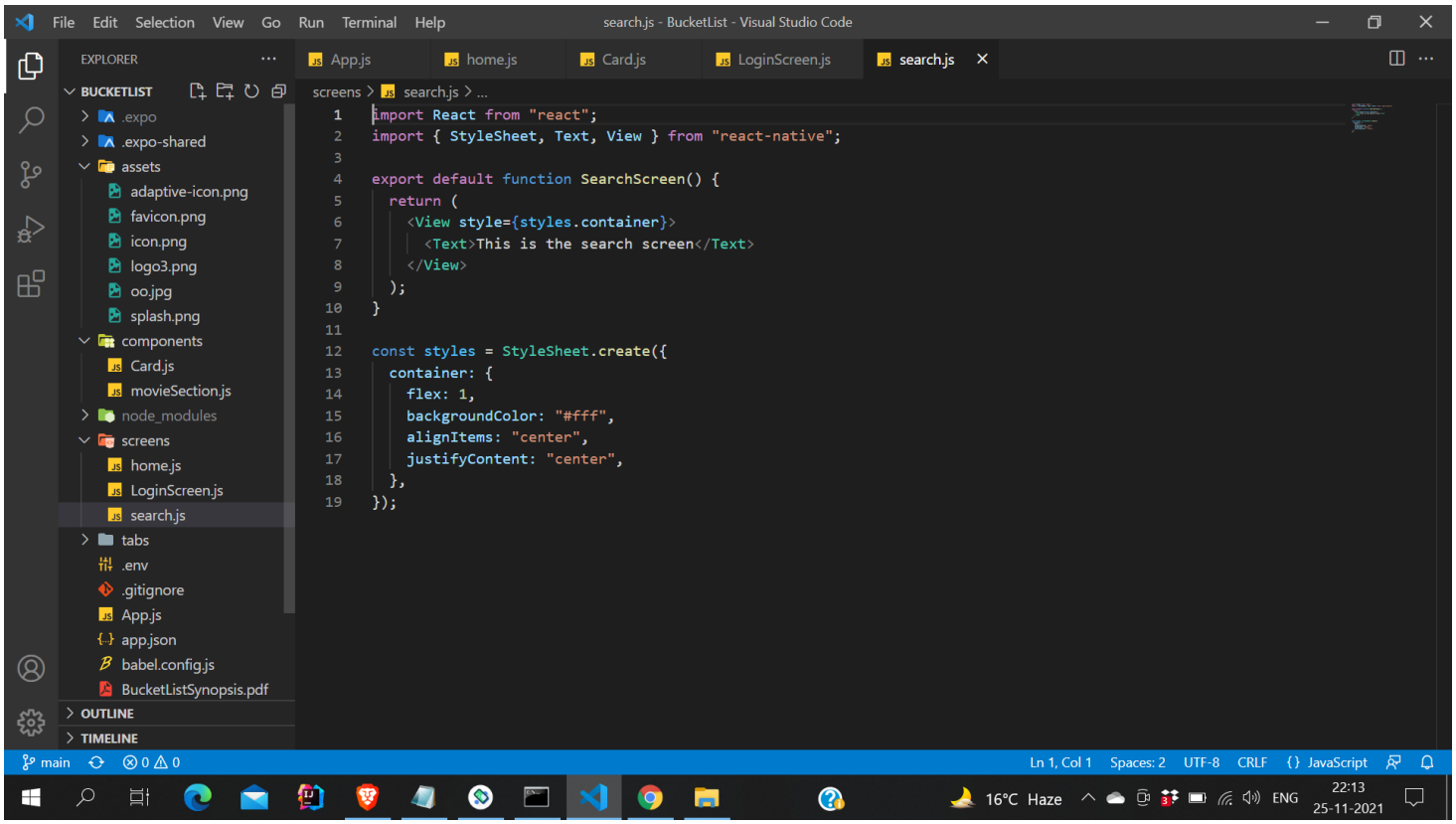
User Interface

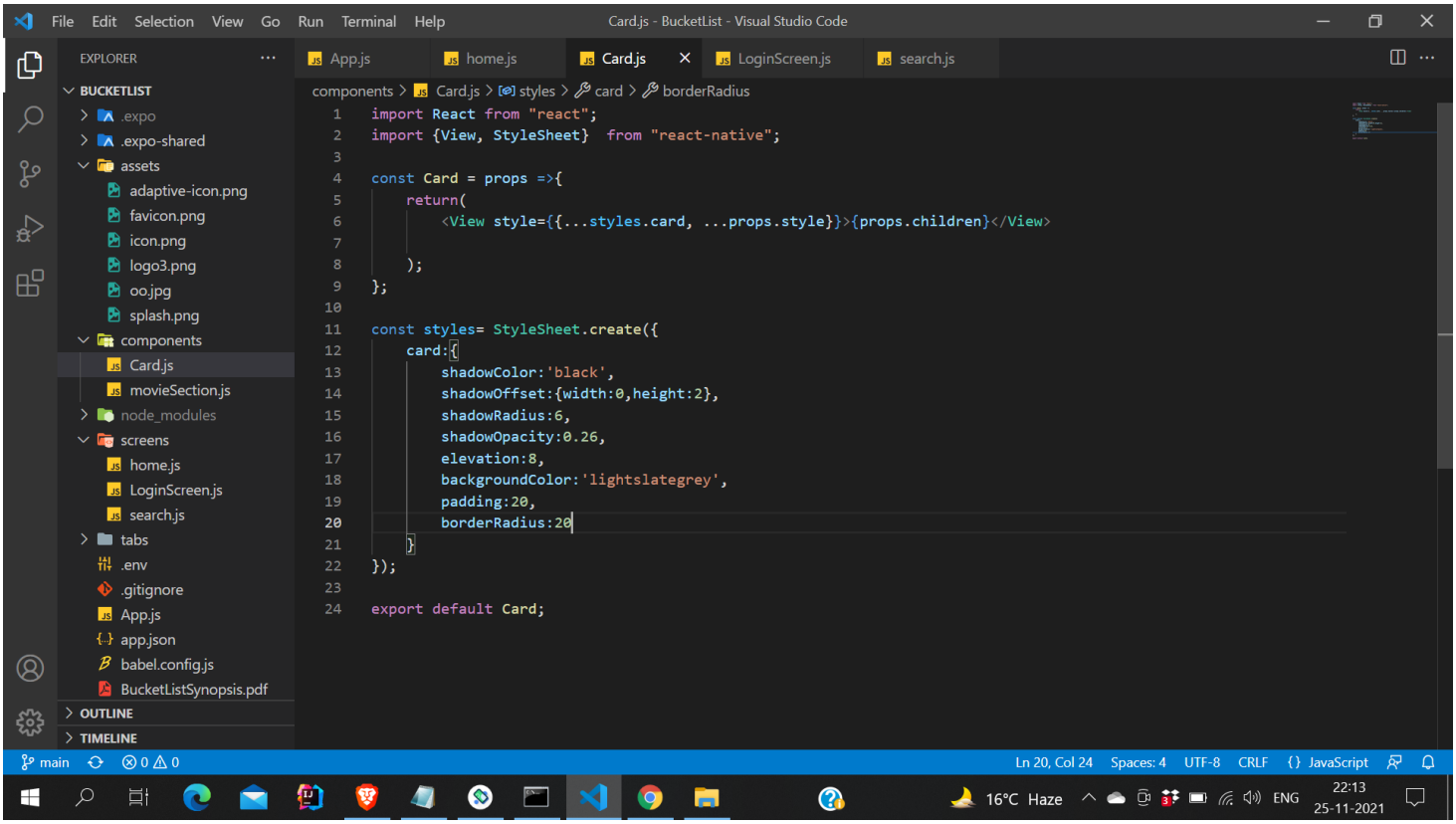
React Application

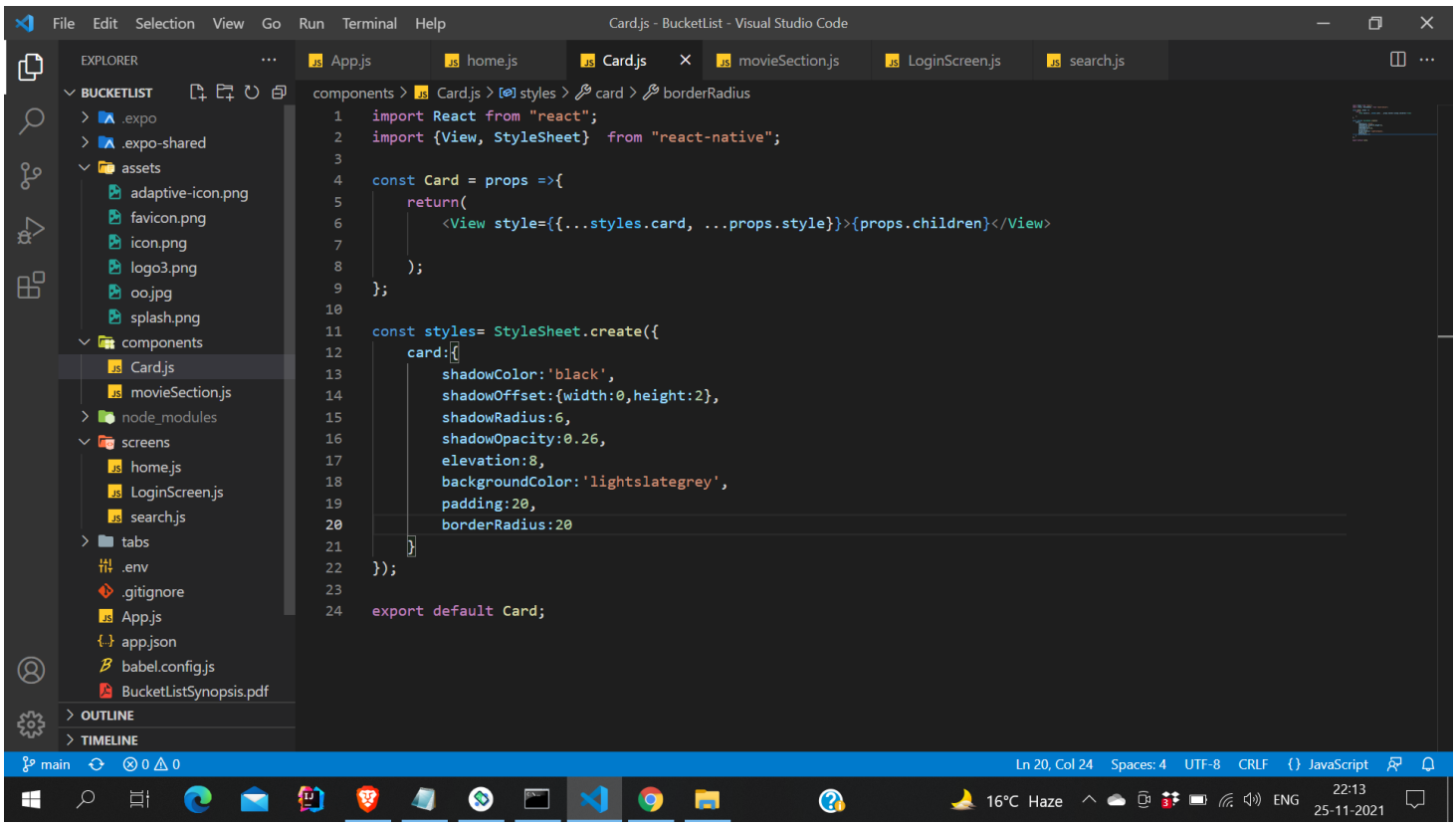



File Edit Selection View Go Run Terminal Help
App.js - BucketList - Visual Studio Code
EXPLORER
BUCKETLIST
App.js
Card.js
LoginScreen.js
import * as React from "react";
import { NavigationContainer } from "@react-navigation/native";
import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
import HomeTab from "./tabs/home";
import SearchTab from "./tabs/search";
import { Ionicons } from "@expo/vector-icons";
import {StyleSheet,ImageBackground} from "react-native";
const Tab = createBottomTabNavigator();
export default function App() {
return (
<ImageBackground source={require('./assets/oo.jpg')} resizeMode="cover" style={styles.image}>
<NavigationContainer>
<Tab.Navigator
screenOptions={({ route }) => ({
tabBarIcon: ({ focused, color, size }) => {
let iconName;
if (route.name === "Home") {
iconName = focused ? "ios-home" : "ios-home-outline";
} else if (route.name === "Search") {
iconName = focused ? "ios-search" : "ios-search-outline";
}
return <Ionicons name={iconName} size={size} color={color} />;
},
})}
tabBarOptions={{
activeTintColor: "tomato",
main
Ln 51, Col 4
Spaces: 2
UTF-8
CRLF
JavaScript
16°C Haze
ENG
22:13
25-11-2021

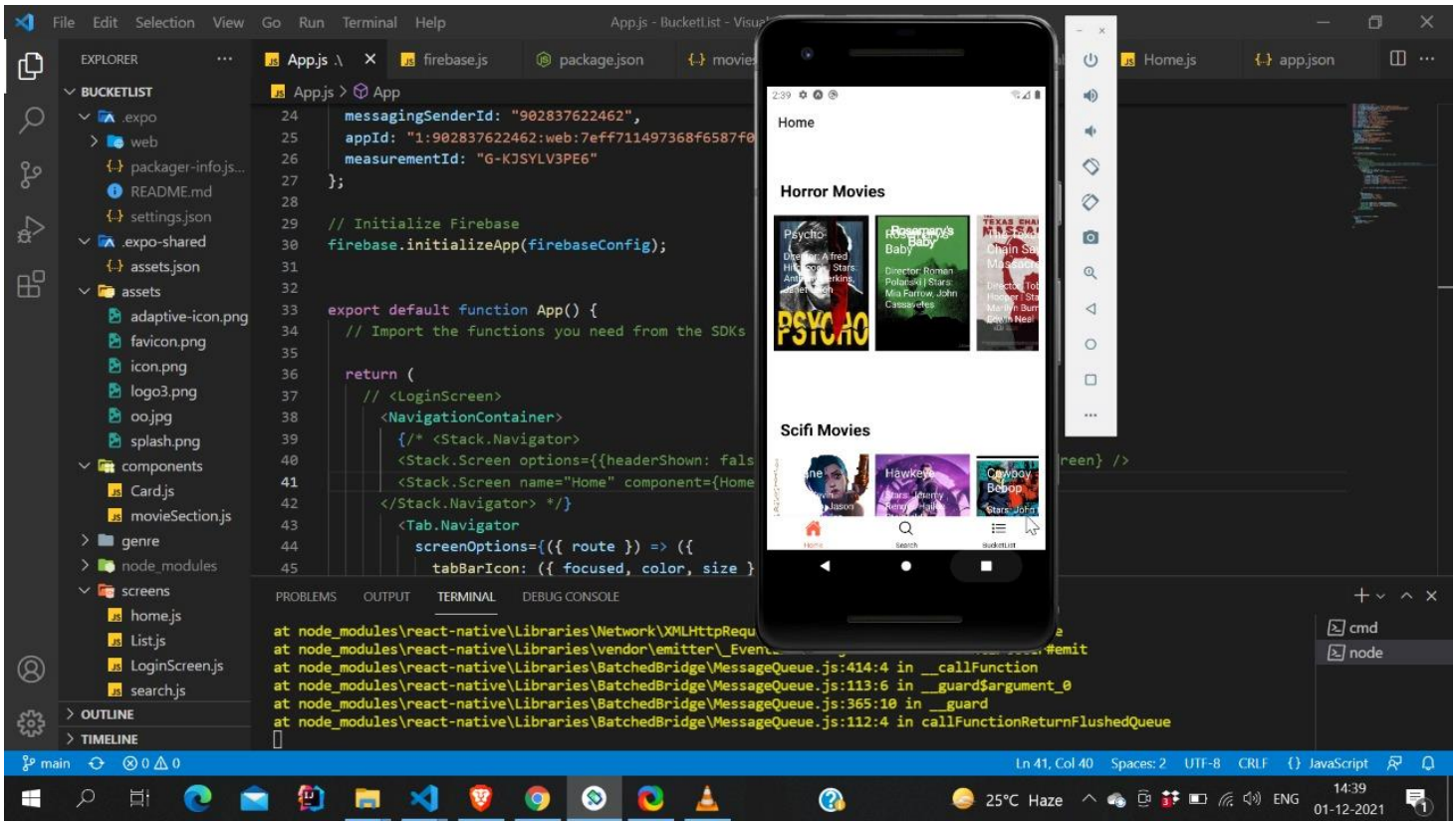

File Edit Selection View Go Run Terminal Help
home.js - BucketList - Visual Studio Code
EXPLORER
BUCKETLIST
App.js
home.js
Card.js
LoginScreen.js
import React, { useEffect } from "react";
import { API_BASE_URL } from "@env";
import {
SafeAreaView,
Text,
View,
StyleSheet,
ActivityIndicator,
} from "react-native";
import MovieSection from "../components/movieSection";
import { ScrollView } from "react-native-gesture-handler";
function HomeScreen({ navigation }) {
const [loading, setLoading] = React.useState(false);
const [error, setError] = React.useState(null);
const [data, setData] = React.useState([]);
const handlePress = ({ title, id }) => {};
useEffect(() => {
setLoading(true);
fetch(`${API_BASE_URL}/movies?p=1&l=6`)
.then((response) => response.json())
.then((data = []) => {
if (!data.length) {
setError("No movies available");
return setLoading(false);
}
setLoading(false);
main
Ln 1, Col 1
Spaces: 2
UTF-8
CRLF
JavaScript
16°C Haze
ENG
22:13
25-11-2021


File Edit Selection View Go Run Terminal Help
search.js - BucketList - Visual Studio Code
EXPLORER
App.js
home.js
Card.js
LoginScreen.js
search.js
BUCKETLIST
screens > search.js > ...
.expo
.expo-shared
assets
adaptive-icon.png
favicon.png
icon.png
logo3.png
oo.jpg
splash.png
components
Card.js
movieSection.js
node_modules
screens
home.js
LoginScreen.js
search.js
tabs
.env
.gitignore
App.js
app.json
babel.config.js
BucketListSynopsis.pdf
OUTLINE
TIMELINE
1 import React from "react";
2 import { StyleSheet, Text, View } from "react-native";
3
4 export default function SearchScreen() {
5 return (
6 <View style={styles.container}>
7 <Text>This is the search screen</Text>
8 </View>
9);
10 }
11
12 const styles = StyleSheet.create({
13 container: {
14 flex: 1,
15 backgroundColor: "#fff",
16 alignItems: "center",
17 justifyContent: "center",
18 },
19 });
main
Ln 1, Col 1
Spaces: 2
UTF-8
CRLF
JavaScript
16°C Haze
ENG
22:13
25-11-2021


File Edit Selection View Go Run Terminal Help
Card.js - BucketList - Visual Studio Code
EXPLORER
BUCKETLIST
.expo
.expo-shared
assets
adaptive-icon.png
favicon.png
icon.png
logo3.png
oo.jpg
splash.png
components
Card.js
movieSection.js
node_modules
screens
home.js
LoginScreen.js
search.js
tabs
.env
.gitignore
App.js
app.json
babel.config.js
BucketListSynopsis.pdf
OUTLINE
TIMELINE
App.js
home.js
Card.js
LoginScreen.js
search.js
components > Card.js > styles > card > borderRadius
1 import React from "react";
2 import {View, StyleSheet} from "react-native";
3
4 const Card = props =>{
5 return(
6 <View style={{...styles.card, ...props.style}}>{props.children}</View>
7
8);
9 };
10
11 const styles= StyleSheet.create({
12 card:{
13 shadowColor:'black',
14 shadowOffset:{width:0,height:2},
15 shadowRadius:6,
16 shadowOpacity:0.26,
17 elevation:8,
18 backgroundColor:'lightslategrey',
19 padding:20,
20 borderRadius:20
21 }
22 });
23
24 export default Card;
main
Ln 20, Col 24 Spaces: 4 UTF-8 CRLF {} JavaScript
16°C Haze
ENG
22:13
25-11-2021


File Edit Selection View Go Run Terminal Help
Card.js - BucketList - Visual Studio Code
EXPLORER
BUCKETLIST
.expo
.expo-shared
assets
adaptive-icon.png
favicon.png
icon.png
logo3.png
oo.jpg
splash.png
components
Card.js
movieSection.js
node_modules
screens
home.js
LoginScreen.js
search.js
tabs
.env
.gitignore
App.js
app.json
babel.config.js
BucketListSynopsis.pdf
OUTLINE
TIMELINE
App.js
home.js
Card.js
movieSection.js
LoginScreen.js
search.js
components > Card.js > styles > card > borderRadius
1 import React from "react";
2 import {View, StyleSheet} from "react-native";
3
4 const Card = props =>{
5 return(
6 <View style={{...styles.card, ...props.style}}>{props.children}</View>
7
8);
9 };
10
11 const styles= StyleSheet.create({
12 card:{
13 shadowColor:'black',
14 shadowOffset:{width:0,height:2},
15 shadowRadius:6,
16 shadowOpacity:0.26,
17 elevation:8,
18 backgroundColor:'lightslategrey',
19 padding:20,
20 borderRadius:20
21 }
22 });
23
24 export default Card;
main
Ln 20, Col 24 Spaces: 4 UTF-8 CRLF {} JavaScript
16°C Haze
ENG
22:13
25-11-2021

Here are some Glimpse of our App



2:54 7.00 KB/S

prattyush online

Search

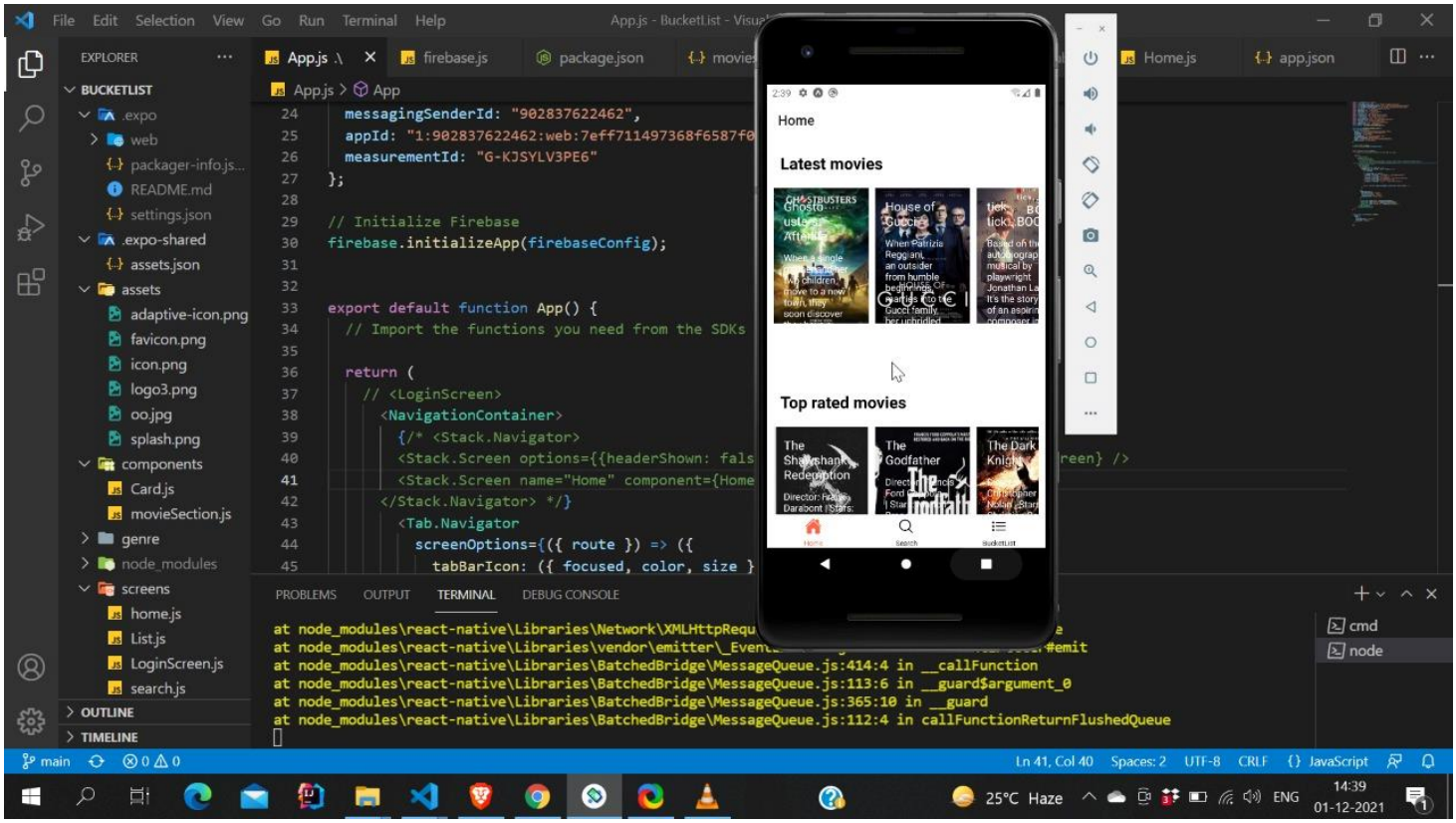
strange

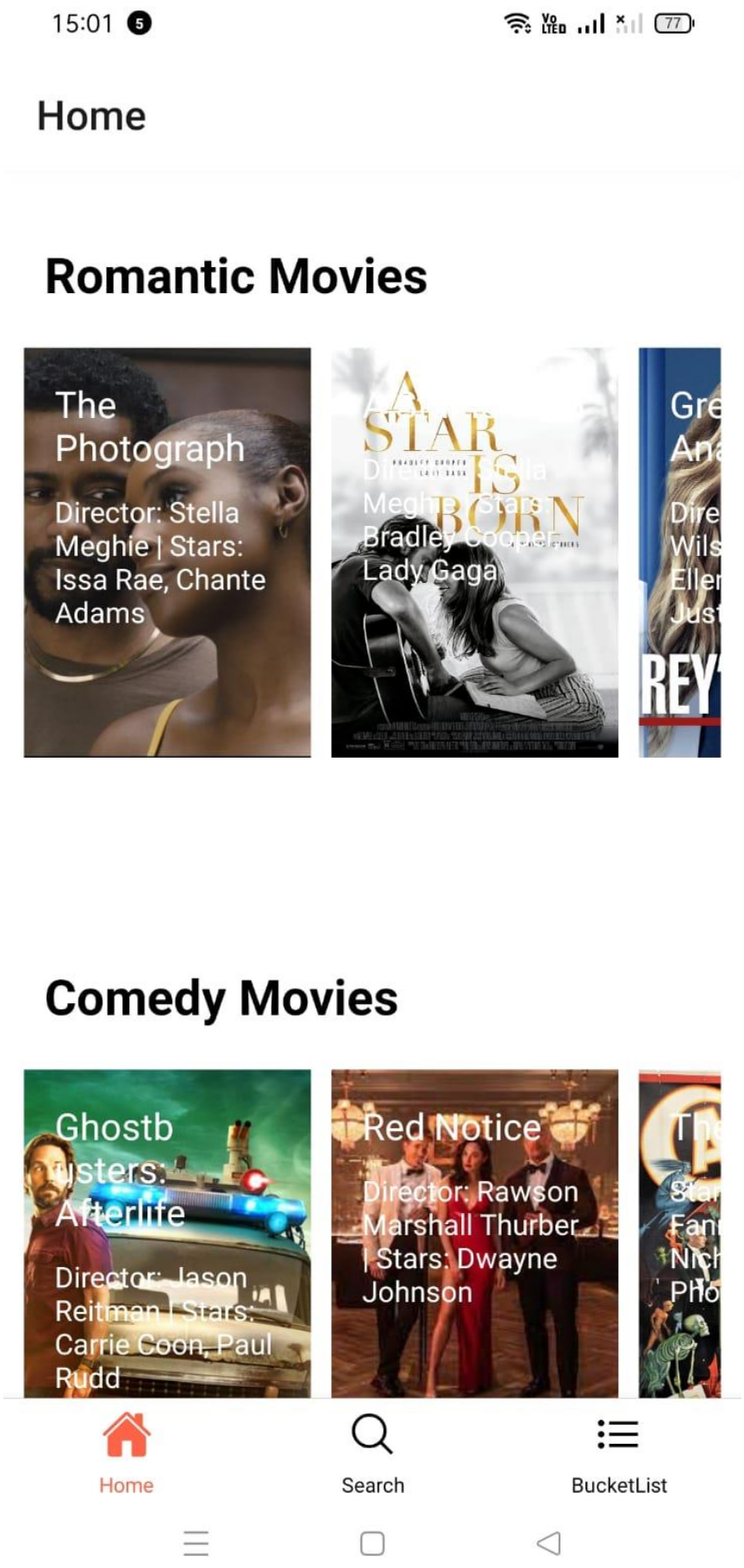
Doctor Strange

Strange Days

Home Search BucketList

React Application




15:01
Home
Romantic Movies
The Photograph
Director: Stella Meghie | Stars: Issa Rae, Chante Adams
A STAR IS BORN
Bradley Cooper
Lady Gaga
Comedy Movies
Ghostb usters: Afterlife
Director: Jason Reitman | Stars: Carrie Coon, Paul Rudd
Red Notice
Director: Rawson Marshall Thurber | Stars: Dwayne Johnson
Home
Search
BucketList

TESTING

Once source code has been generated, software must be tested to uncover as many errors as possible before delivery. It is very important to work the system successfully and achieve high quality of software. Testing include designing a series of test cases that have a high likelihood of finding errors by applying software-testing techniques.

System testing makes logical assumptions that if all the parts of the system are correct, the goal will be successfully achieved. The system should be checked logically. Validations and cross checks should be there. Avoid duplications of record that cause redundancy of data.

In other Words, Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. It is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

6.1 Installation Testing:

There are two types of apps on an Android device i.e., Pre-installed applications and the applications which are installed later by the user.

For both of the above, installation testing is carried out by our teammates. It is ensuring smooth installation of the application without ending up in errors, partial installation etc.

6.2 Unit Testing

It focuses on smallest unit of software design. In this we test an individual unit or groups of inter related units. It is often done by programmer by using sample input and observing its corresponding outputs. In this testing technique we are primarily focuses on:

- Loop methods and function is working fine or not.
- Misunderstood or incorrect Arithmetic precedence
- Incorrect Initialization

User Testing

User testing is the process through which the interface and functions of a website, app, product, or service are tested by real users who perform specific tasks in realistic conditions. The purpose of this process is to evaluate the usability of that website or app and to decide whether the product is ready to be launched for real users.

This app was tested by our team mates and friends who are using different mobile phones (and having different android version) also tested on different emulator to check its performance and it seems to be working fine and users of this app are satisfied with the facilities and performance of the app and like the way how the app is worked.

Performance Testing

In this type of testing we have checked the performances of our application under some peculiar conditions are checked. Those conditions include:

- Low memory in the device.
- The battery in extremely at a low level.
- Poor/Bad network reception.

Performance is basically tested from 2 ends, application end, and the application server end. Our app is also performing well in this phase of testing as well. And we are getting positive feedback from user of our app.

Compability Testing

This application was tested and used on different devices like LG G3, Google Nexus 4. The application worked fine and is stable. The application worked fine in portrait mode and there isn't any problem with compatibility.

On all types of testing (that we have performed above) our performing well on our app i.e. BucketList

CONCLUSION

Proposed BucketList App is a react based application that will allow users to search for their favorites by title. This application takes in a user input and searches the API with the user input and gets a list of movies and web series. Search result screen will contain a list of movies and shows To get the information of the particular content user can click upon the show or movie from the list and then will be taken to the new tab where description and other information related to the show will be available. Users can also add the movie or show to the favorites.

This application has wide range of scope in the upcoming era. It is impossible to arrange the hard space drive of every movie and shows in a personal device so this type of application can reduce the barrier of space in a cost effective, productive way. For youngsters who are interested in watching these contents online can use this application and watch the series.