

PROFESSIONAL TRAINING REPORT

at

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY (Deemed to be University)

Submitted in partial fulfillment of the requirements for the award

of Bachelor of Technology Degree in

Information Technology

By

KAVEYHA G

(Reg. No.39120052)



DEPARTMENT OF INFORMATION TECHNOLOGY

SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI - 600 119

APRIL 2022



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade “A” by NAAC
(Established under Section 3 of UGC Act, 1956)
JEPPIAAR NAGAR, RAJIV GANDHI SALAI
CHENNAI– 600119
www.sathyabama.ac.in

DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KAVEYHA G**
(Reg No. 39120052) and who carried out the project entitled “**MOVIE SEARCHING**
WEB APPLICATION USING REACT-JS” under our supervision from March 2022 to
April 2022.

INTERNAL GUIDE

Dr. A. SIVASANGARI M.E., Ph.D.,

HEAD OF THE DEPARTMENT

Dr. R. SUBHASHINI M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **KAVEYHA G (Reg No:39120052)** hereby declare that the Project Report entitled **“MOVIE SEARCHING WEB APPLICATION USING REACT-JS”** done by me under the guidance of **Dr. A. SIVASANGARI M.E., Ph.D.**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge our sincere thanks to Board of management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey our thanks to **Dr. T. SASIKALA M.E., Ph.D.**, Dean, School of Computing and **Dr. R. SUBHASHINI M.E.,Ph.D.**, Head of the Department, Department of Computer Science and Engineering for providing us the necessary support and details at the right time during the progressive reviews.

I would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. A. SIVASANGARI M.E.,Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express our thanks to all Teaching and Non-teaching staff members of the Department of **INFORMATION TECHNOLOGY** who were helpful in many ways for the completion of the project.

TRAINING CERTIFICATE



ABSTRACT

React Js is a JavaScript-based UI development library. Facebook and an open-source developer community run it. Although React is a library rather than a language, it is widely used in web development. The library first appeared in May 2013 and is now one of the most commonly used frontend libraries for web development.

ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based frontend library which is responsible only for the view layer of the application. It was initially developed and maintained by Facebook and later used in its products like WhatsApp & Instagram. The main objective of ReactJS is to develop User Interfaces (UI) that improves the speed of the apps. It uses virtual DOM (JavaScript object), which improves the performance of the app. The JavaScript virtual DOM is faster than the regular DOM. We can use ReactJS on the client and server-side as well as with other frameworks. It uses component and data patterns that improve readability and helps to maintain larger apps.

React is a library for building composable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time. Lots of people use React as the V in MVC. React abstracts away the DOM from you, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding.

TABLE CONTENT

CHAPTER NO:	TITLE	PAGE NO:
	ABSTRACT	i
	LIST OF FIGURES	ii
1.	INTRODUCTION 1.1 REACT JS 1.2 OUTLINE OF THE PROJECT 1.3 ORGANIZATION OF THE REPORT	 1 1 2
2.	AIM 2.1 AIM 2.2 REACT FEATURES 2.3 LITERATURE REVIEW 2.4 SCOPE	 3 3 3 4
3.	ALGORITHMS AND METHODS 3.1 GENERAL 3.2 ALGORITHM 3.3 PROJECT DESCRIPTION	 5 5 6
4.	SYSTEM IMPLEMENTATION 4.1 REQUIREMENTS 4.2 ROLE OF REACT JS 4.3 RESULTS	 8 8 9-11

CHAPTER NO	TITLE	PAGE NO
5.	CONCLUSION AND FUTURE WORK	
	5.1 CONCLUSION	12
	5.2 FUTURE WORK	12
	REFERENCES	38
	APPENDIX	
	A. STEP BY STEP PROCEDURE	13-34
	B. SOURCE CODE	35-17

FIGURE NO.	LIST OF FIGURE	PAGE NO.
1.1	Download visual studio code	13
1.2	Visual Studio code is downloaded and installed	14
1.3	React Installation	15
1.4	Start react app using npm start	16
1.5	React app started in browser	17
1.6	Getting OMDBI API key	18
1.7	Installing and importing styled components and axios	19
1.8	Setting up Icons and Search box	20
1.9	Setting up Movie Component	21
1.10	Movie Component	22
1.11	Setup API key	23
1.12	Setting up movie info component	24-25
1.13	Integrating Movie component with API call	26
1.14	Movie Info Component	27
1.15	Loading Display	28
1.16	Close button	29
1.17	Final output	30

CHAPTER 1

INTRODUCTION

1.1 REACT JS

React uses a declarative paradigm that makes it easier to reason about your application and aims to be both efficient and flexible. It designs simple views for each state in your application, and React will efficiently update and render just the right component when your data changes. The declarative view makes your code more predictable and easier to debug.

A React application is made of multiple components, each responsible for rendering a small, reusable piece of HTML. Components can be nested within other components to allow complex applications to be built out of simple building blocks. A component may also maintain an internal state – for example, a TabList component may store a variable corresponding to the currently open tab.

ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based front end library responsible only for the view layer of the application. It was created by **Jordan Walke** , who was a software engineer at **Facebook**. It was initially developed and maintained by Facebook and was later used in its products like **WhatsApp** & **Instagram**. Facebook developed ReactJS in **2011** in its newsfeed section, but it was released to the public in the month of **May 2013**.

1.2 OUTLINE OF THE PROJECT

- To build a Movie Searching App using ReactJS
- Developing a well defined website accessing system.
- React Movie Search App will be having the feature to Search the Movies from OMDb APIs and will provide info like Release Date, Genre, Director ,Movie description etc.

1.3 ORGANIZATION OF THE REPORT

Chapter 2 deals with Aim, React Features, Literature Review and Scope. Chapter 3 deals with General, Algorithm, Project description. Chapter 4 deals with Requirement, Role of React JS and Results and Discussion. Chapter 5 deals with Conclusion and Future work.

CHAPTER 2

AIM & SCOPE

2.1 AIM

To build a Movie Searching App using ReactJS. This React Movie Search App will be having the feature to Search the Movies from OMDb APIs and will provide info like Release Date, Genre, Director, Movie description etc.

2.2 REACT FEATURES

- **JSX** – JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.
- **Components** – React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.
- **Unidirectional data flow and Flux** – React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keeping your data unidirectional.
- **License** – React is licensed under the Facebook Inc. Documentation is licensed under CC BY 4.0.

2.3 LITERATURE REVIEW

React JS is a JavaScript library that makes building user interfaces much better and life much better for developers. It has already become mainstream and is used by a number of big names including Facebook, Netflix, AirBNB, DropBox, IMDb, PayPal, Tesla Motors, Walmart and many others.

- Uses virtual DOM which is a JavaScript object. This will improve apps performance, since JavaScript virtual DOM is faster than the regular DOM.
- Can be used on client and server side as well as with other frameworks.
- Component and data patterns improve readability, which helps to maintain larger apps.

2.4 SCOPE

React development has emerged as the future of web creation with its extra versatility and ease. **React saves you time and money on development because it's component-based.** You can break down an interface into reusable components that allow you to build dynamic user interfaces.

React JS is very easy to learn and more focused than some other JavaScript frameworks. Many businesses are shifting or adopting React library because of the simplicity it provides and ease of use. Ease of Learning React is the best advantage as compared to other popular front-end frameworks like Angular and Vue.

When technology is difficult to learn, it gets difficult to complete a project on it and make it successful. So, enterprises prefer to deploy that technology which is easy to learn for the developing team and finish the project in scheduled time.

CHAPTER 3

ALGORITHMS AND METHODS

3.1 GENERAL

ReactJS is a declarative, efficient JavaScript framework to build interactive user interfaces (UI) with greater ease and flexibility. It provides many advantages in designing simple views for different states in an application and greater ease in updating and rendering the right components efficiently just when the data changes.

3.2 ALGORITHM

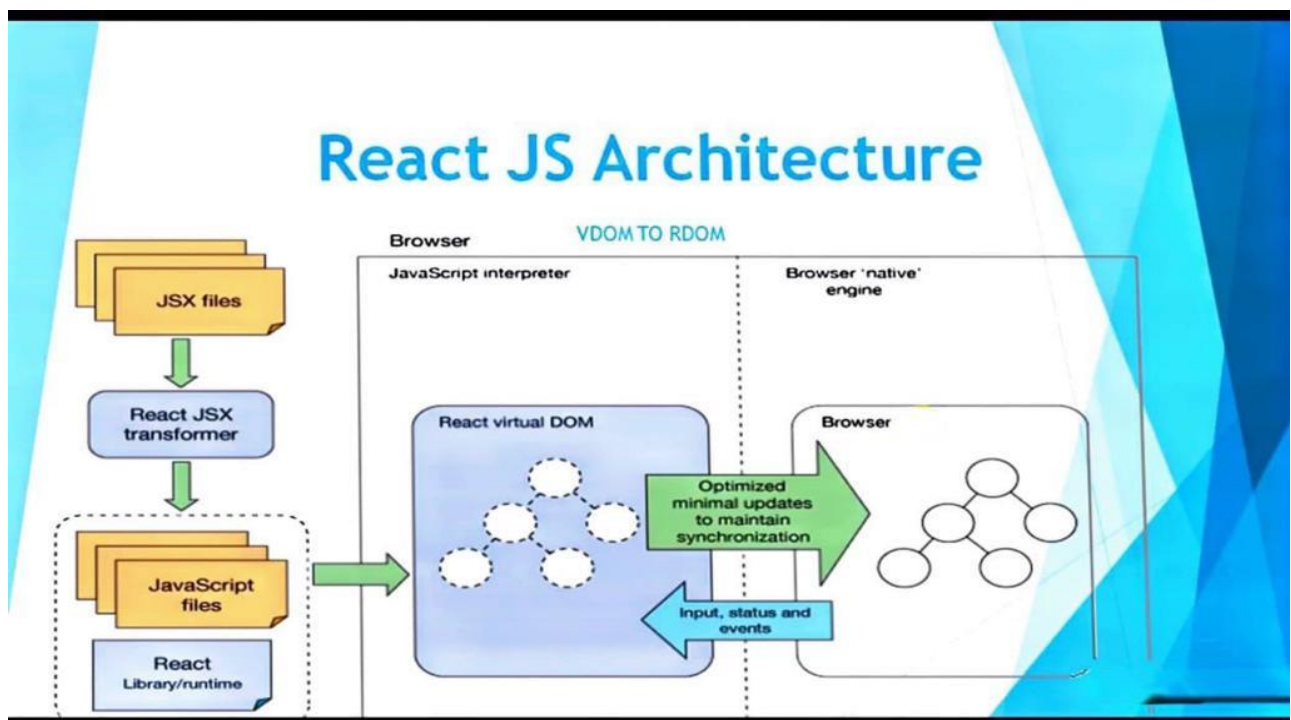
1. Download visual studio code
2. Visual Studio code is downloaded and installed
3. React Installation
4. Start react app using npm start
5. React app started in browser
6. Getting OMDBI API key
7. Installing and importing styled components and axios
8. Setting up Icons and Search box
9. Setting up Movie Component
10. Movie Component
11. Setup API key
12. Setting up movie info component
13. Integrating Movie component with API call
14. Movie Info Component
15. Loading Display
16. Close button
17. The final Output of the Site

3.3 PROJECT DESCRIPTION

React's popularity today has eclipsed that of all other front-end development frameworks. Here is why:

- Easy creation of dynamic applications: React makes it easier to create dynamic web applications because it requires less coding and offers more functionality, as opposed to JavaScript, where coding often gets complex very quickly.
- Improved performance: React uses Virtual DOM, thereby creating web applications faster. Virtual DOM compares the components' previous states and updates only the items in the Real DOM that were changed, instead of updating all of the components again, as conventional web applications do.
- Reusable components: Components are the building blocks of any React application, and a single app usually consists of multiple components. These components have their logic and controls, and they can be reused throughout the application, which in turn dramatically reduces the application's development time.
- Unidirectional data flow: React follows a unidirectional data flow. This means that when designing a React app, developers often nest child components within parent components. Since the data flows in a single direction, it becomes easier to debug errors and know where a problem occurs in an application at the moment in question.
- Small learning curve: React is easy to learn, as it mostly combines basic HTML and JavaScript concepts with some beneficial additions. Still, as is the case with other tools and frameworks, you have to spend some time to get a proper understanding of React's library.
- It can be used for the development of both web and mobile apps: We already know that React is used for the development of web applications, but that's not all it can do. There is a framework called React Native, derived from React itself, that is hugely popular and is used for creating beautiful mobile applications. So, in reality, React can be used for making both web and mobile applications.
- Dedicated tools for easy debugging: Facebook has released a Chrome extension that can be used to debug React applications. This makes the process of debugging React web applications faster and easier.

- The above reasons more than justify the popularity of the React library and why it is being adopted by a large number of organizations and businesses. Now let's familiarize ourselves with React's features



3.3.1 REACT JS ARCHITECTURE

In a Model View Controller (MVC) architecture, React is the 'View' responsible for how the app looks and feels.

MVC is an architectural pattern that splits the application layer into Model, View, and Controller. The model relates to all data-related logic; the view is used for the UI logic of the application, and the controller is an interface between the Model and View.

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 REQUIREMENTS

Tools used:

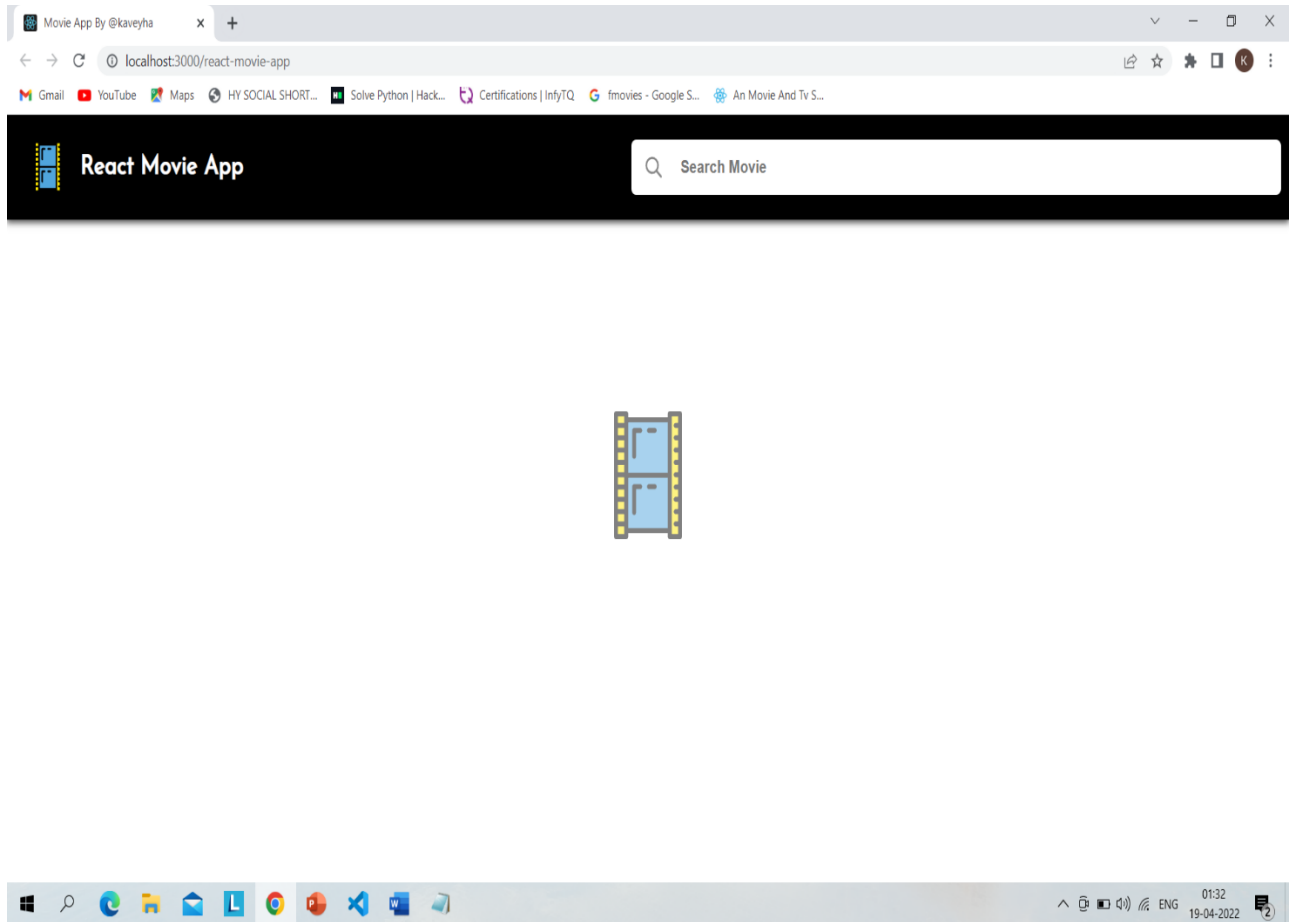
- Visual studio code
- OMDBI api
- Styled Component
- Axios

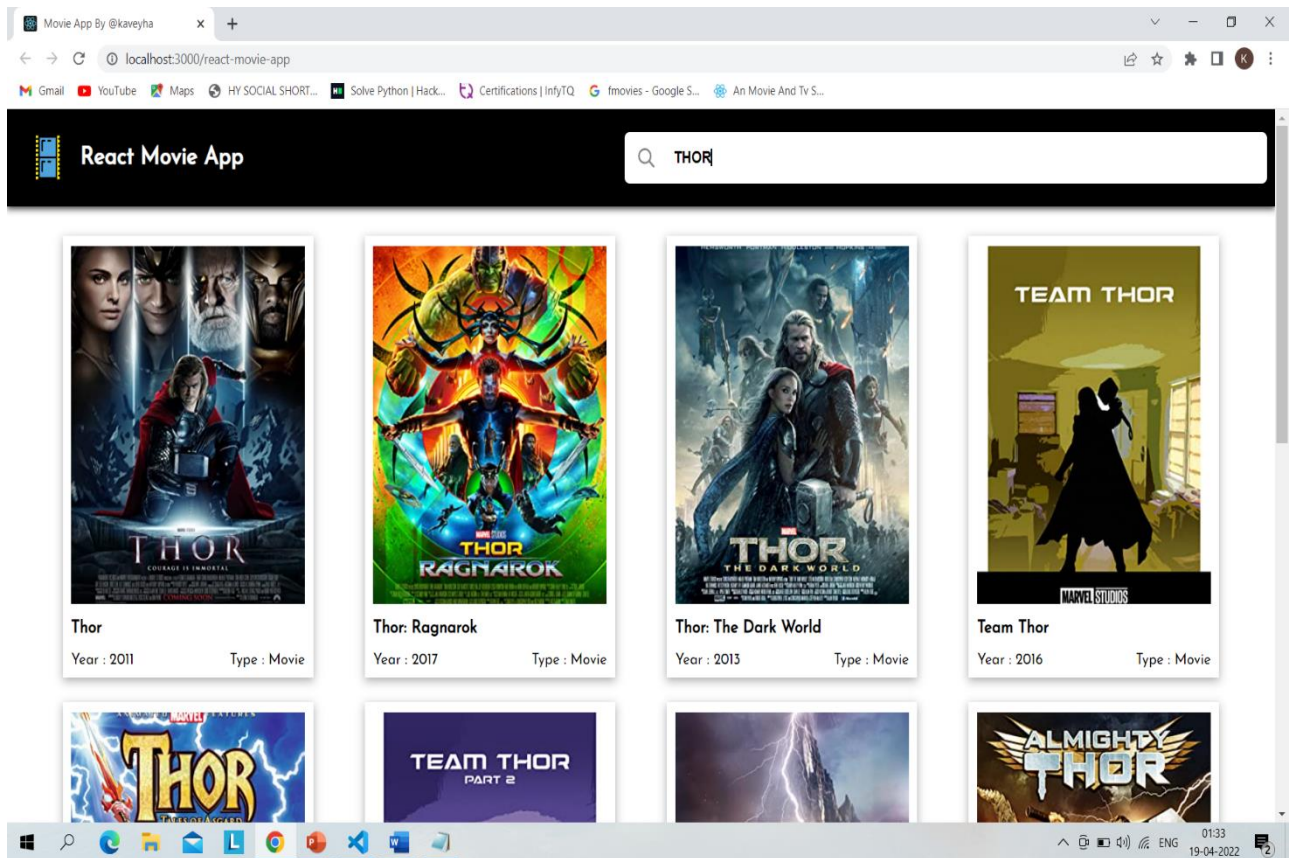
The objective is reached by using these above tools in an efficient manner ,by a step by step component based development. A responsive UI using FLEX concept is developed.

4.2 ROLE OF REACT JS

React. js is an open-source JavaScript library that is used for building user interfaces specifically for single-page applications. It's used for handling the view layer for web and mobile apps. React also allows us to create reusable UI components.

4.3 RESULTS





Movie App By @kaveyha

localhost:3000/react-movie-app

GmailYouTubeMapsHY SOCIAL SHORT...Solve Python | Hack...Certifications | InfyTQfmovies - Google S...An Movie And Tv S...

React Movie App

THOR



Movie: Thor

IMDB Rating: 7.0

Year: 2011

Language: English

Rated: PG-13

Released: 06 May 2011

Runtime: 115 Min

Genre: Action, Adventure, Fantasy

Director: Kenneth Branagh

Actors: Chris Hemsworth, Anthony Hopkins, Natalie Portman

Plot: The Powerful But Arrogant God Thor Is Cast Out Of Asgard To Live Amongst Humans In Midgard (Earth), Where He Soon Becomes One Of Their Finest Defenders.



01:34

19-04-2022

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

- We built a component that lets a user search for a movie by title, then stores the movie title in that component's state.
- Then, we stored the response in the Movies state and built a search component that displays the response data we got from the API. We then show the results to the user.

5.2 FUTURE WORK

React JS is going to be a game-changer even in the future. Developers proficient in JavaScript will be able to take advantage of React to build innovative products. Businesses will continue to hire JavaScript developers, but React programmers will gain the upper hand. You can expect bright days ahead and hope to cash in on your React skills without any exception.

APPENDICES

A. STEP BY STEP PROCEDURE :

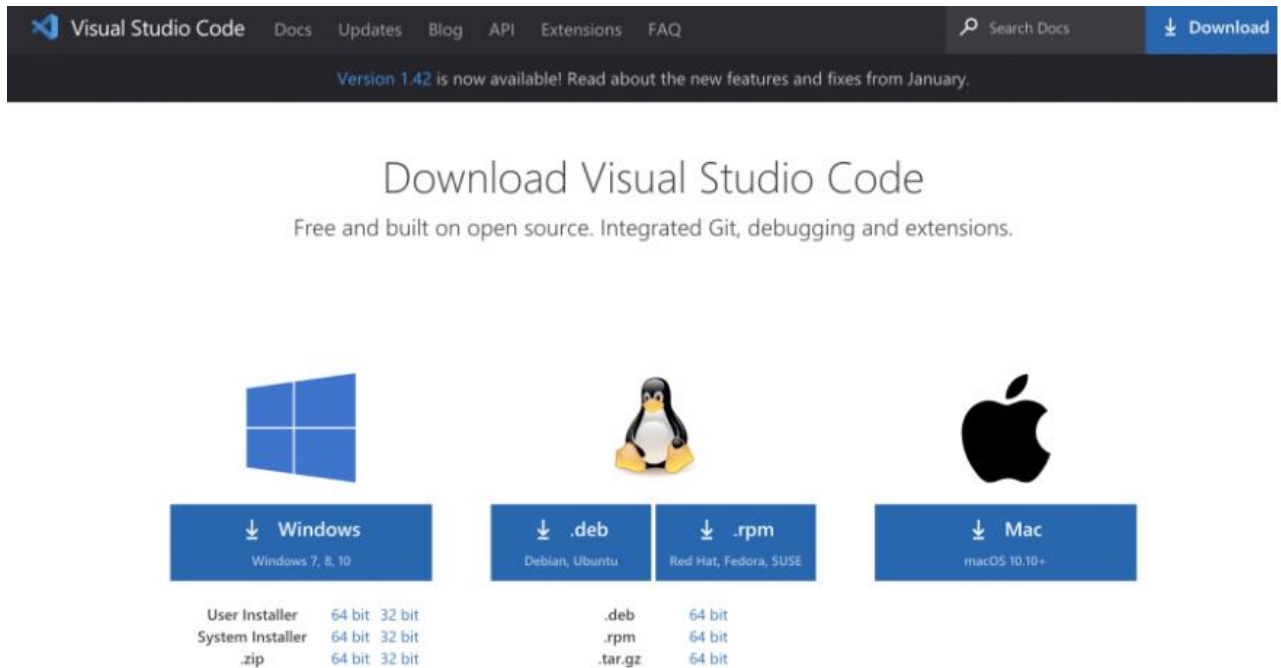


Fig 1.1 Download visual studio code

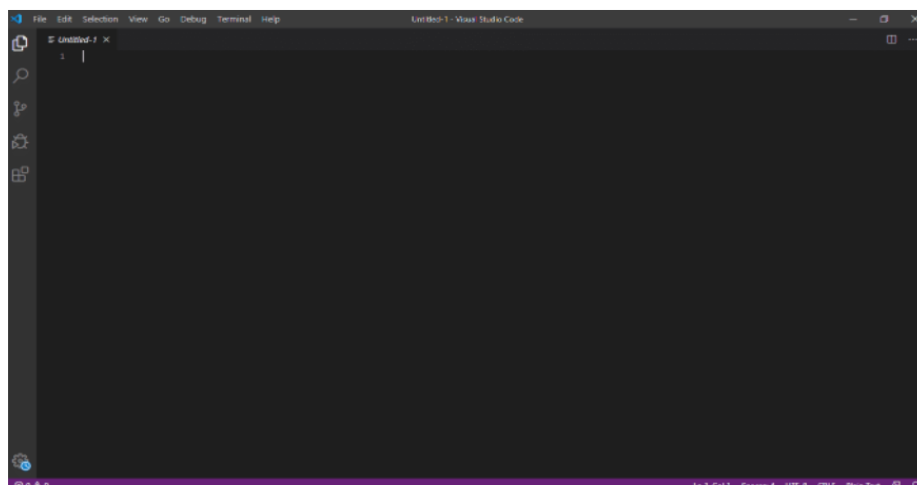
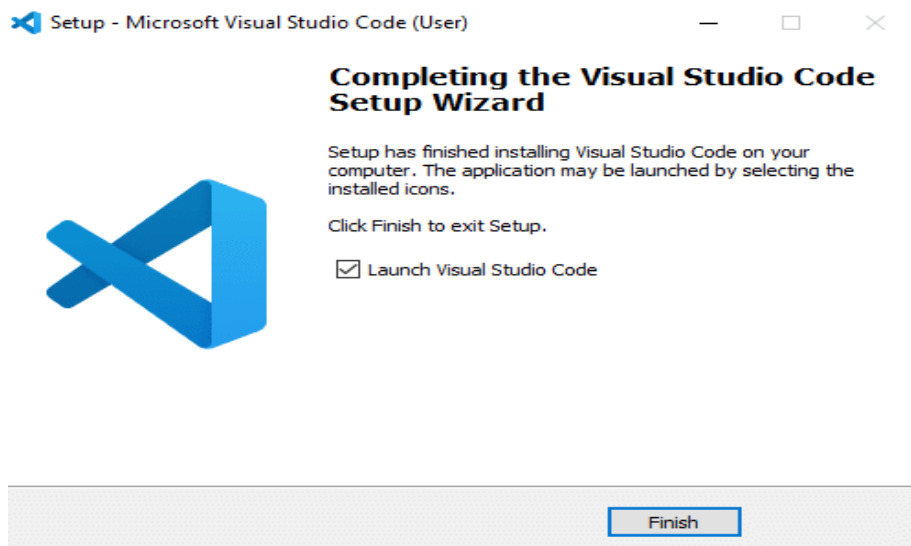
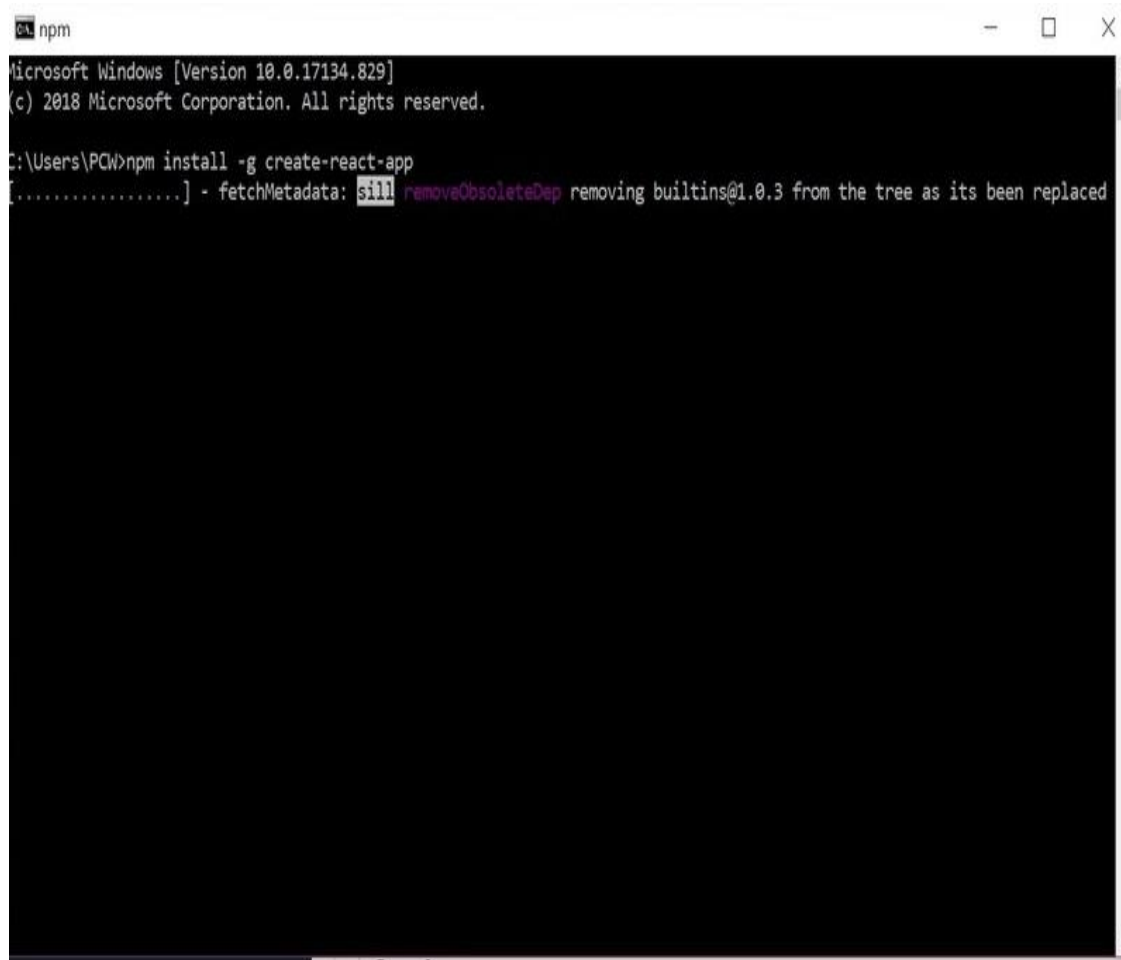


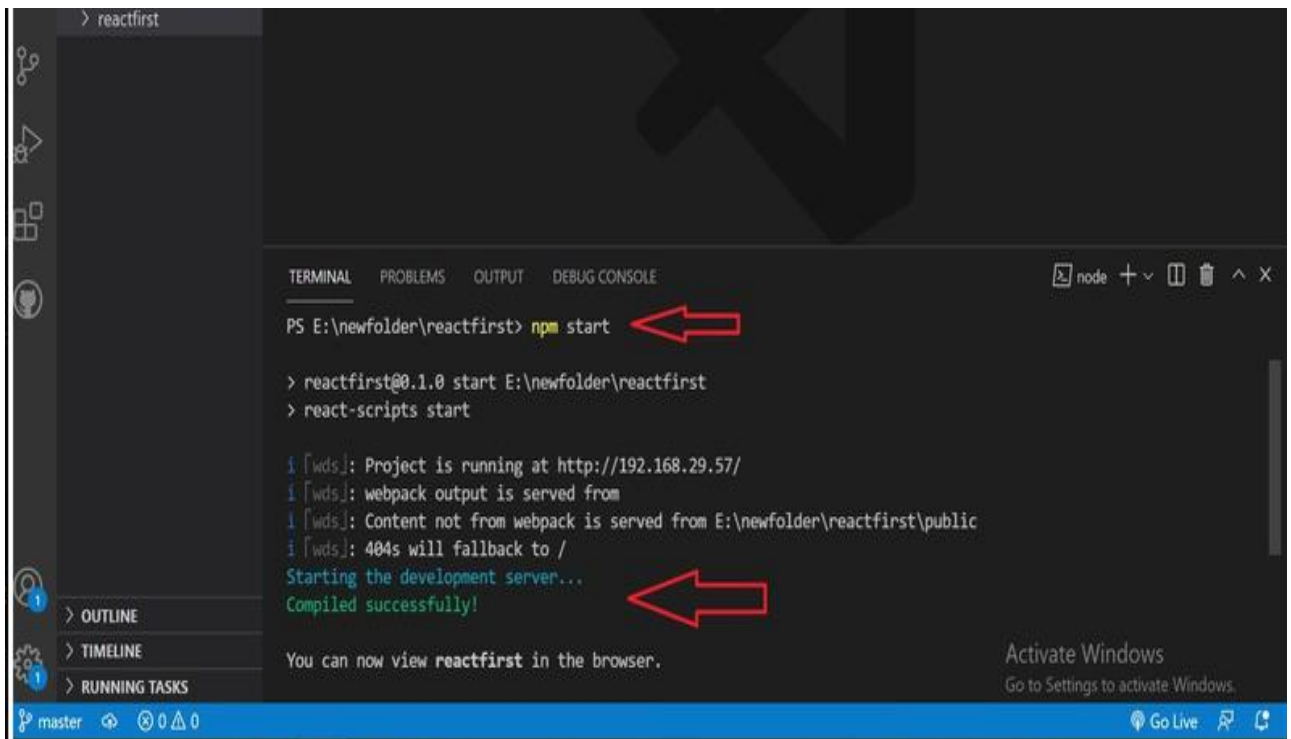
Fig1.2 Visual Studio code is downloaded and installed



```
npm
Microsoft Windows [Version 10.0.17134.829]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\PCW>npm install -g create-react-app
[.....] - fetchMetadata: sill removeObsoleteDep removing builtins@1.0.3 from the tree as its been replaced
```

Fig1.3 React Installation



The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal title is 'reactfirst'. The command prompt shows 'PS E:\newfolder\reactfirst> npm start', with a red arrow pointing to 'npm start'. The output of the command is as follows:

```
> reactfirst@0.1.0 start E:\newfolder\reactfirst
> react-scripts start

i [wds]: Project is running at http://192.168.29.57/
i [wds]: webpack output is served from
i [wds]: Content not from webpack is served from E:\newfolder\reactfirst\public
i [wds]: 404s will fallback to /
Starting the development server...
Compiled successfully!
```

Two red arrows point to 'npm start' and 'Compiled successfully!'. Below the output, it says 'You can now view reactfirst in the browser.' The status bar at the bottom shows 'master' and 'Go Live'.

Fig1.4 Start react app using npm start



Fig 1.5 React app started in browser

OMDb API

Usage

Parameters

Examples

Change Log

API Key

Become a Patron

Donate

Contact

OMDb API

The Open Movie Database

The OMDb API is a RESTful web service to obtain movie information, all content and images on the site are contributed and maintained by our users.

If you find this service useful, please consider making a [one-time donation](#) or [become a patron](#).



Poster API

The Poster API is only available to patrons.

Currently over 280,000 posters, updated daily with resolutions up to 2000x3000.

Attention Users

04/08/19 - Added support for eight digit IMDb IDs.

01/20/19 - Supressed adult content from search results.

01/20/19 - Added Swagger files ([YAML](#), [JSON](#)) to expose current API abilities and upcoming REST functions.

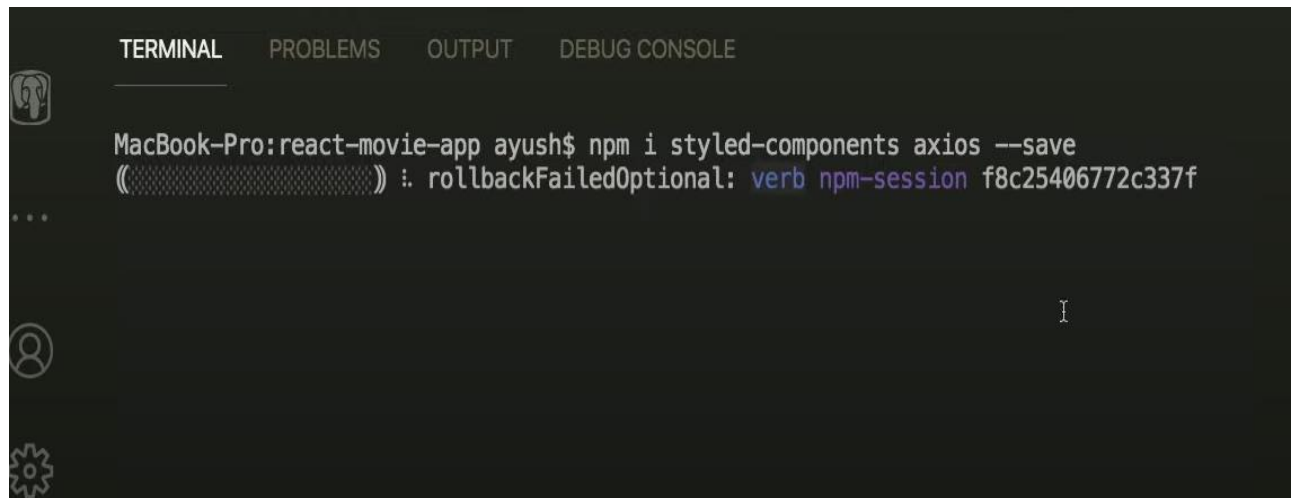

[Become a Patron](#)

Sponsors

Emby, Trakt, FileBot, Reelgood, Xirvik Servers, Yidio, mi.tv, Couchpop, What's on Netflix, Edu Reviewer, Flixboss, StreamingMoviesRight, Streamcentrum.se, Scripts on Screen, Writers Per Hour, Classic Movies, Website Design Companies, Write my paper, Web app development agency

Fig 1.6 Getting OMDb API key

1



A screenshot of a terminal window within a code editor. The terminal has tabs for 'TERMINAL', 'PROBLEMS', 'OUTPUT', and 'DEBUG CONSOLE'. The 'TERMINAL' tab is active. The prompt is 'MacBook-Pro:react-movie-app ayush\$'. The command entered is 'npm i styled-components axios --save'. The output shows a progress bar and the message ':. rollbackFailedOptional: verb npm-session f8c25406772c337f'. On the left side of the terminal, there are icons for Explorer, Search, Source Control, and Run and Debug.

```
MacBook-Pro:react-movie-app ayush$ npm i styled-components axios --save
((.....)) :. rollbackFailedOptional: verb npm-session f8c25406772c337f
```

```
JS App.js > App
import React , {useState } from 'react';
import Axios from 'axios';
import styled from 'styled-components'
```

Fig 1.7 Installing and importing styled components and axios

```

const Container: StyledComponent<"div", any, {}, never>
<Container>
  <Header>
    <AppName>
      <MovieImage src="movie-icon.svg"/>
      React Movie App
    </AppName>
    <SearchBox>
      <SearchIcon src="/search-icon.svg"/>
      <SearchInput
        placeholder="Search Movie"
        value={searchQuery}
        onChange={onTextChange}
      />
    </SearchBox>
  </Header>
</Container>

```

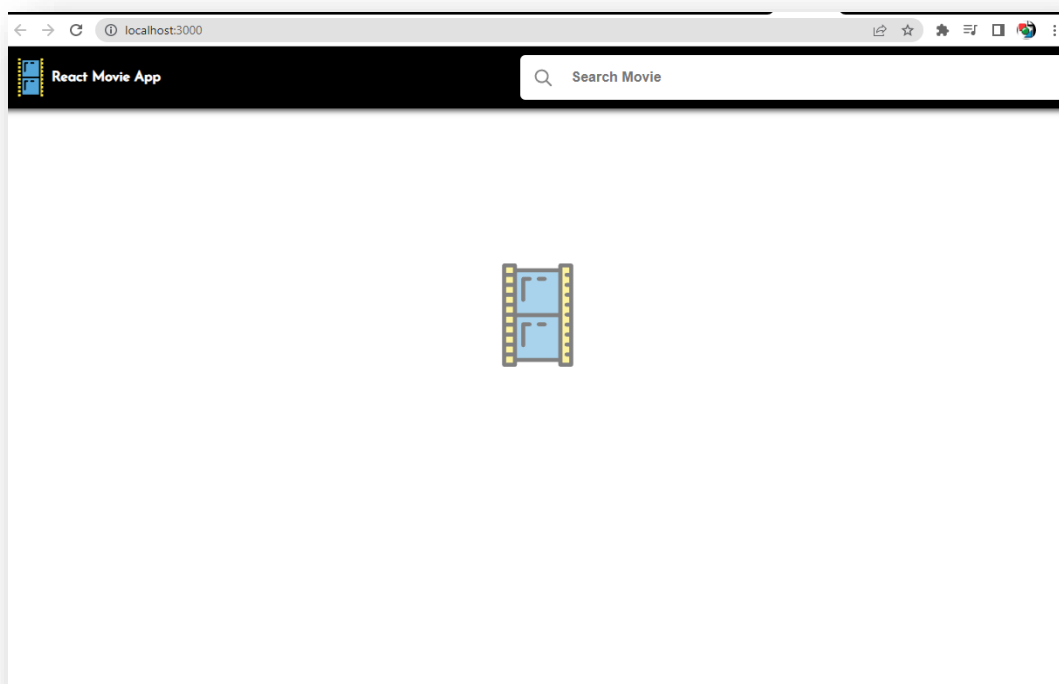


Fig 1.8 Setting up Icons and Search box

```
import styled from "styled-components";
const MovieContainer = styled.div`
  display: flex;
  flex-direction: column;
  padding: 10px;
  width: 280px;
  box-shadow: 0 3px 10px 0 #aaa;
  cursor: pointer;
`;
const CoverImage = styled.img`
  object-fit: cover;
  height: 362px;
`;
const MovieName = styled.span`
  font-size: 18px;
  font-weight: 600;
  color: black;
  margin: 15px 0;
  white-space: nowrap;
  overflow: hidden;
  text-overflow: ellipsis;
`;
const InfoColumn = styled.div`
  display: flex;
  flex-direction: row;
  justify-content: space-between;
`;
const MovieInfo = styled.span`
  font-size: 16px;
  font-weight: 500;
  color: black;
  white-space: nowrap;
  overflow: hidden;
  text-transform: capitalize;
  text-overflow: ellipsis;
`;
```

```
const MovieComponent = (props) => {
  const {Title, Year, imdbID, Type, Poster} = props.movie;
  return (
    <MovieContainer
      onClick={() =>
        props.onMovieSelect(imdbID)}>
      <CoverImage src={Poster}/>
      <MovieName>{Title}</MovieName>
      <InfoColumn>
        <MovieInfo>Year : {Year}</MovieInfo>
        <MovieInfo>Genre : {Type}</MovieInfo>
      </InfoColumn>
    </MovieContainer>
  )
}
```

Fig 1.9 Setting up Movie Component

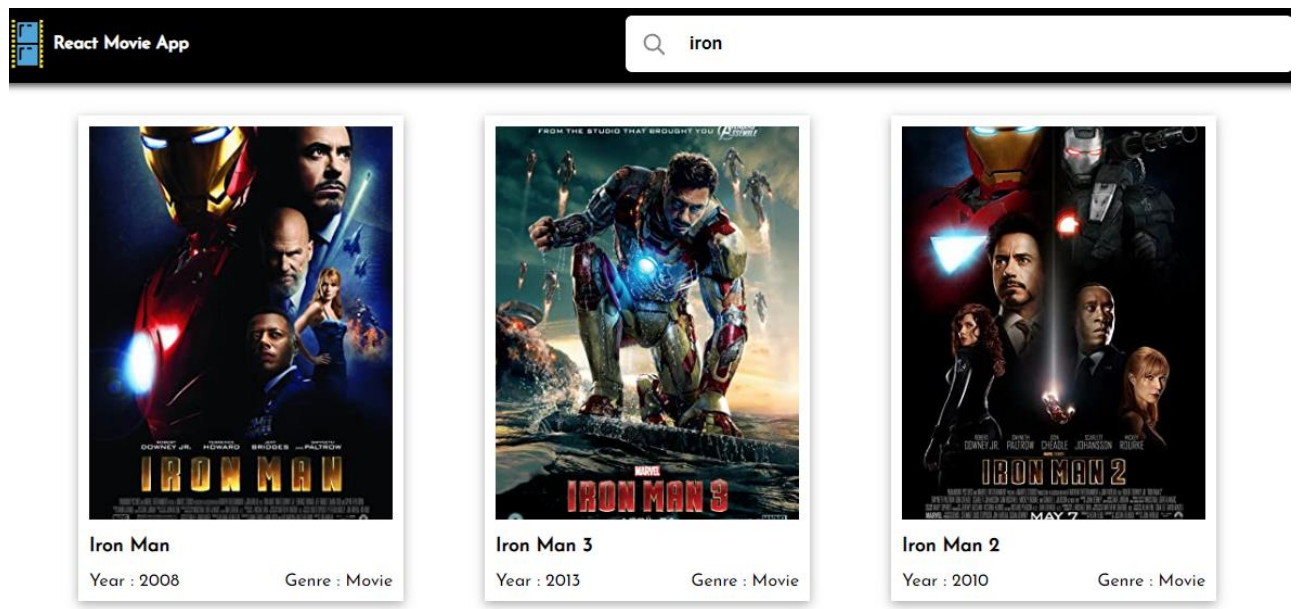


Fig 1.10 Movie Component

```
export const API_KEY="a9118a3a";
```

```
function App() {  
  const [searchQuery, updateSearchQuery] = useState();  
  const [timeoutId, updateTimeoutId]= useState();  
  const [movieList, updateMovieList] = useState([]);  
  const [selectedMovie, onMovieSelect] = useState();  
  
  const fetchData = async (searchString) => {  
    const response=await Axios.get(  
      `https://www.omdbapi.com/?s=${searchString}&apikey=${API_KEY}`,  
    );  
  
    updateMovieList(response.data.Search);  
  };  
};
```

Fig 1.11 Setup API key

```

import React, { useEffect, useState } from "react";
import Axios from "axios";
import { API_KEY } from "../App";
import styled from "styled-components";

const Container = styled.div`
  display: flex;
  flex-direction: row;
  padding: 20px 30px;
  justify-content: center;
  border-bottom: 1px solid lightgray;
`;

const CoverImage = styled.img`
  object-fit: cover;
  height: 350px;
`;

const InfoColumn = styled.div`
  display: flex;
  flex-direction: column;
  margin: 20px;
`;

const MovieName = styled.span`
  font-size: 22px;
  font-weight: 600;
  color: black;
  margin: 15px 0;
  white-space: nowrap;
  overflow: hidden;
  text-transform: capitalize;
  text-overflow: ellipsis;
  & span {
    opacity: 0.8;
  }
`;

```

Fig 1.12 Setting up movie info component

```
const MovieInfo = styled.span`
  font-size: 16px;
  font-weight: 500;
  color: black;
  overflow: hidden;
  margin: 4px 0;
  text-transform: capitalize;
  text-overflow: ellipsis;
  & span {
    opacity: 0.5;
  }
`;

const Close = styled.span`
  font-size: 16px;
  font-weight: 600;
  color: black;
  background: lightgray;
  height: fit-content;
  padding: 8px;
  border-radius: 50%;
  cursor: pointer;
  opacity: 0.8;
`;
```

Fig 1.12 Setting up movie info component

```

const MovieInfoComponent = (props) => {
  const [movieInfo, setMovieInfo] = useState();
  const { selectedMovie } = props;

  useEffect(() => {
    Axios.get(
      `https://www.omdbapi.com/?i=${selectedMovie}&apikey=${API_KEY}`,
    ).then((response) => setMovieInfo(response.data));
  }, [selectedMovie]);

  return (
    <Container>
      {movieInfo ? (
        <>
          <CoverImage src={movieInfo?.Poster} alt={movieInfo?.Title} />
          <InfoColumn>
            <MovieName>
              {movieInfo?.Type}: <span>{movieInfo?.Title}</span>
            </MovieName>
            <MovieInfo>
              IMDB Rating: <span>{movieInfo?.imdbRating}</span>
            </MovieInfo>
            <MovieInfo>
              Year: <span>{movieInfo?.Year}</span>
            </MovieInfo>
            <MovieInfo>
              Language: <span>{movieInfo?.Language}</span>
            </MovieInfo>
            <MovieInfo>
              Rated: <span>{movieInfo?.Rated}</span>
            </MovieInfo>
            <MovieInfo>
              Released: <span>{movieInfo?.Released}</span>
            </MovieInfo>
            <MovieInfo>
              Runtime: <span>{movieInfo?.Runtime}</span>
            </MovieInfo>
            <MovieInfo>
              Genre: <span>{movieInfo?.Genre}</span>
            </MovieInfo>
            <MovieInfo>
              Director: <span>{movieInfo?.Director}</span>
            </MovieInfo>
            <MovieInfo>
              Actors: <span>{movieInfo?.Actors}</span>
            </MovieInfo>
            <MovieInfo>
              Plot: <span>{movieInfo?.Plot}</span>
            </MovieInfo>
          </>
        </>
      ) : null}
    </Container>
  );
};

```

Fig 1.13 Integrating Movie component with API call



Movie: Iron Man

IMDB Rating: 7.9

Year: 2008

Language: English, Persian, Urdu, Arabic, Kurdish, Hindi, Hungarian

Rated: PG-13

Released: 02 May 2008

Runtime: 126 Min

Genre: Action, Adventure, Sci-Fi

Director: Jon Favreau

Actors: Robert Downey Jr., Gwyneth Paltrow, Terrence Howard

Plot: After Being Held Captive In An Afghan Cave, Billionaire Engineer Tony Stark Creates A Unique Weaponized Suit Of Armor To Fight Evil.

Fig 1.14 Movie Info Component

```
</Container>
): (
  "Loading..."
)}
</Container>
```

Loading...



Genre : Movie



Iron Man 3

Year : 2013

Genre : Movie



Iron Man 2

Year : 2010

Gen

Fig 1.15 Loading Display

```
const Close = styled.span`
  font-size: 16px;
  font-weight: 600;
  color: black;
  background: lightgray;
  height: fit-content;
  padding: 8px;
  border-radius: 50%;
  cursor: pointer;
  opacity: 0.8;
;`
```

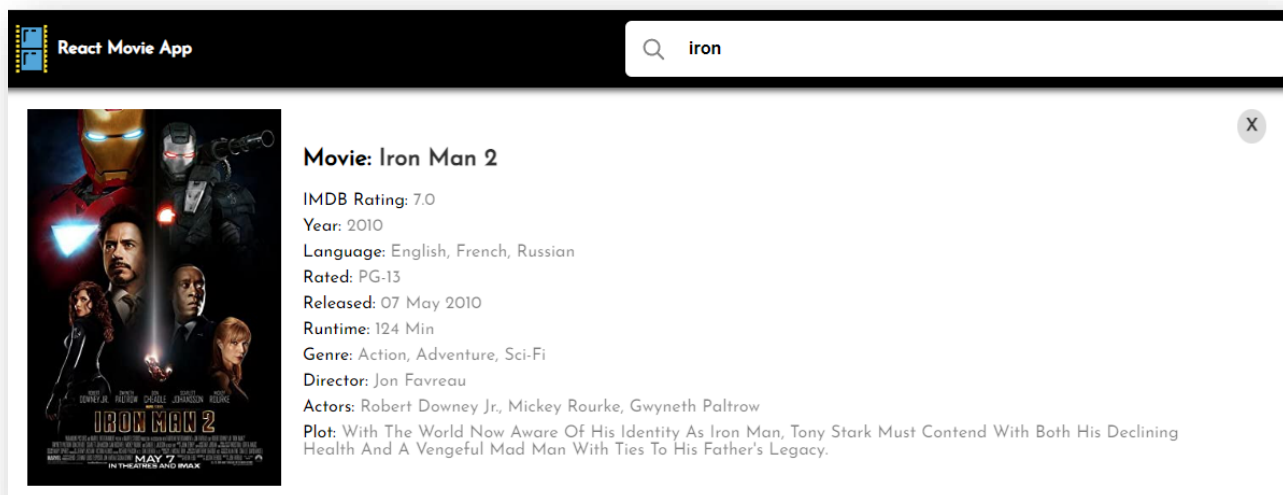


Fig 1.16 Close button

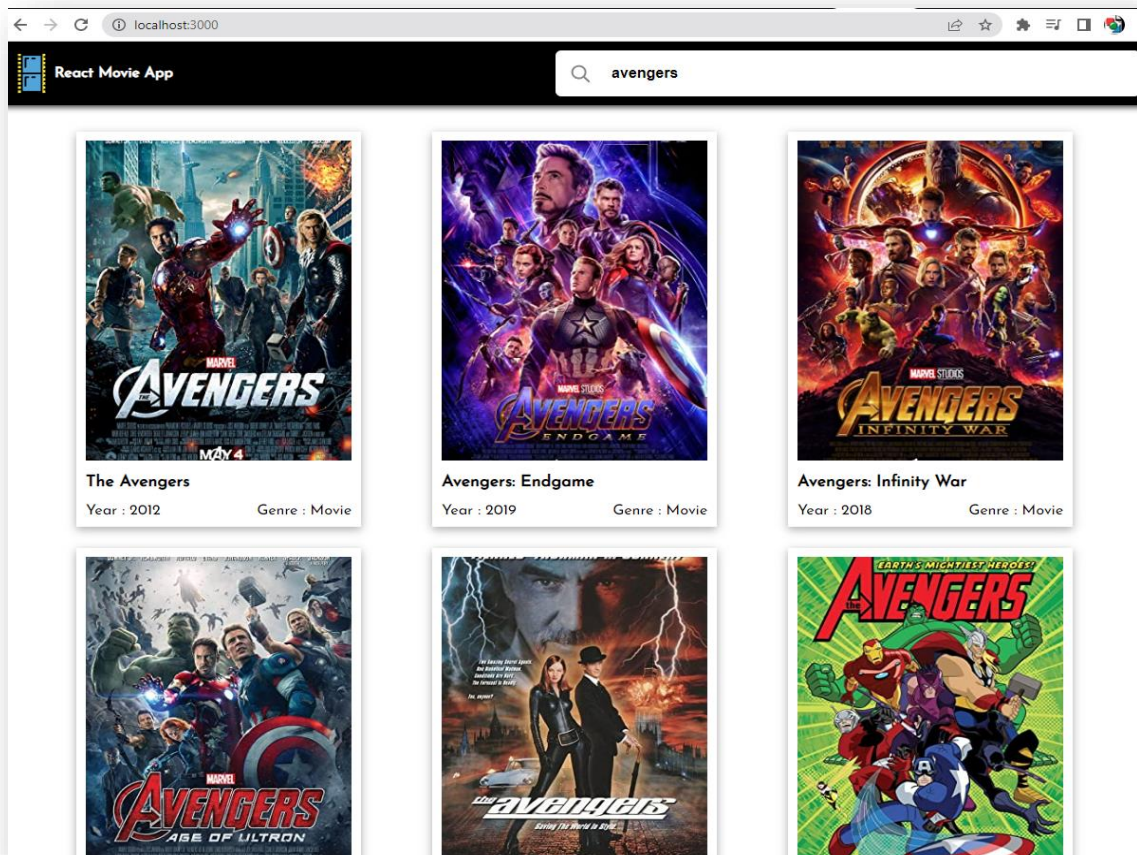
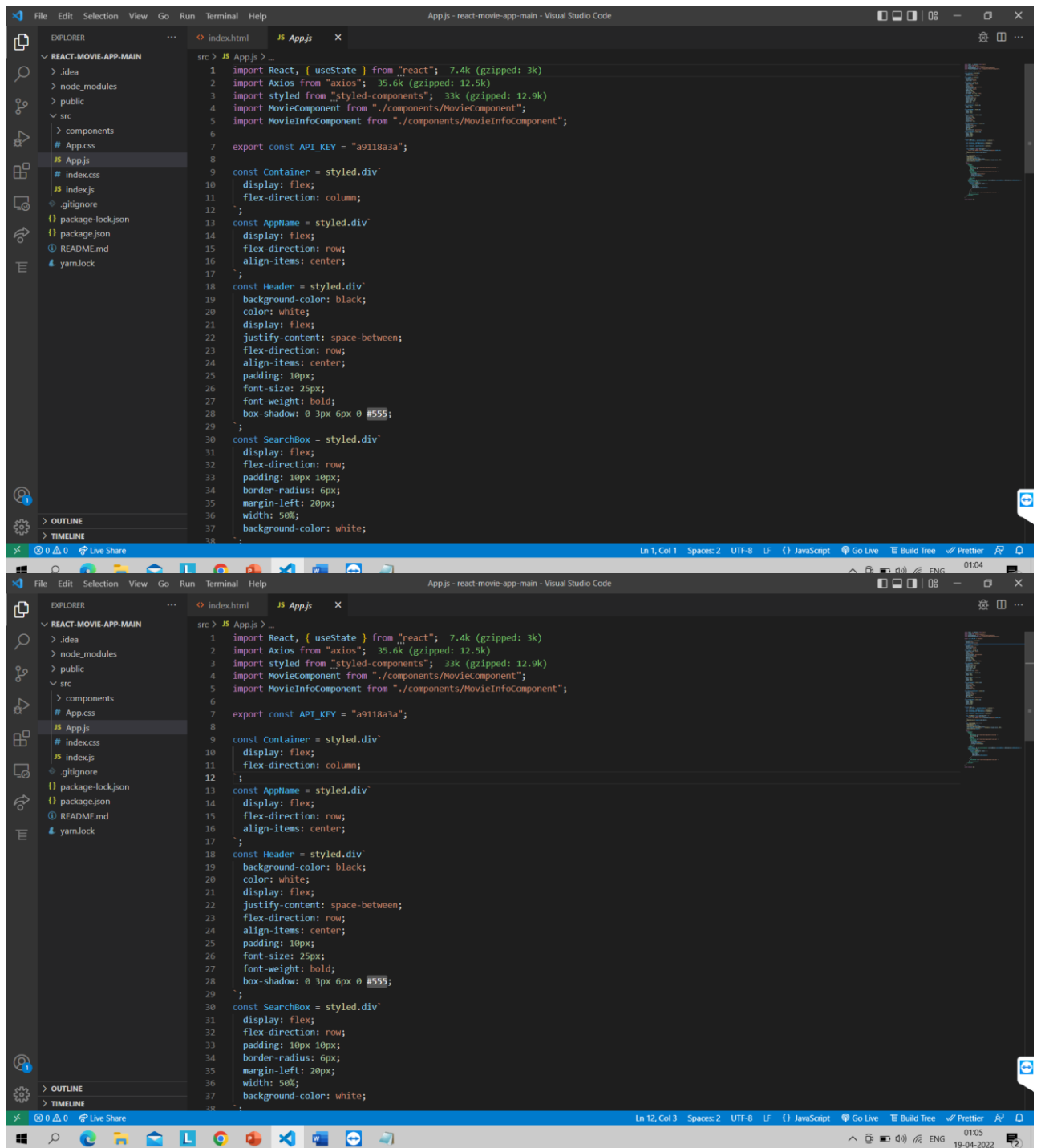
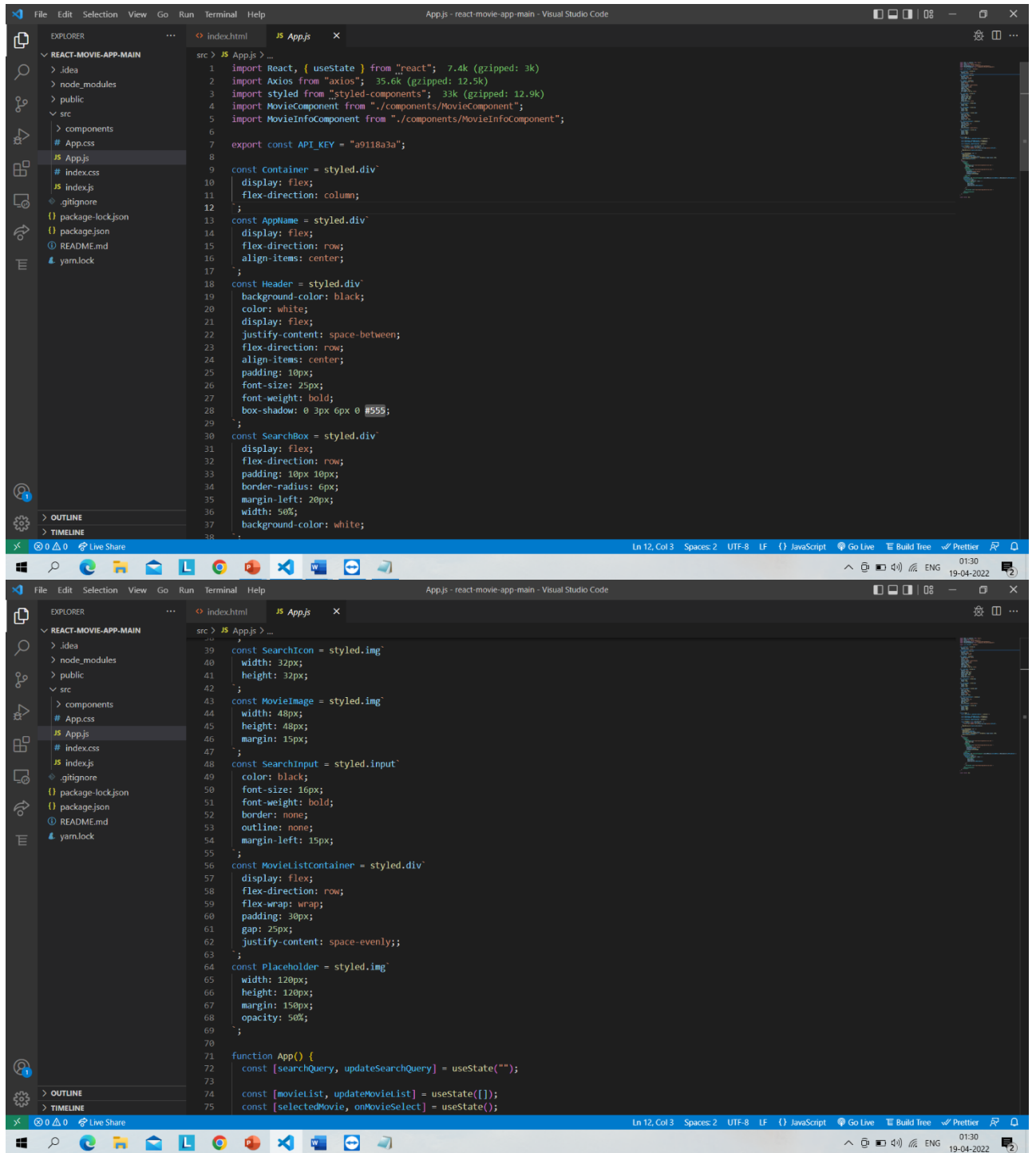
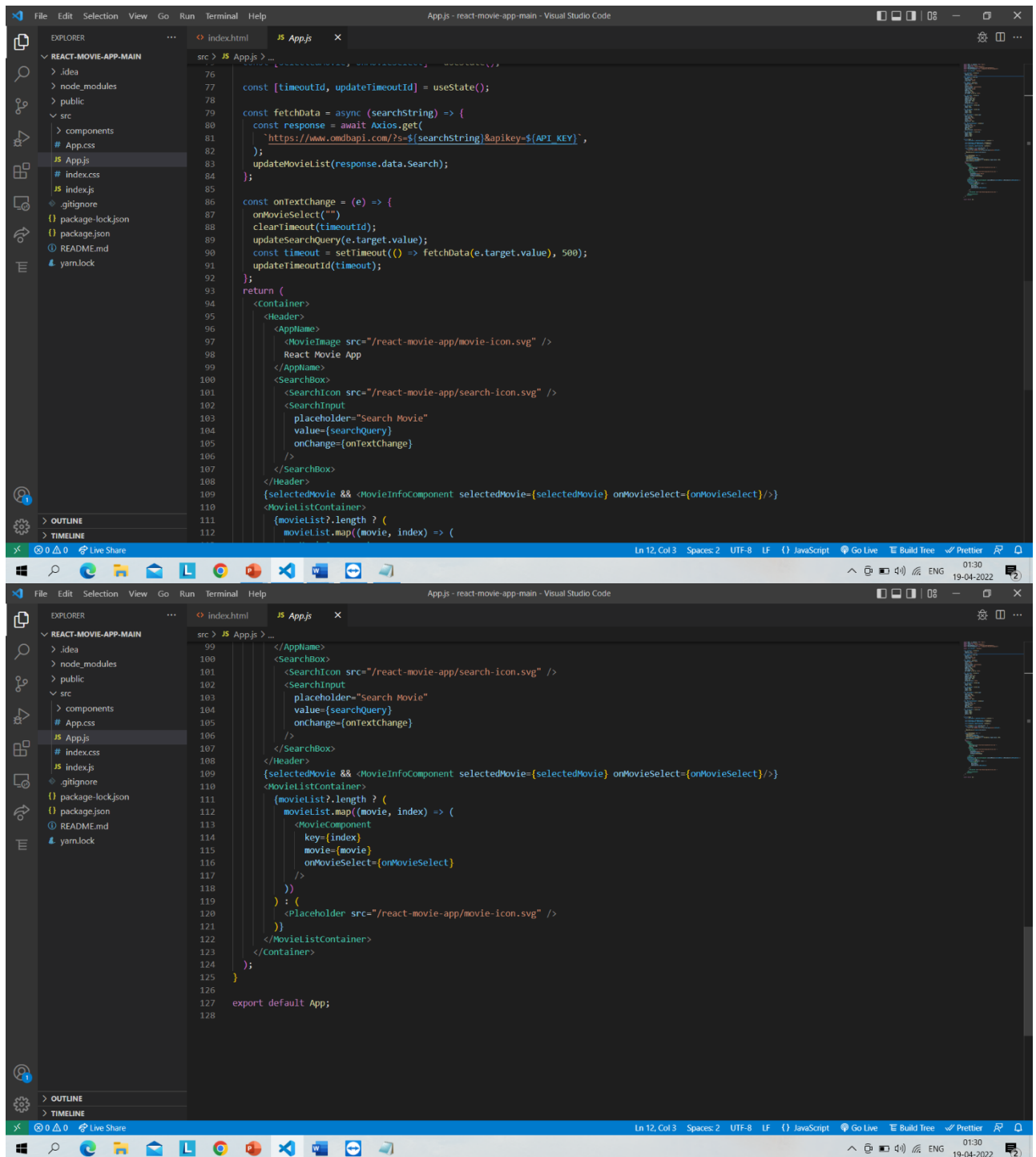
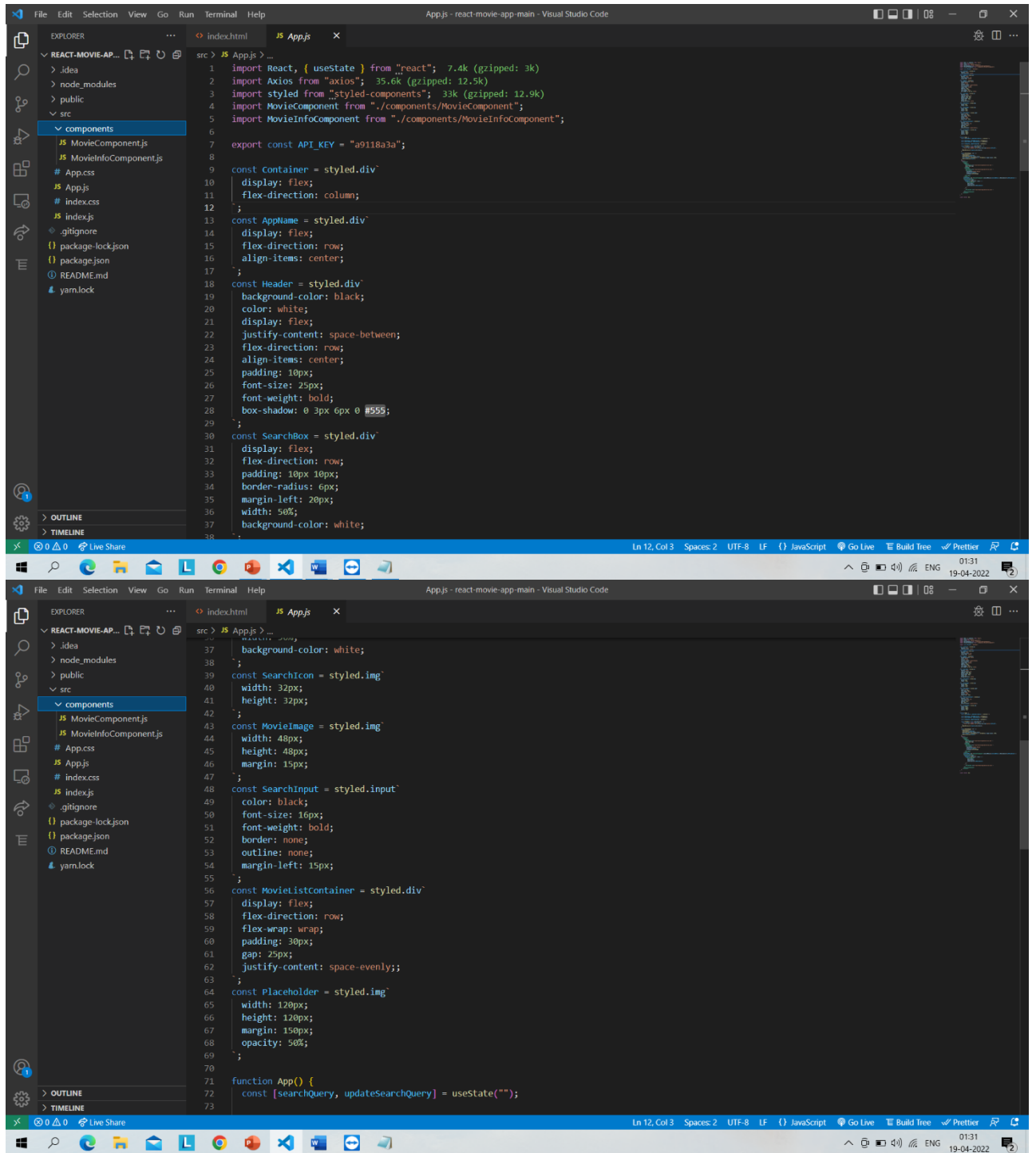


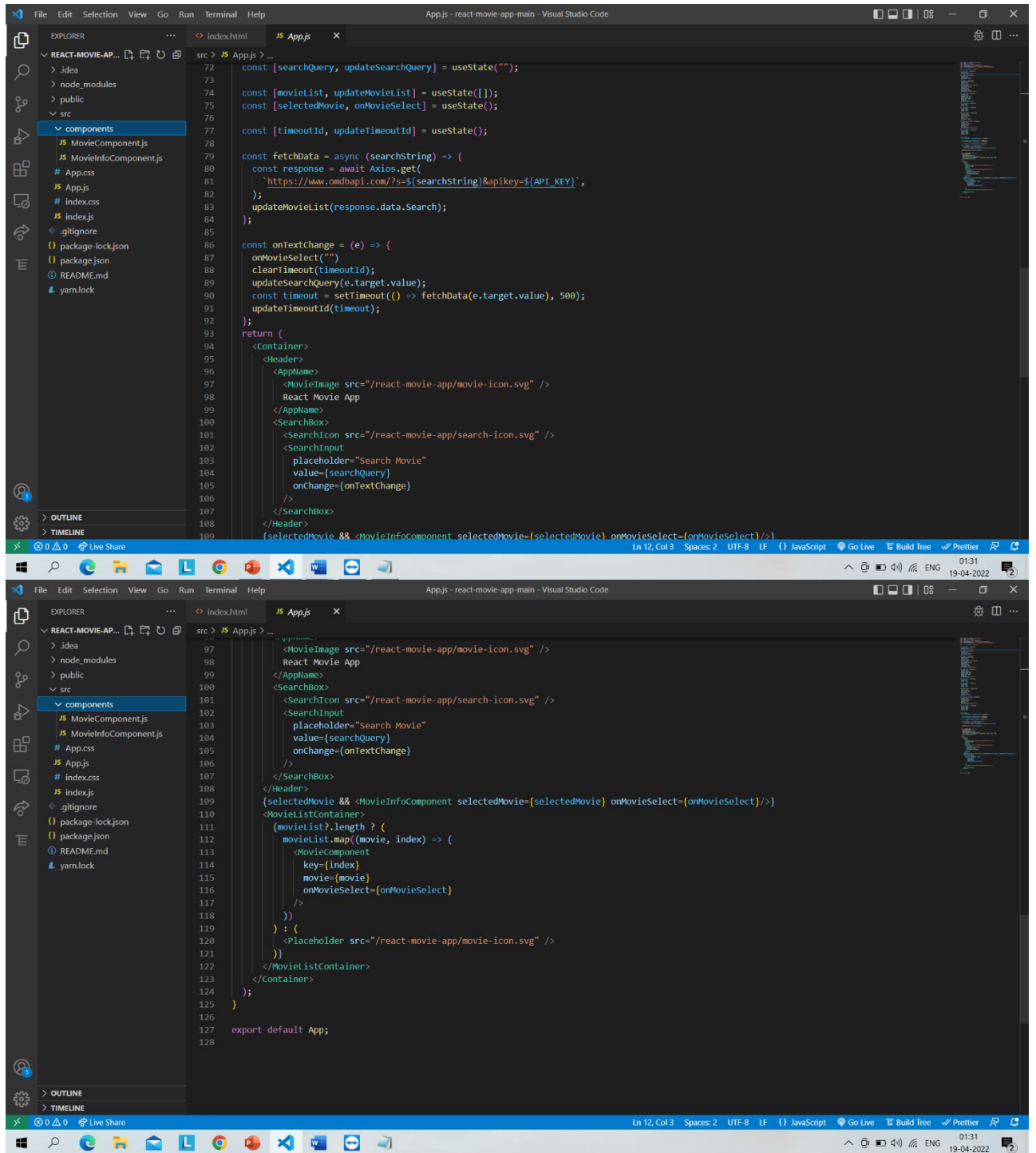
Fig 1.17 Final output











B. SOURCE CODE :

```
import React, { useState } from "react";
import Axios from "axios";
import styled from "styled-components";
import MovieComponent from "../components/MovieComponent";
import MovieInfoComponent from "../components/MovieInfoComponent";

export const API_KEY = "a9118a3a";

const Container = styled.div`
  display: flex;
  flex-direction: column;
`;

const AppName = styled.div`
  display: flex;
  flex-direction: row;
  align-items: center;
`;

const Header = styled.div`
  background-color: black;
  color: white;
  display: flex;
  justify-content: space-between;
  flex-direction: row;
  align-items: center;
  padding: 10px;
  font-size: 25px;
  font-weight: bold;
  box-shadow: 0 3px 6px 0 #555;
`;

const SearchBox = styled.div`
  display: flex;
  flex-direction: row;
  padding: 10px 10px;
  border-radius: 6px;
  margin-left: 20px;
  width: 50%;
  background-color: white;
`;

const SearchIcon = styled.img`
  width: 32px;
  height: 32px;
`;

const MovieImage = styled.img`
  width: 48px;
  height: 48px;
  margin: 15px;
```

```

`;
const SearchInput = styled.input`
  color: black;
  font-size: 16px;
  font-weight: bold;
  border: none;
  outline: none;
  margin-left: 15px;
`;
const MovieListContainer = styled.div`
  display: flex;
  flex-direction: row;
  flex-wrap: wrap;
  padding: 30px;
  gap: 25px;
  justify-content: space-evenly;
`;
const Placeholder = styled.img`
  width: 120px;
  height: 120px;
  margin: 150px;
  opacity: 50%;
`;

function App() {
  const [searchQuery, updateSearchQuery] = useState("");

  const [movieList, updateMovieList] = useState([]);
  const [selectedMovie, onMovieSelect] = useState();

  const [timeoutId, updateTimeoutId] = useState();

  const fetchData = async (searchString) => {
    const response = await Axios.get(
      `https://www.omdbapi.com/?s=${searchString}&apikey=${API_KEY}`,
    );
    updateMovieList(response.data.Search);
  };

  const onTextChange = (e) => {
    onMovieSelect("")
    clearTimeout(timeoutId);
    updateSearchQuery(e.target.value);
    const timeout = setTimeout(() => fetchData(e.target.value), 500);
    updateTimeoutId(timeout);
  };

  return (
    <Container>
      <Header>
        <AppName>

```

```

    <MovieImage src="/react-movie-app/movie-icon.svg" />
    React Movie App
  </AppName>
  <SearchBox>
    <SearchIcon src="/react-movie-app/search-icon.svg" />
    <SearchInput
      placeholder="Search Movie"
      value={searchQuery}
      onChange={onTextChange}
    />
  </SearchBox>
</Header>
{selectedMovie && <MovieInfoComponent selectedMovie={selectedMovie}
onMovieSelect={onMovieSelect}/>}
<MovieListContainer>
  {movieList?.length ? (
    movieList.map((movie, index) => (
      <MovieComponent
        key={index}
        movie={movie}
        onMovieSelect={onMovieSelect}
      />
    ))
  ) : (
    <Placeholder src="/react-movie-app/movie-icon.svg" />
  )}
</MovieListContainer>
</Container>
);
}

export default App;

```

REFERENCES

- <https://www.omdbapi.com/>
- OMDb Usage <http://www.omdbapi.com/#usage>
- <https://github.com/reactjs>
- <https://github.com/gitname/react-gh-pages>