A Software Project
On

# MOVIE EXPLORATION USING PYTHON

Submitted in partial fulfilment of requirements for the award of the degree of

**Bachelor of Technology**
**In**
**Computer Science and Engineering**

By

**M. Anil Kumar (o180677)**

Under the guidance of

**Mrs P. SINDHU,**(Assistant Professor)


**Department of Computer Science &Engineering**

**RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES**
**(An Autonomous Institution, Approved by AICTE)**

**Kurnool Road, Ongole, Prakasam(Dt.), AndhraPradesh – 523225**

# DECLARATION

We hereby declare that the project work entitled **MOVIE EXPLORATION USING PYTHON** submitted to the **RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES ONGOLE (IIITcampus)** in partial fulfilment of the requirements for the project of the **Software Engineering** in Computer Science and Engineering is a record of an original work done by us under the guidance of **P.SINDHU**,(Assistant Professor) and this project work have not been submitted to any other university for the project of any other degree or diploma.

Date: Signature

(M. Anil Kumar)

## ACKNOWLEDGEMENT

It is our privilege and pleasure to express a profound sense of respect, gratitude and indebtedness to our guide Mrs **P.SINDHU** (Assistant Professor), Dept. Of Computer Science and Engineering, RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES (IIIT Ongole campus) ,for his/her indefatigable inspiration, guidance, cogent discussion, constructive criticisms and encouragement throughout this dissertation work.

We express our sincere gratitude to Mr **Buthapati Sampath Babu Sir,**(Assistant Professor and Head Department of Computer Science and Engineering), RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES (IIIT Ongole campus), for his suggestions, motivations and co-operation for the successful completion of the work.

We extend our sincere thanks to **Mr.RUPAS KUMAR Sir**, Dean Academics, RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES (IIIT Ongole campus) for his encouragement and constant help.

We extend our sincere thanks to **DR.B.JAYARAMI REDDY Sir**, Director of RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES (IIIT Ongole campus).

I have taken efforts in this project. However, it could not have been possible without the kind support and help of many individuals and RGUKT. We would like to extend
sincere thanks to all of them.

My thanks and appreciation also go to my colleagues in developing the project and people who have willingly helped me out with their abilities.

Date:

With sincere Regards
M. Anil Kumar

## ABSTRACT

Here the user can request information about his/her Desired Movie. We provide Information like Cast,Crew,Story line,Genre,Reviews and Ratings. We also provide related movie photos & videos and social links. User can also know about Movie budget ,gross and collections.User can also know about upcoming movies and shows. The users can also get the latest movie availability in there Desired loacality. Movie ratings and reviews at sites such as IMDb or Amazon are commonly used by moviegoers to decide which movie to watch or buy next. Currently, moviegoers base their decisions as to which movie to watch by looking at the ratings of movies as well as reading some of the reviews at IMDb or Amazon. Reviewers movie scores and reviews can be analyzed with respect to their emotion content, aggregated and projected onto a movie, resulting in an emotion map for a movie. One can then make a decision on which movie to watch next by selecting those movies having emotion maps with certain emotion map patterns desirable for him/her. It is common knowledge that, usually, moviegoers, called users or reviewers in the rest of this application, utilize movieratings and reviews in selecting their next movie to see/watch.This is indeed the case for the authors of this paper. And,unfortunately, sometimes movie reviews and ratings do not help users make the right choices, as evidenced by their emotional feelings after watching the movie.This is perhaps because users desire a certain emotional state after watching a movie, which does not match the emotions evoked by the selected movie.

# Table of Contents


| S.No | Topic | Page No |
|---|---|---|
| 1 | Introduction | 1 |
| | 1.1. Motivation | 1 |
| | 1.2. Problem Definition | 1 |
| | 1.3. Objective of the Project | 1 |
| 2 | Analysis | 2 |
| | 2.1. Existing System | 2 |
| | 2.2. Proposed System | 2 |
| | 2.3. Future Scope | 2 |
| | 2.3.1 Project Requirements | 3 |
| | 2.3.2 Hardware Requirements | 3 |
| 3 | Literature review | 4 |
| 4 | Design | |
| | 4.1. UML diagrams | 5 |
| 5 | Implementation | |
| | 5.1. Modules | 20 |
| | 5.2. Module description | 20 |
| | 5.3. Introduction of technologies used | 21 |
| | 5.4. Sample Code | 22 |
| 6 | Testcases | 44 |
| 7 | Screenshots | 45 |
| 8 | Conclusion | 48 |
| 9 | Future Enhancement | 49 |
| 10 | Bibliography | 50 |

# 1.INTRODUCTION

Here we are introducing a application named Movie Exploration Using Python.Here we get reviews and videos of the user desired movies and webseries.We get the information like cast,crew, ratings,reviews and budget of all languages movies and web series.Here we get trustworthy content of each and every movie.This application gives accurate reviews and ratings of all movie.We include all geners like comedy,horror,drama,romance,thrillers,action movies.

## 1.1:MOTIVATION:

**Giving the accurate reviews:**Movie exploration helps us in giving accurate reviews of the movie according to our fantasies.

**Get Entertainment:**Every age people need some fn and entertainment in their lives.We are adding entertainment content for them.

**Information collection:**This movie Exploration project is the collection every industry and every genre of movies.

**Insight of movies:**We have content about each and every ones individual desired movies.

## 1.2.PROBLEM DEFINATION:

Who likes to watch movies here?! And who here have checked IMBD ratings before watching any movie? So,adding reviews on movie is a common tendency .Let’s build a neural network from scratch which can determine the sentiment on a movie review.Pretty cool,huh!? Here the user can request information about his/her Desired Movie. We provide Information like Cast,Crew,Story line,Genre,Reviews and Ratings. We also provide related movie photos & videos and social links. User can also know about Movie budget ,gross and collections.User can also know about upcoming movies and shows.

## 1.3.OBJETIVE OF THE PROJECT:

Main objective of the project is to give c.omplete accurate and faithful review to the users.We are adding some simple videos and trailers of the content what the user need .This application is trust worthy for each and every user.Every person have their individual tastes on movies.Some one prefers horror,some one thrillers,some one love etc.. So we including every genre and every language information.

## 2.ANALYSIS

### 2.1 EXSISTED SYSTEM:

Existed consists of only reviews and ratings of movie which has been entered by the user.Existed system provide information like cast,crew and their co ordination.Currently, moviegoers base their decisions as to which movie to watch by looking at the ratings of movies as well as reading some of the reviews at IMDb or Amazon. And,unfortunately, sometimes movie reviews and ratings do not help users make the right choices, as evidenced by their emotional feelings after watching the movie.

### 2.2 PROPOSED SOFTWARE:

*Features:*

- Remainder
- Audio to text generator
- Relaxing music
- Embedding search engine
- Editing and Efects
- Sound and Music
- Cinematography

### 2.3 FUTURE SCOPE OF THIS SOFTWARE:

Our main idea for building this software to give society a better environment. Movie exploration:This feature helps people to feel comfortable while working and help thems tho feel relax. Audio to text generator:With the help of this we can easily convert any important audio file into text.Every stage involved in making a film needs a lot of effort and hard work,and it a career that a wide scope and opportunities Film making is a long process invovling a group of individuals with different skills and sets of ideas.

## 2.3.1PROJECT REQUIREMENTS:

### Software Requirements:

- Python
- D Jango
- HTML
- CSS
- Java script
- Python modules
-

## 2.3.2 Hardware Requirements:

Android Studio with minimum of 8GB RAM

For Figma your operating system has to be windows 8.1 or later

Processor:Multi-core Intel i5/i7/i9 models(i9 is best)

Storage:At least 256 GB hard drive,7200 RPM, preferably SSD(fatest),Hdd also good.....

Graphics Card:Depends on video editing software.

## 3. LITERATURE REVIEW:

We have reviewed several papers and articles based on Movie Review and Exploration.

**1**.Title of the Paper :A Case-Based Recommendation Approach for Market Basket Data.
Key points:CF; CB; AR (Association Rule); CBR (Case-Based Reasoning)
Conclusion:After compared the performance of developed RS conclude that CBR is the good method in case of transactions
References:A Case-Based Recommendation Approach for Market Basket Data Anna Gatzioura and Miquel Snchez-Marr IEEE INTELLIGENT SYSTEMS 2015.

**2**.Title of the Paper : Recommender Systems: An overview of different approaches to recommendations
Key points:Recommendation System; Information Retrieval System; CF; CBF; Hybrid Filtering
Conclusion:The three approaches of recommendation system and their advantages and disadvantages.
References:Recommender Systems: An overview of different approaches to recommendations Kunal Shah, Akshaykumar Salunke, Saurabh Dongare, Kisandas Antala SIT, Lonavala India 2017.

**3**.Title of the Paper : Recommendation analysis on Itembased and User-based Collaboration Filtering
Key points:IBCF; UBCF; Recommender System
Conclusion:IBCF and UBCF with implementation metrics, and conclude that IBCF provide better results than UBCF.
References:Recommendation analysis on Item-based and Userbased Collaboration Filtering GarimaGupta, Rahul Katarya, India.

**4.** Title of the Paper : Towards privacy in a context-aware social network based recommendation system
Key points:Content aware; social networking; privacy
Conclusion:Focus on protecting data and request for data, at the point of data collection.
References:P. W. Yau and A. Tomlinson, “Towards Privacy in a Context Aware Social Network Based Recommendation System,” Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE

Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on, Boston,MA,2011,pp.862-865. Doi:10.1109/PASSAT/SocialCom.2011.87.

**5**.Title of the Paper : A study of hybrid recommendation algorithm based on user.
Key points:Personalization; recommendation technology; collaborative filtering; hybrid algorithm
Conclusion:Hybrid algorithms are generates the results according to user’s rating and history record.
References:“A Study of Hybrid Recommendation Algorithm Based On User” Junrui Yang1, Cai Yang2, Xiaowei Hu3 2016 8th International Conference on Intelligent HumanMachine Systems and Cybernetics
Here we took references from all these IEEE papers of different projects and developing our website Movie Exploraiton

## 4. UML DIAGRAMS:

A UML Diagram is based on UML(Unified Modelling Language)with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain,or document information about the system. The UML diagrams are divided into Structural and Behavioral UML Diagrams.

### Structural diagrams:

Structural diagrams depict a static view of a structure of a system. It is widely used in the Documentation of software architecture. The Structural UML Diagrams involves 7 diagrams They are:

Class

Object

Component

Compsite structure

Profile

Package

Deployment

### Behavioural diagrams:

Behavioural diagrams portray a dynamic view of a system or the behaviour of a system, which describes the functioning the system. it involves 7 diagrams They are:

State machine

Sequential

Use case

Interaction

Timing

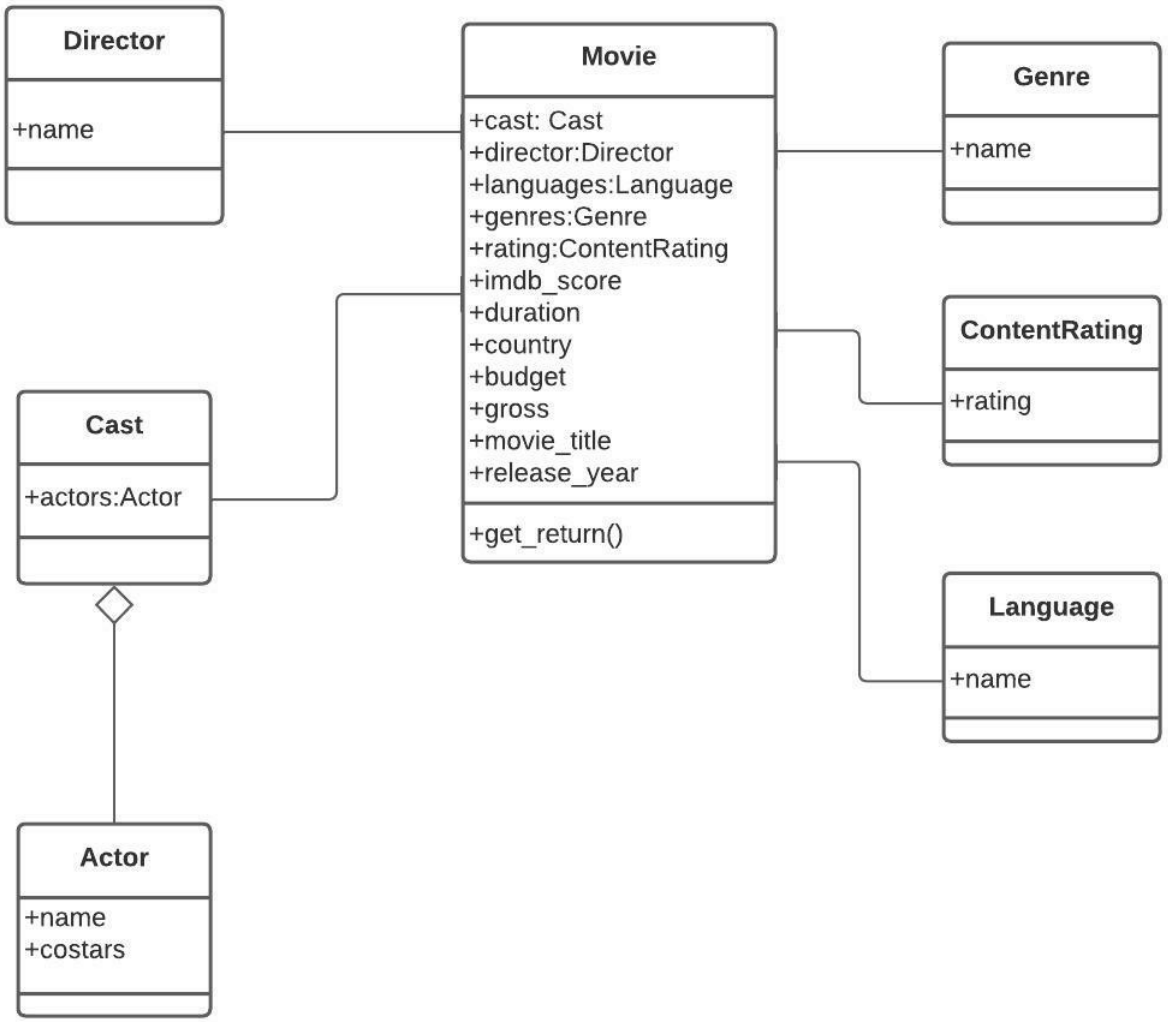
Activity

Communication

## STRUCTURAL DIAGRAMS:

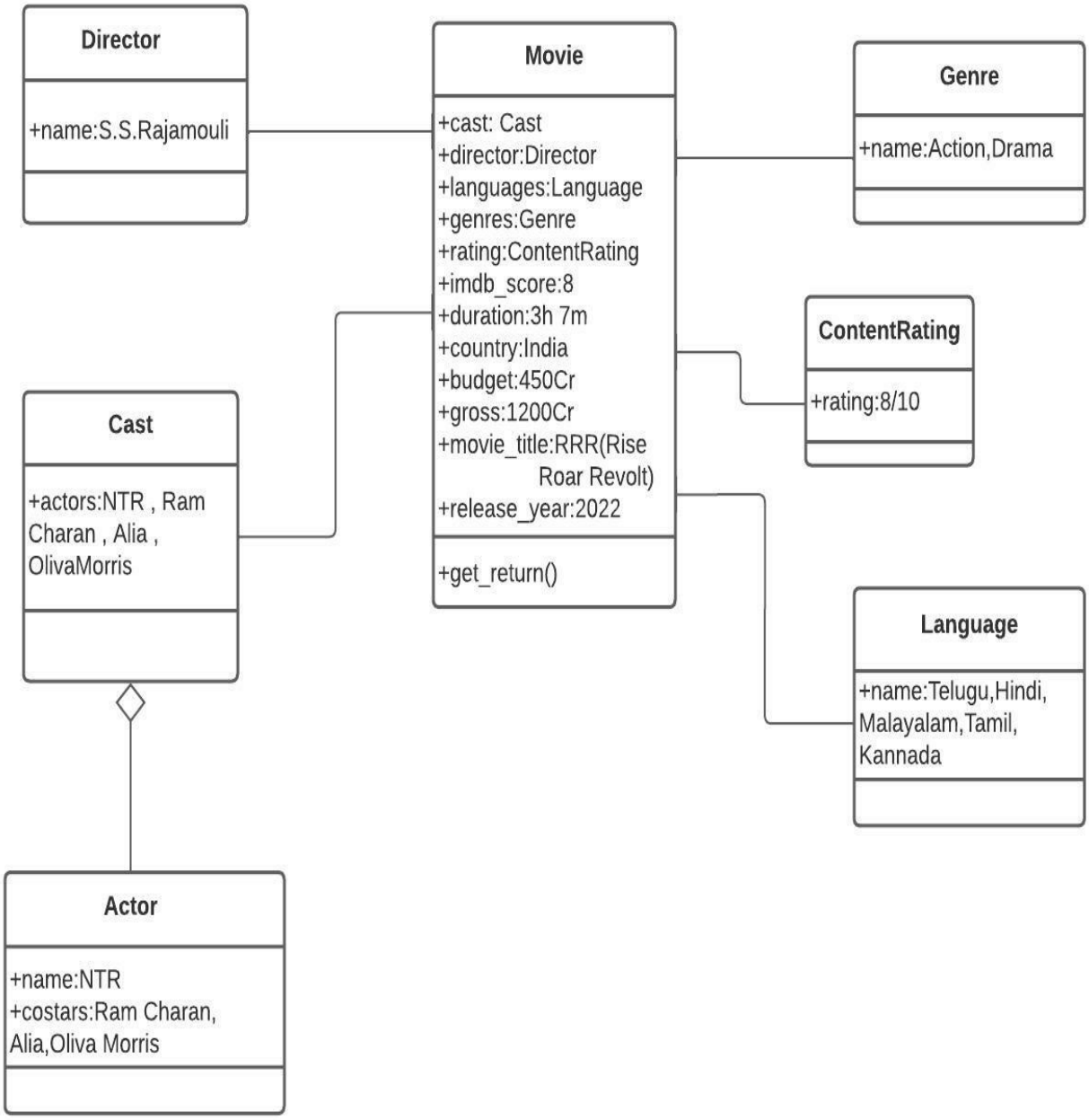
## 1.Class diagram:

Class diagrams are one of the most widely used diagrams.It is the back bone of all the object-oriented software systems.It depicts the static structure of the system.It displays the system's class,attributes,and methods.



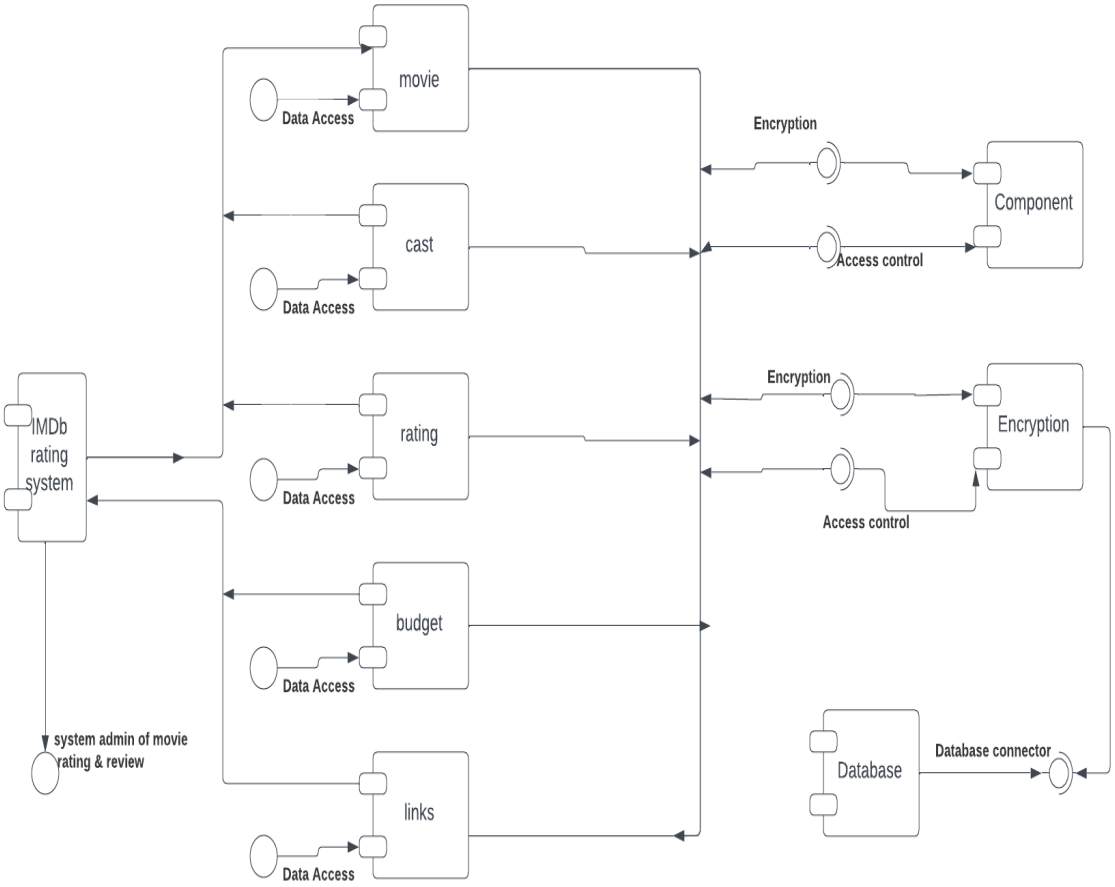
## 2.Object diagram:

It describes the static structure of a system at a particular point in time.It can be used to test the accurancy of class diagrams.It represents distinct instances of classes and the relationship between them at a time.



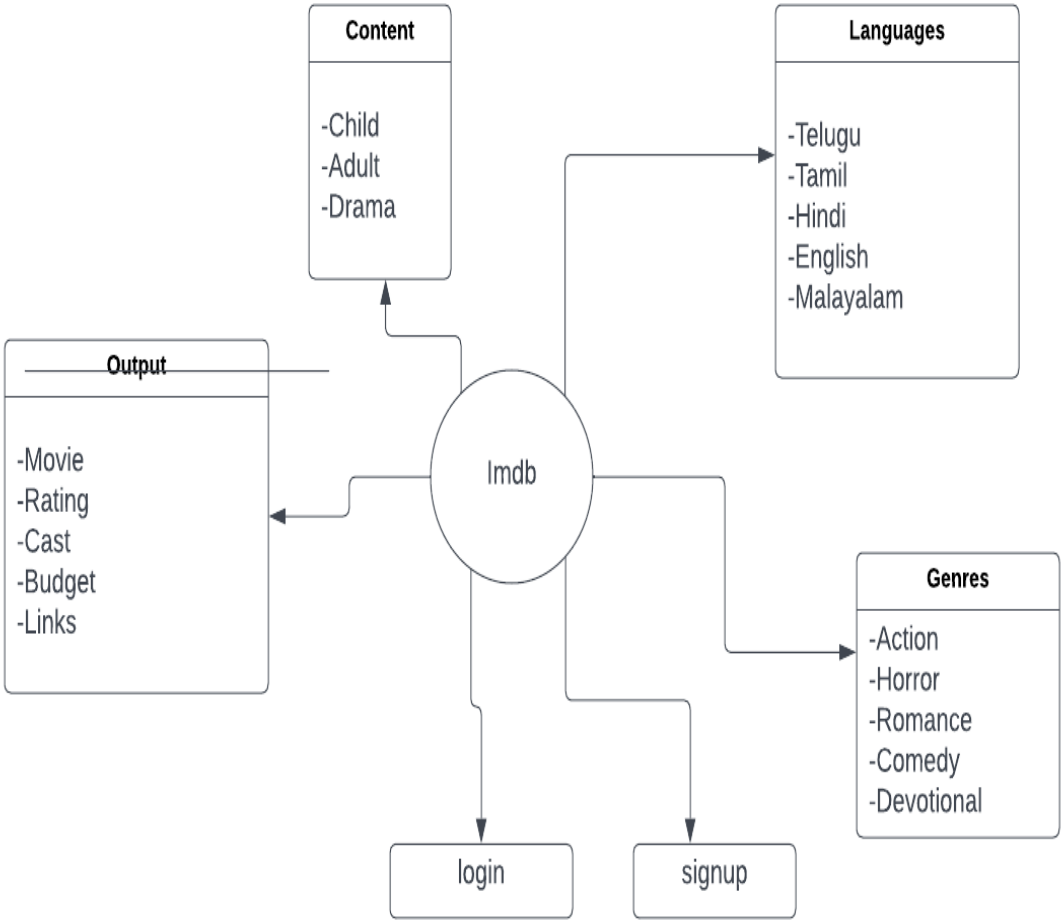
## 3.Component diagram:

Component diagrams are used in modelling the physical aspects of object-oriented systems that are used for visualizing,specifying,and documenting component-based systems and also for constructing excuteable systems through forward and reverse engineering.



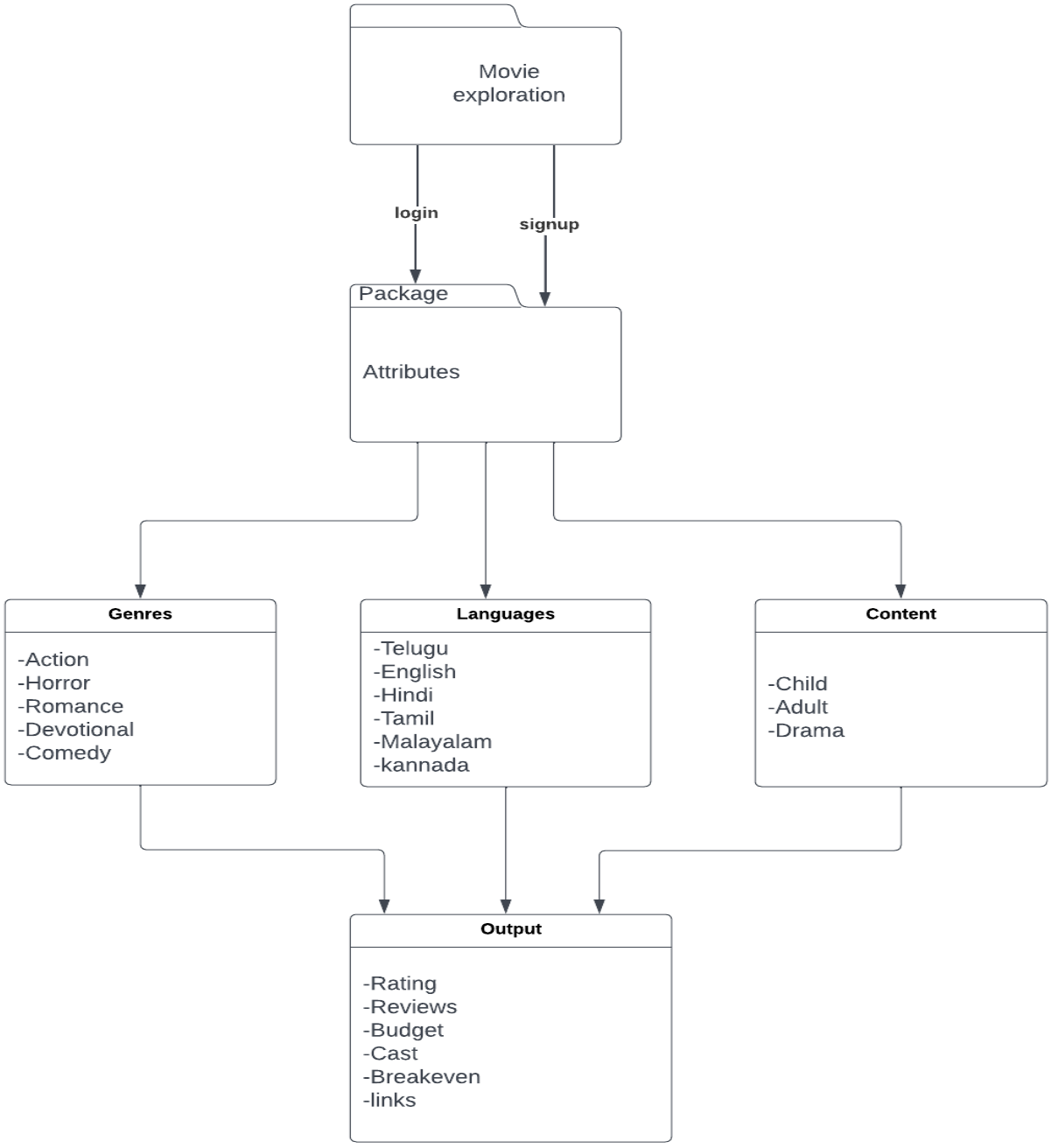
## 4.Composite structure:

Commposite structure diagram is one of the new artifactes added to uml 2.0. Acomposite structure diagram is a uml structural diagram that contains classes, interfaces, packages, and their relationships and that provides a logical view of all, or part of a software system.



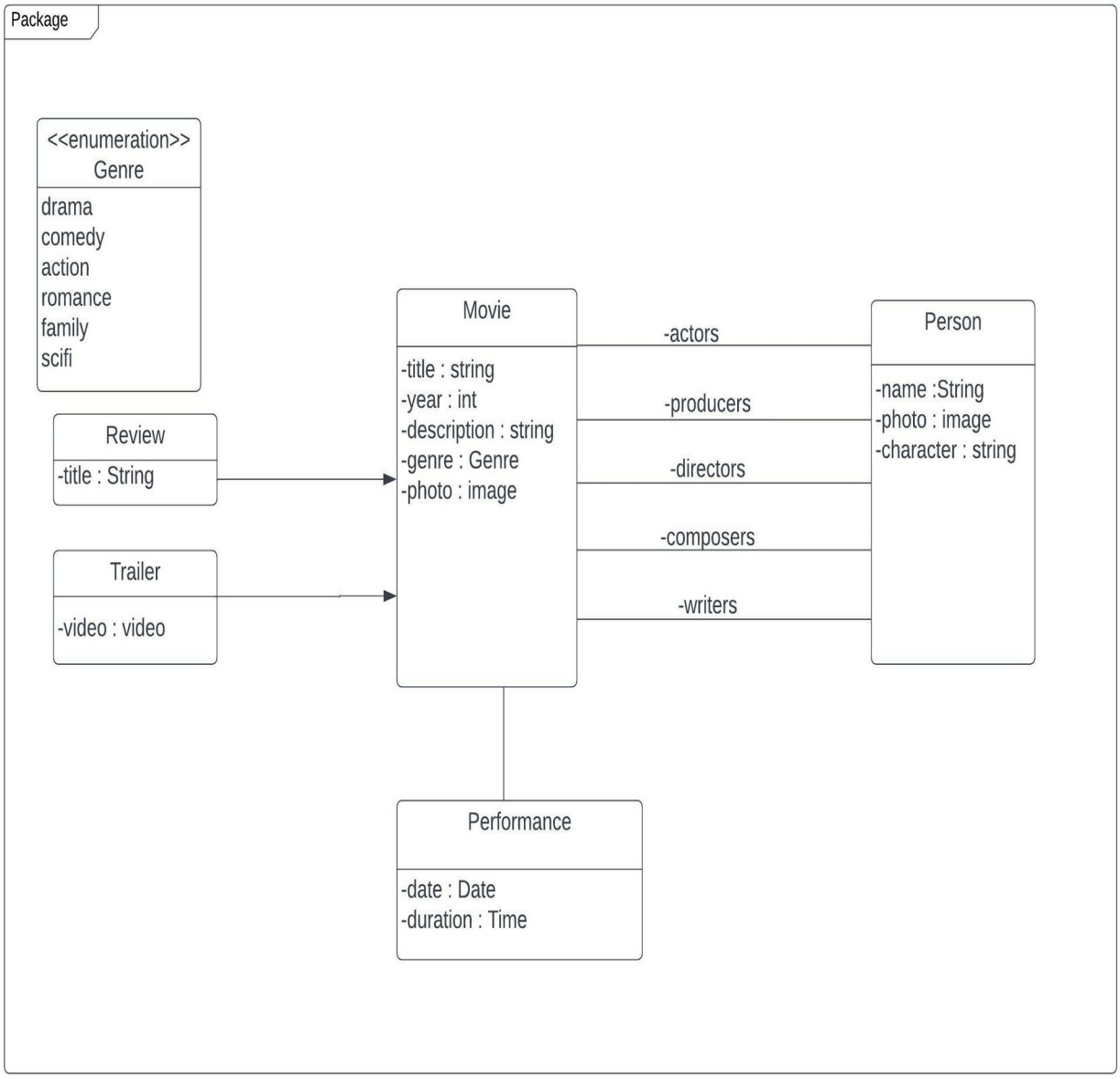
## 5.Profile diagram:

Profile diagram is basically an extensibility mechanism that allows you to extend and customize uml by adding new building blocks,creating new properties and new semantics in order to make the language suitable to your specific problem domain.



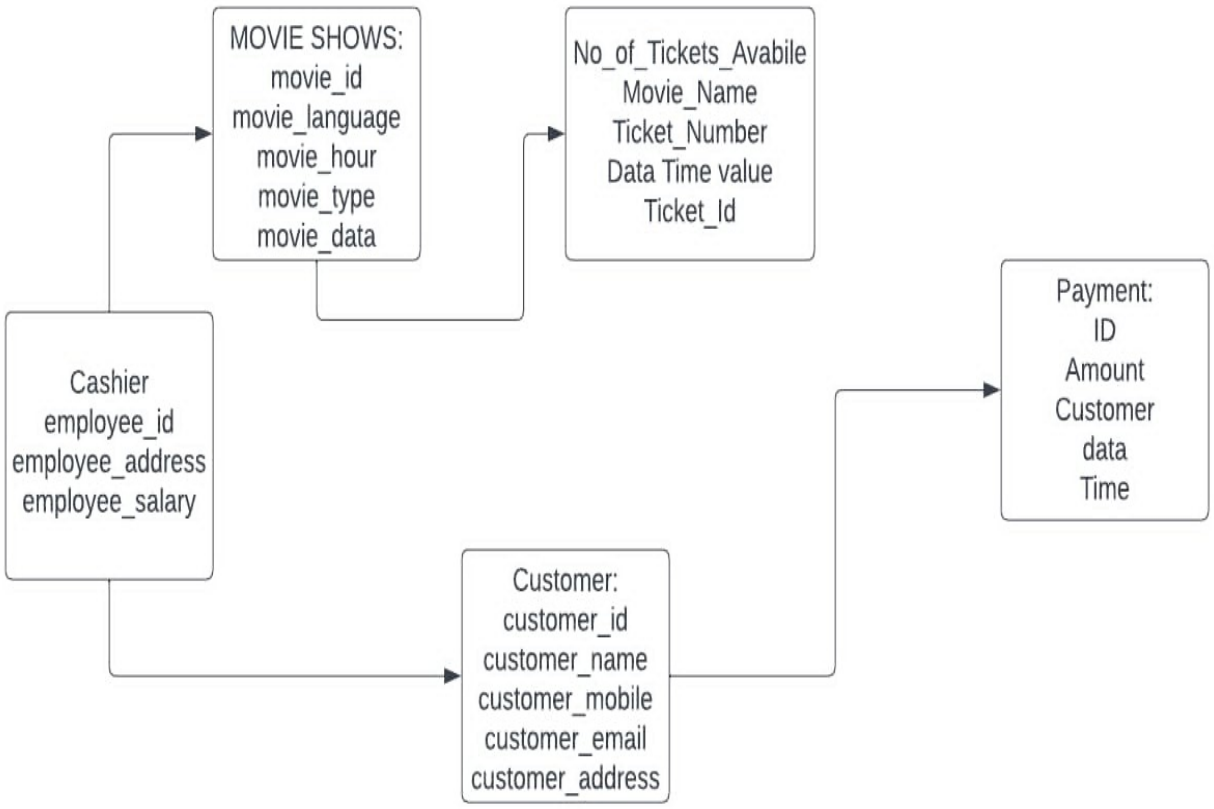
## 6.Package:

Package diagrams are the used in the path,to depict import access dependence between packages,classes,components,and other named elements within your system.Each dependency is rendered as a connecting line with an arrow representing the type of relationship between the two or more elements.



## 7.Deployment:

A deployment diagram is a uml diagram type that shows the execution achitecture of a system,including nodes such as hardware or software execution environments,and the middleware connecting them.

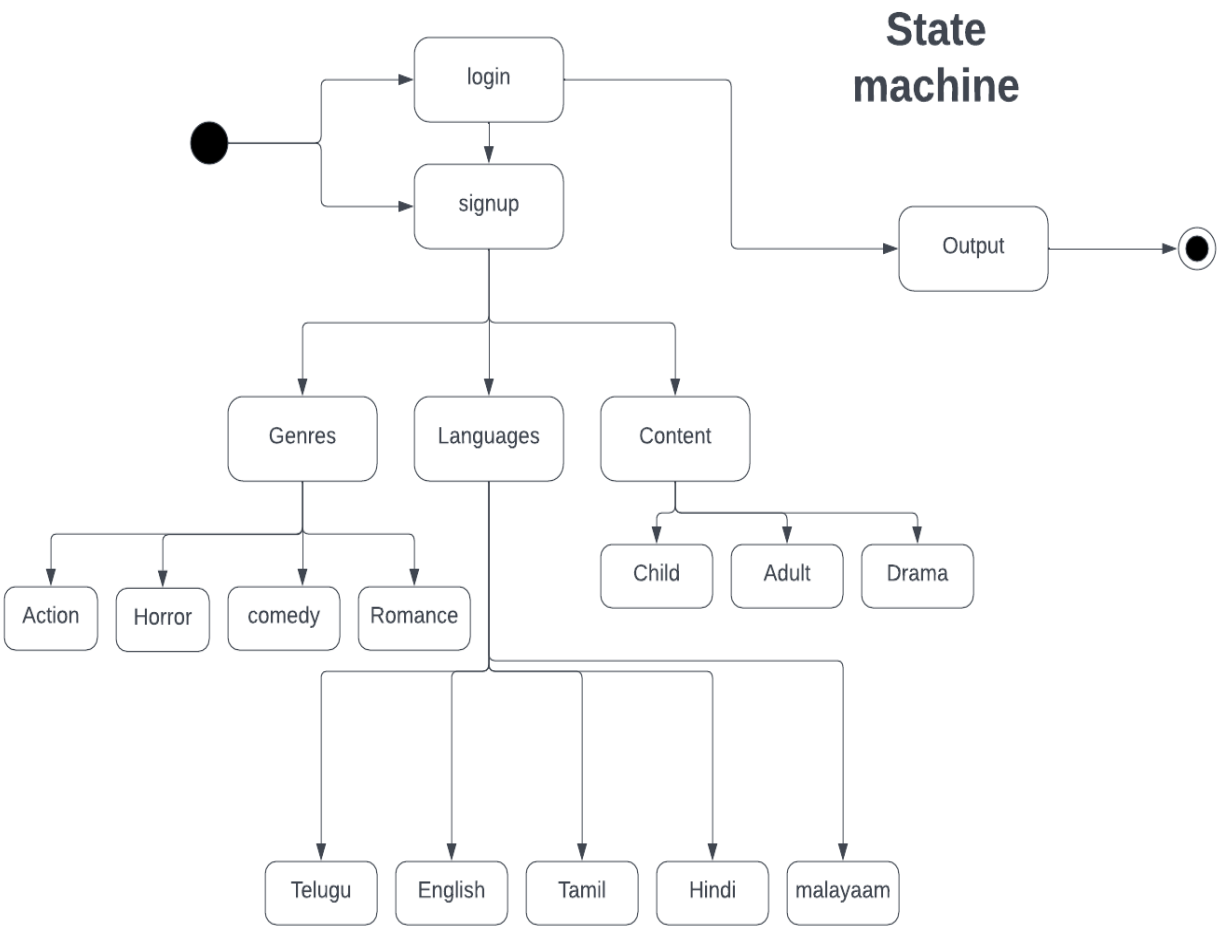



Deployment Diagram

## BEHAVIOURAL DIAGRAMS:

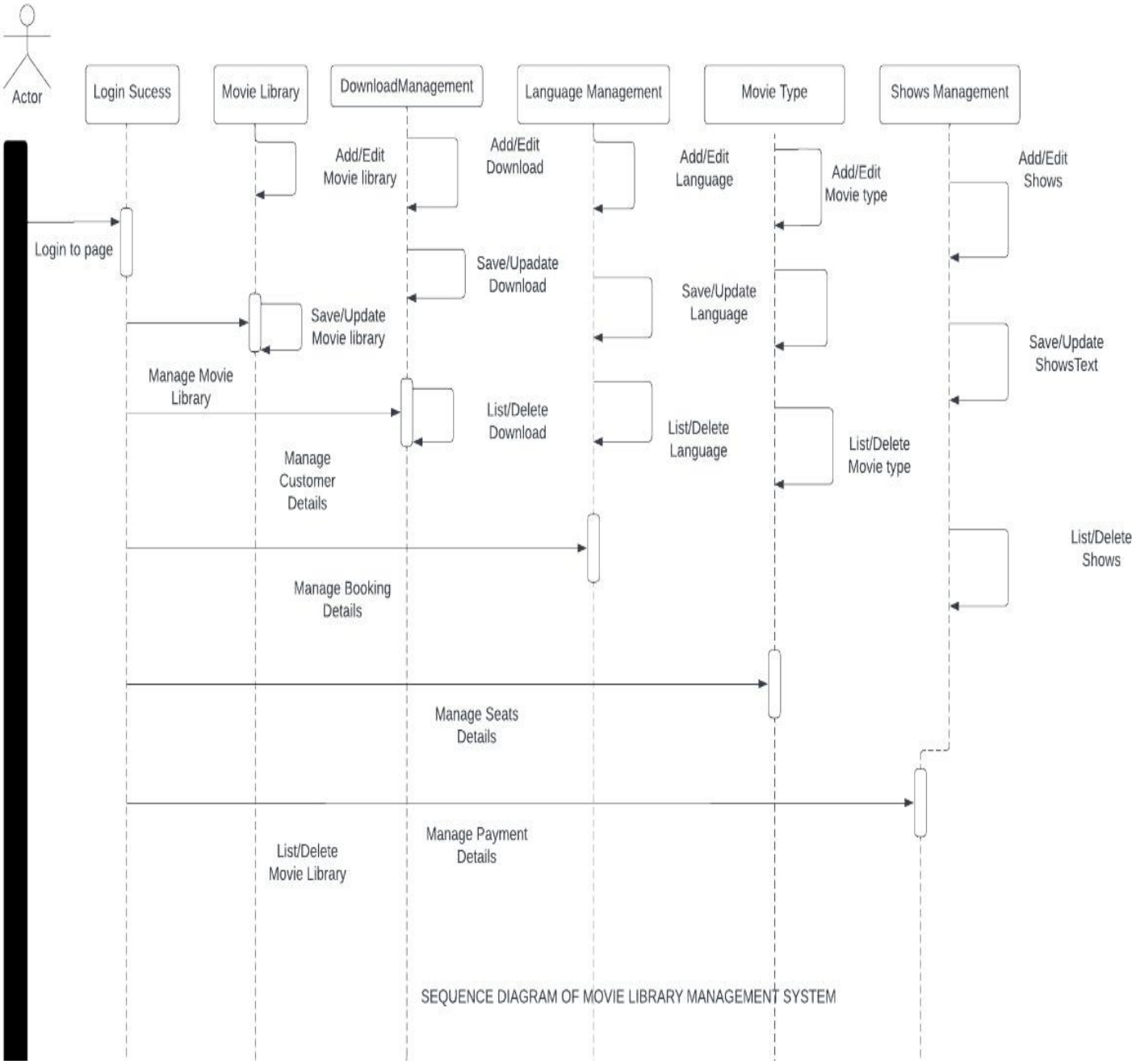
### 1.State machine:

A state diagram,also known as a state machine diagram or statechart diagram,is an illustration of the states an object can attain as well as the transitions between between those states in the unified modelling langauage.



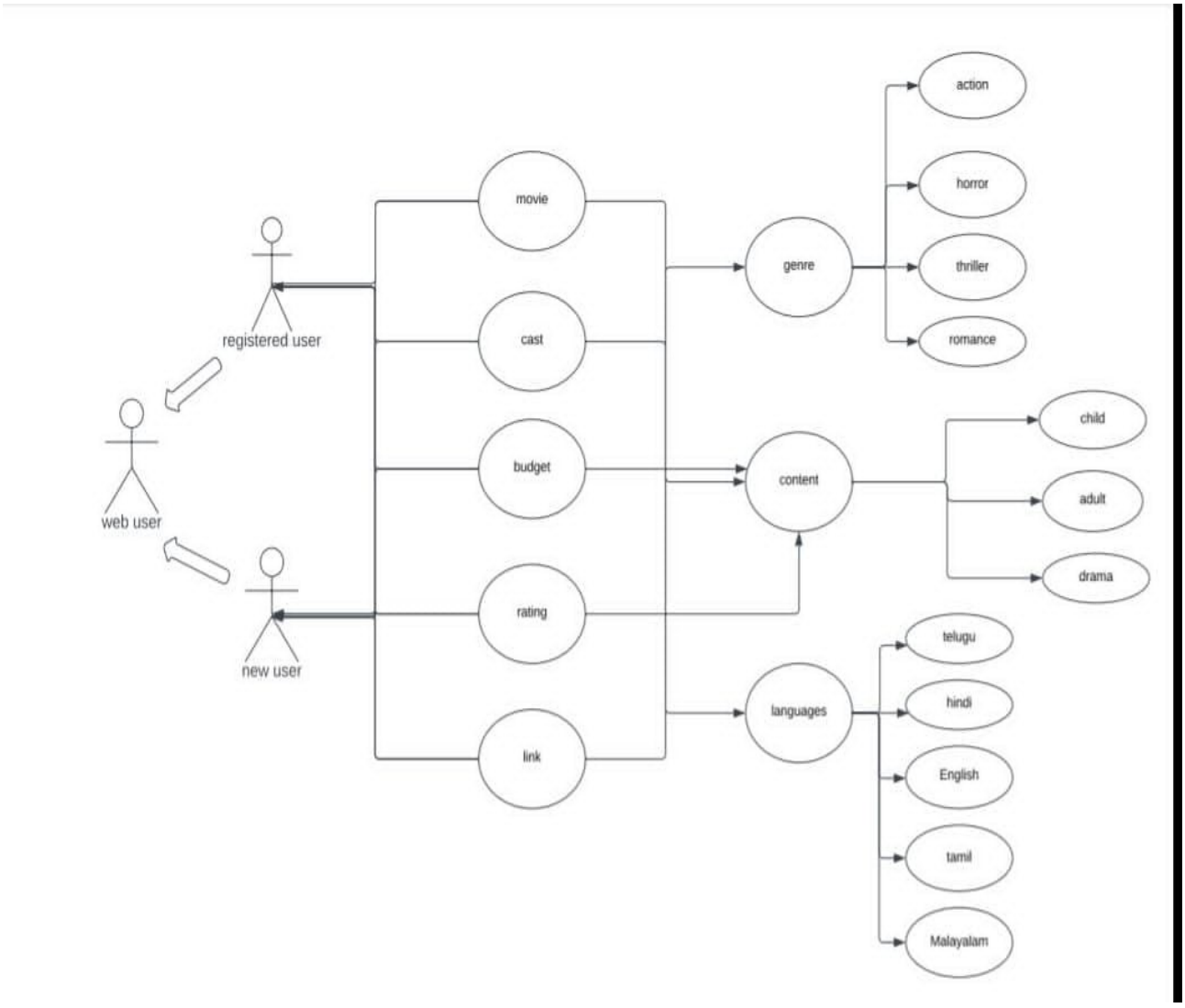
## 2.Sequential:

A sequence diagram consists of a group of objects that are represented by lifelines,and the messages that they exchange overtime during the interaction.A sequence diagram shows the sequence of message passed between objects.



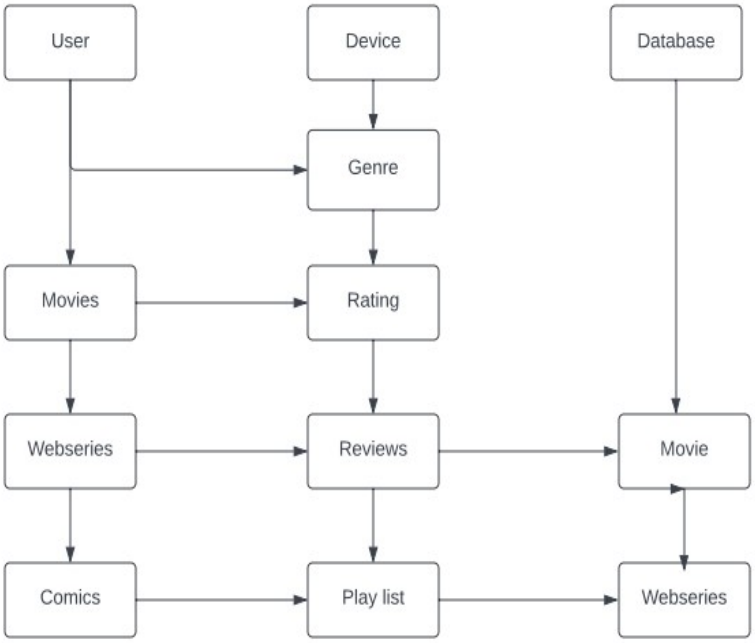
## 3.Use case:

Use case diagrams describe the high-leveled functions and scope of a systens.These diagrams also identify the interaction between the system and it’s actors.The use case and actors in use-case diagrams describe what the system does and how the actors use it,but not how the system operates internally.



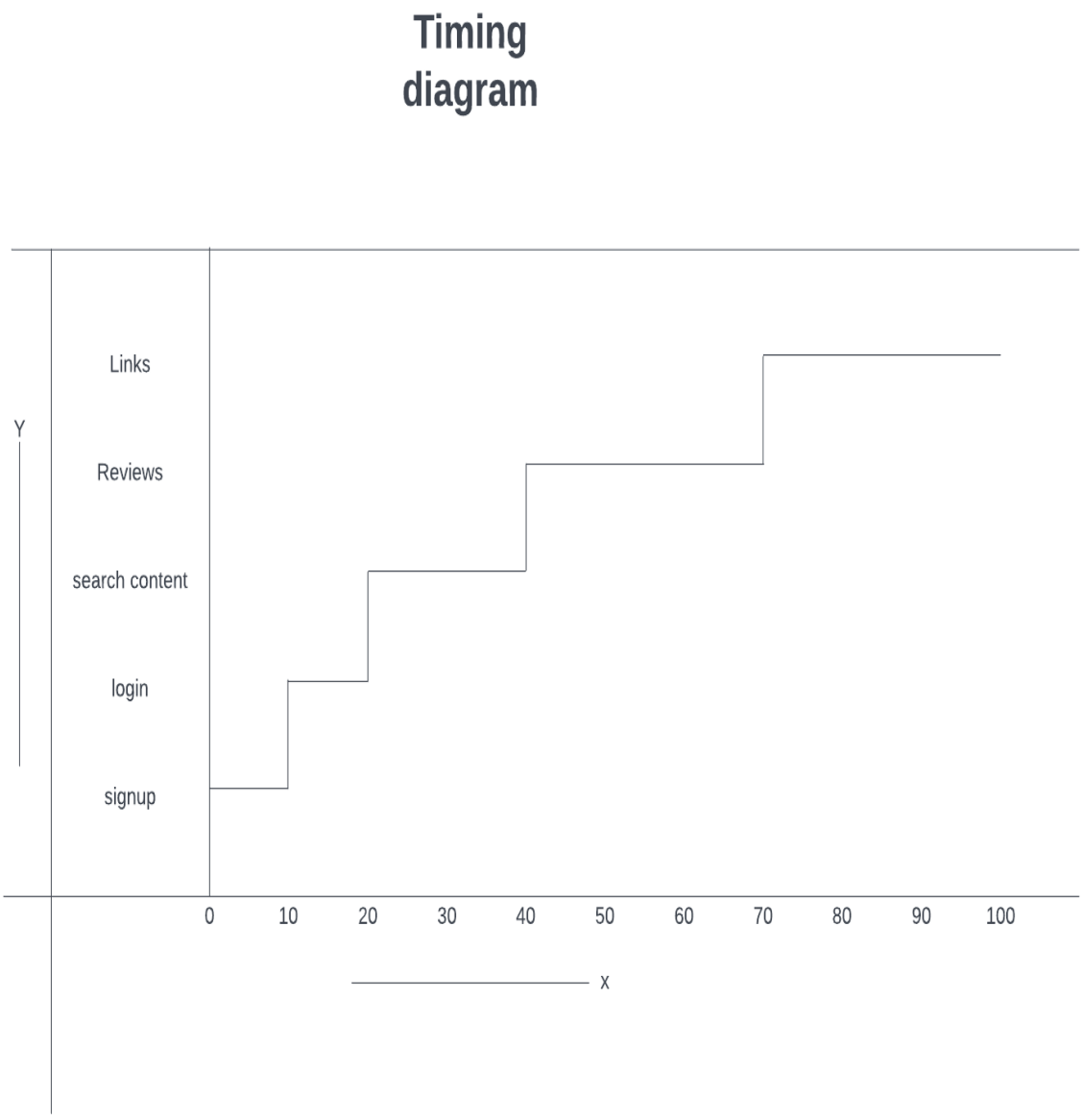
## 4.Interaction:

A interaction diagram is a type of UML diagram that's used to capture the interactive behaviour of a system.Interaction diagrams focus on describing the flow of messages within a system,providing context for one or more lifelines within a system.



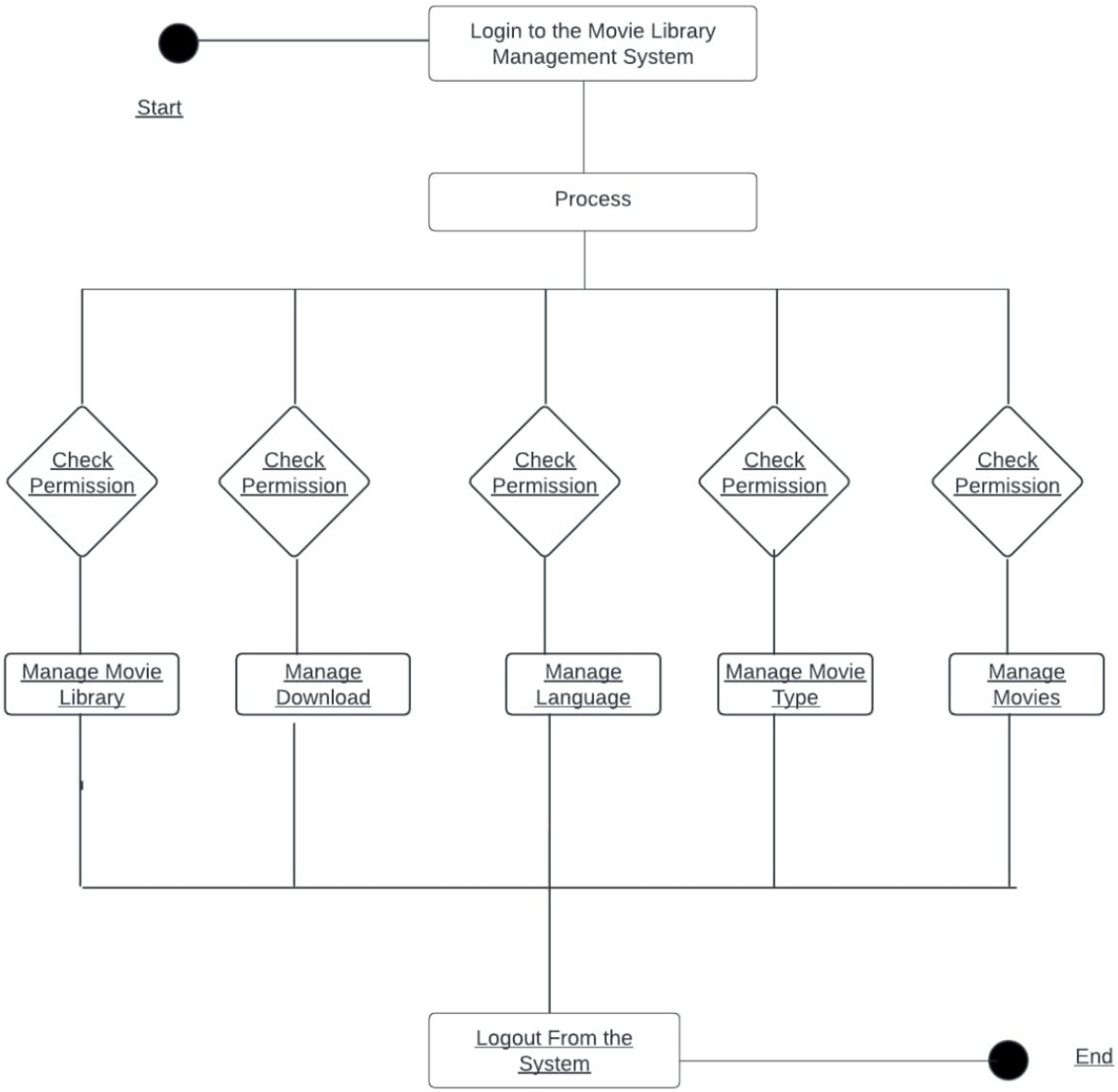
## 5.Timing:

A timing diagram includes timing data for atleast one horizontal lifeline,with vertical messages exchanged between states.Timing diagrams represent timing data for individual classifiers and interactions of classifiers.You can use this diagram to provide snapshot of timing data for a system



## 6.Activity:

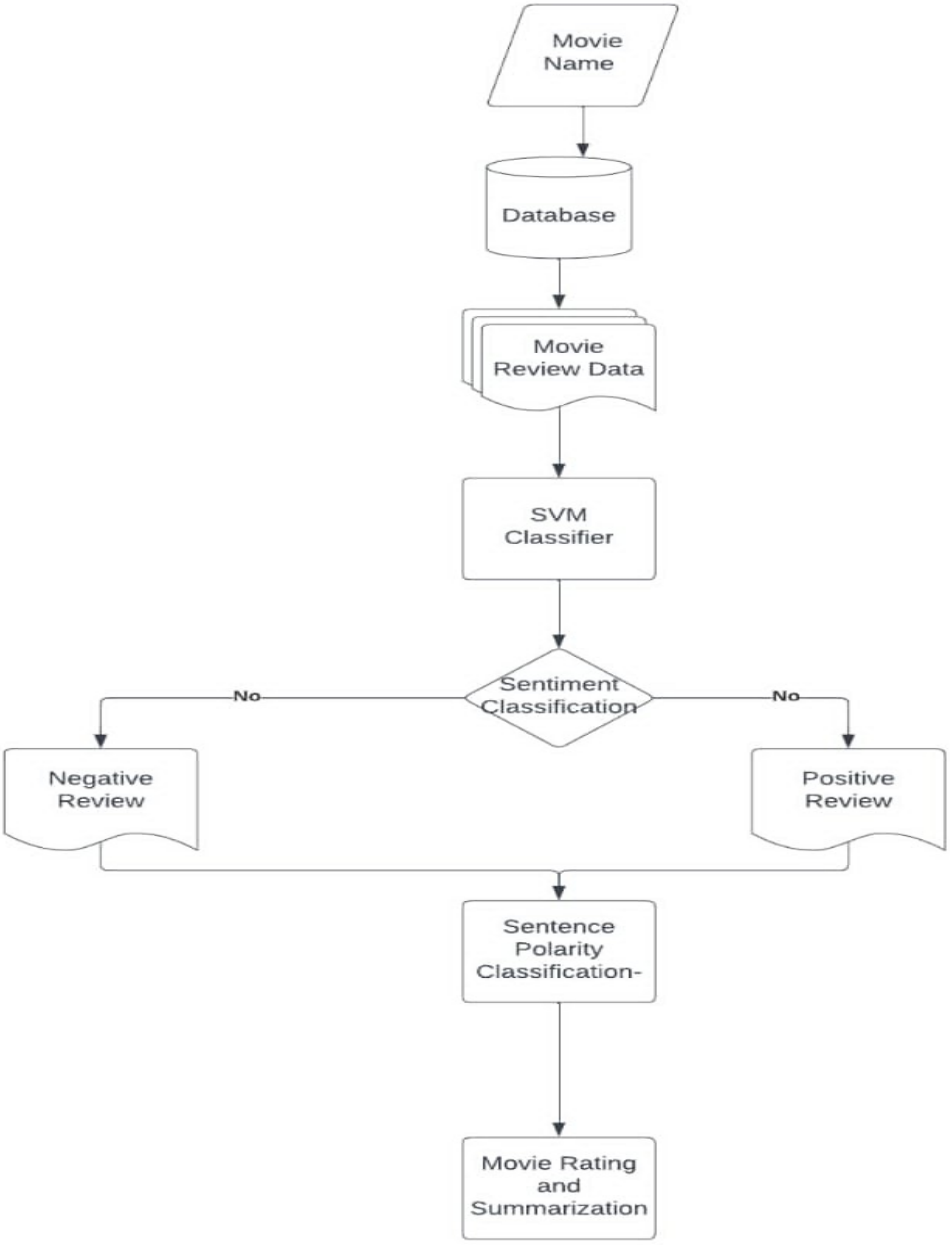
An Activity diagram visually presence a series of action or flow of control in a system similar to a flow chart or a data flow diagram.Activity diagrams are often used in business process modelling.They can also describe the steps in use case diagram.




Activity Diagram

## 7.Communication:

A Communication diagram model is the interaction between objects or parts in terms of sequenced messages.Communication diagrams represent a combination of information taken from class,sequence,and use case diagrams describing the static structure and dynamic of a system.




Communication Diagram

# 5.IMPLEMENTATION

## 5.1 MODULES

- Home page
- Movie Trailer
- Movie related images
- Movie details

## 5.2 MODULES DESCRIPTION

### Home Page:

The website home page consists a glance of entry section about our website.User can enter what the desired movie of him/her.user can enter the movie name on the lets explore bar and after clicking on send button user can receive the information about the desired movie.

### Movie Trailer:

After clicking on the send button ,user get the interface with the movie poster and aside we get the movie trailer.The movie trailer is directly connected to youtube.

### Movie related images:

Every user expect some movie related images to get the recommended movies to watch.So here we are ataaching some movie related images open in separate interface.

### Movie details:

In this movie details section user get the output like cast,crew,rating,reviews,production& management ,music,direction,and budget of the movie .We get remaining all the details related each and every language movies.

## 5.3 INTRODUCTION OF TECHNOLOGIES USED

### Introduction of Front-end techonologies:

**HyperText Markup Language (HTML)** used to describe the presentation of a documentwritten in HTML or XML (including XML dialects such as SVG, MathML or XHTML). Each individual piece markup code (which would fall between "<" and ">" characters) is referred to as an element, though many people also refer to it as a tag. Some elements come in pairs that indicate when some display effect is to begin and when it is to end.

**Cascading Style Sheets (CSS)** is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS describeshow elements should be rendered on screen, on paper, in speech, or on other media. CSS can define color,font, text alignment, size, borders, spacing, layout and many other typographic characteristics, and can do so independently for on-screen and printed views.

**Javascript (JS)** is used by programmers across the world to create dynamic and interactive web content like applications and browsers. JavaScript is so popular that it's the most used programming language inthe world, used as a client-side programming language by 97.0% of all websites. JavaScript is a scriptinganguage used to create and control dynamic website content, i.e. anything that moves, refreshes, orotherwise changes on your screen without requiring you to manually reload a web page.

### Introduction to backend technologies:

**Python** is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

## 5.4. SAMPLE CODE

## HTML:

**direction.html:**

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>Direction</title>
        <link rel="stylesheet" href="{% static 'myfirst.css' %}">
        <link rel="stylesheet"href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/1.0.0/
cssfontawesome.css"integrity="sha512YX9NLok9RLg+o5JkA2Whp1kPDCBCEUolTyYykJVOe1
xcaTNebXU/x02TxV0LpVYqVNJV5NVjyoMW0jHnJWICKg==" crossorigin="anonymous"
referrerpolicy="no-referrer" />
</head>
<body>
<div class="gallery" style="background:#1D1E22;padding: 50px;">
</div>
</body>
</html>
```

**home.html:**

```
\
% load static %}
<!DOCTYPE html>
<html>
<head>
        {<meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>{{newdata.name}}</title>
        <link rel="stylesheet" href="{% static 'myfirst.css' %}">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/1.0.0/css/
fontawesome.css"integrity="sha512YX9NLok9RLg+o5JkA2Whp1kPDCBCEUolTyYykJVOe1xca
TNebXUx02TxV0LpVYqVNJV5NVjyoMW0jHnJWICKg=="crossorigin="anonymous"
referrerpolicy="no-referrer" />
        <linkrel="stylesheet"href="https://fonts.googleapis.com/css2?
family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@20..48,100..700,0..1,-50..200" />
        <linkrel="stylesheet"href="https://fonts.googleapis.com/css2?
family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@48,400,0,0" />
</head>
```

```
<style>

        body{
background:lineargradient(109.6deg,rgb(36,45,57)11.2%,rgb(16,37,60)51.2%,rgb(0,0,0)98.6%);
        }
.material-symbols-outlined {
  font-variation-settings:
  'FILL' 0,
  'GRAD' 0,
  'opsz' 48
}
nav {
        position: relative;
        width:100%;
        }
nav ul {
        text-align: left;
        color: white;
        width: 100%;
        font-size: 18px;
        border: none;
        cursor: pointer;
        list-style: none;
        }
nav ul li ul li {
  padding-left: 50px;
}
nav ul li a {
        font-weight: bold;
        display: block;
        padding: 10px 15px;
        color: white;
        text-decoration: none;
        -webkit-transition: 0.2s linear;
        -moz-transition: 0.2s linear;
        -ms-transition: 0.2s linear;
        transition: 0.2s linear;
}
```

```
nav ul li{
        background-color: grey !important;
        margin-bottom: 10px !important;
}
nav ul li a .fa {
        width: 16px;
        text-align: center;
        margin-right: 5px;
        float:right;
}
nav ul ul {
        background-color:;
}
.ps15{
        background-color: grey !important;
        margin-bottom: 10px;
}
</style>
<body>
<div class="navbar">
        <div class="navhead">
                <h3>Movie Exploration</h3>
        </div>
</div>
<div class="ps0" style="color: white;">
        <div class="ps1">
                <div class="ps11">
                        <h1 id="search">{{newdata.name}}</h1>
                <div class="ps12">
                        <div class="ps121">
                                <img src={{newdata.image}}>
                        </div>
                        <div class="ps1212" id="videos">
                        </div>
                </div>
                <div class="ps13">
                        <p>Genre : </p>
                        {% for i in newdata.genre %}{% endfor %}
```

```
                        {% endfor %}
                    </ul>
                </li>
                <li class='sub-menu'><a href='#message'>Music<div class='fa fa-caret-down right'></div></a>
                    <ul>
                        {% for key in newdata.music %}
                            {% for i in key %}
                                <li>{{ i }}</li>
                            {% endfor %}
                        {% endfor %}
                    </ul>
                </li>
            </ul>
            </nav>
        </div>
</div>
</body>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.1/jquery.min.js"></script>
<script>
    function loadFunction(){
        var API_KEY = "AIzaSyCUWkB4D7laAKpWDce2Wyy10f2u_2gOQEg"
        let search = document.getElementById("search").innerText
        var video = ''
        search = search + " trailer"
        videoSearch(API_KEY,search,1)
        function videoSearch(key,search,maxres){
            $.get("https://www.googleapis.com/youtube/v3/search?key="+key+"&type=video&part=snippet&maxResults="+maxres+"&q="+search,function(data){
                data.items.forEach(item => {
                    video = `<iframe width="420px" height="315px" src="http://youtube.com/embed/${item.id.videoId}" frameborder="0" allowfullscreen></iframe>`

                    $("#videos").append(video)

                });
            })
        }
    }
    window.onload = loadFunction();
```

```
</script>
<script>
	$('.sub-menu ul').hide();
$(".sub-menu a").click(function () {
	$(this).parent(".sub-menu").children("ul").slideToggle("100");
	$(this).find(".right").toggleClass("fa-caret-up fa-caret-down");
});
</script>
</html>
```

**index.html:**

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
	<meta charset="utf-8">
	<meta name="viewport" content="width=device-width, initial-scale=1">
	<title>Movie Expo</title>
	<link rel="stylesheet" href="{% static 'myfirst.css' %}">
</head>
<body>
<div class="wrapper">
	<div class="navbar">
		<div class="navhead">
			<h3>Movie Exploration</h3>
		</div>
	</div>
	<div class="home">
		<img src="{% static 'home.jpg' %}"/>
		<div class="title">
			<h1>Movie</h1>
			<h2>Exploration !!</h2>
			<form class="quick-subscribe" action="{% url 'home' %}" method="POST">
				{% csrf_token %}
			  <div class="subscribe-field">
			    <label for="subscribe-input">Enter Movie Name</label>
			    <input type="text" name="title" id="subscribe-input" required />
```

```
                        </div>
                        <div class="subscribe-sent">Please wait..!</div>
                        <button type="submit" class="subscribe-submit">Send</button>
                      </form>
                </div>
        </div>
</div>
<script>
        /**
 * Quick Subscribe Class
 */
class QuickSubscribe {
  constructor(element) {
    this._element = element;
    this._button = element.querySelector('.subscribe-button');
    this._field = element.querySelector('.subscribe-field');
    this._input = element.querySelector('#subscribe-input');
    this._bindEventListeners();
  }

  _bindEventListeners() {
    this._element.addEventListener('submit', (e) => {
      this._submit();
    });

    this._button.addEventListener('click', (e) => {
      e.preventDefault();
      this._showField();
    this._input.addEventListener('focus', (e) => {
      this._inputFocused(true);
    });

    this._input.addEventListener('blur', (e) => {
      this._inputFocused(false);
    });

    this._input.addEventListener('input', (e) => {
      this._inputHandleInput();
```

```
    });
  }

  _showField() {
    this._element.classList.add('show-field');
  }

  _hideField() {
    this._element.classList.remove('show-field');
  }

  _submit() {
    this._element.classList.add('submitted');
    this._hideField();
  }

  _inputFocused(focused = true) {
    this._field.classList.toggle('focused', focused);
  }
```

**music.html:**

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
	<meta charset="utf-8">
	<meta name="viewport" content="width=device-width, initial-scale=1">
	<title>Music</title>
	<link rel="stylesheet" href="{% static 'myfirst.css' %}">
	<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/1.0.0/css/font-awesome.css" integrity="sha512-YX9NLok9RLg+o5JkA2Whp1kPDCBCEUolTyYykJVOe1xcaTNebXU/x02TxV0LpVYqVNJV5NVjyoMW0jHnJWICKg==" crossorigin="anonymous" referrerpolicy="no-referrer" />
</head>
<body>
<div class="gallery" style="background:#1D1E22;padding: 50px;">
</div>
</body>
</html>
```

**photos.html:**

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
	<meta charset="utf-8">
	<meta name="viewport" content="width=device-width, initial-scale=1">
	<title>Photos</title>
	<link rel="stylesheet" href="{% static 'myfirst.css' %}">
	<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/1.0.0/css/fontawesome.css"integrity="sha512YX9NLok9RLg+o5JkA2Whp1kPDCBCEUolTyYykJVOe1xcaTNebXUx02TxV0LpVYqVNJV5NVjyoMW0jHnJWICKg=="crossorigin="anonymous"referrerpolicy="no-referrer" />
	<style type="text/css">
	.gallery{
	}
	.gal{
	  display: flex;
	  flex-wrap: wrap;
	}

	.gallery img{
	  max-width: 1000px;
	  max-height: 500px;
	  border-radius: 20px;
	  margin: 5px;
	}
	</style>
</head>
<body>
<div class="gallery" style="background:#1D1E22;padding: 50px;">
	<div class="gal">
	{%  for i in piclist %}
		<img src={{i}}>
	{% endfor %}
	</div>
</div>
</body>
</html>
```

```
<!--
<img
    src="https://mdbcdn.b-cdn.net/img/Photos/Horizontal/Nature/4-col/img%20(18).webp"
    class="w-100 shadow-1-strong rounded mb-4"
    alt="Waves at Sea"
  /> -->
```

**production.html:**

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <link rel="stylesheet" href="{% static 'myfirst.css' %}">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/1.0.0/css/fontawesome.css"integrity="sha512YX9NLok9RLg+o5JkA2Whp1kPDCBCEUolTyYykJVOe1xcaTNebXUx02TxV0LpVYqVNJV5NVjyoMW0jHnJWICKg=="crossorigin="anonymous"referrerpolicy="no-referrer" />
</head>
<body>
<div class="gallery" style="background:#1D1E22;padding: 50px;">
</div>
</body>
</html>
```

**writing.html:**

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>Writing</title>
        <link rel="stylesheet" href="{% static 'myfirst.css' %}">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/1.0.0/css/fontawesome.css"integrity="sha512YX9NLok9RLg+o5JkA2Whp1kPDCBCEUolTyYykJVOe1xcaTNebXUx02TxV0LpVYqVNJV5NVjyoMW0jHnJWICKg=="crossorigin="anonymous"referrerpolicy="no-referrer" />
</head>
<body>
<div class="gallery" style="background:#1D1E22;padding: 50px;">
```

```
</div>
</body>
</html>
```

**CSS:**

```
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@300&family=Open+Sans:wght@300;400;500;700&family=Poppins:wght@200&display=swap');
*{
  margin: 0;
 padding: 0;
}
::-webkit-scrollbar {
  width: 10px;
}

/* Track */
::-webkit-scrollbar-track {
  background: #f1f1f1;
}

margin-right: 70px;
  display: inline-flex;
}
.navlinks a{
  color: white;
  text-align: center;
  font-size: 20px;
  text-decoration: none;
  padding: 15px 20px;
}
.navlinks a:hover{
  background-color: white;
  color: black;
}
.title{
  position: absolute;
  top: 40vh;
}
```

```
.home h1{
}
.home h1,h2{
 margin-left: 100px;
 margin-bottom: 0;
 color: white;
font-
family:"Alliance1","Alliance1HeaderFallback",applesystem,BlinkMacSystemFont,"Segoe
UI",Helvetica,Arial,sans-serif,"Apple Color Emoji","Segoe UI Emoji","Segoe UI Symbol";
  font-feature-settings: unset;
  display: block;
  font-weight: bold;
}
.home h2{
  margin-top: -40px;
}
 .quick-subscribe {
  margin-top: 20px;
  margin-left: 100px;
   position: relative;
   background: #fff;
   border-radius: 2rem;
   box-shadow: 0 1rem 1rem rgba(124, 126, 130, .05);
   transition: max-width 0.3s ease;
   width: 100%;
   max-width: 13.75rem;
}
 .quick-subscribe #subscribe-input, .quick-subscribe .subscribe-sent, .quick-subscribe button {
font-family: 'Gothic A1';
 padding: 0.7rem 2rem 1rem;
   border: 0;
}
.quick-subscribe #subscribe-input:focus, .quick-subscribe .subscribe-sent:focus, .quick-subscribe
button:focus {
   outline: 0;
}
.quick-subscribe input{
  font-size: 22px;
}
```

```
.quick-subscribe button {
 font-size: 22px;
  font-weight: 700;
  cursor: pointer;
}
.quick-subscribe .subscribe-field {
  visibility: hidden;
  opacity: 0;
  transition: all 0.3s ease;
  position: relative;
}
.quick-subscribe .subscribe-field #subscribe-input {
  width: 100%;
  background: none;
}
.quick-subscribe .subscribe-field label {
  font-family: 'Gothic A1';
  color: #bdbec0;
  position: absolute;
  left: 2rem;
  top: 1rem;
  visibility: hidden;
  opacity: 0;
  transition: all 0.3s 0.3s ease;
  transform: scale(0);
  cursor: text;
}
.quick-subscribe .subscribe-button, .quick-subscribe .subscribe-sent {
  background: none;
  color: #7c7e82;
  display: inline-block;
  position: absolute;
  top: 0;
  bottom: 0;
  left: 0;
  margin: 0 auto;
  text-align: center;
  transition: all 0.3s ease;
```

```
}
 .quick-subscribe .subscribe-submit {
   background: #7c7e82;
   color: #fff;
   padding-top: 0.5rem;
   padding-bottom: 0.5rem;
   position: absolute;
   border-radius: 2rem;
   top: 0.3rem;
   right: 0.5rem;
   z-index: 5;
   visibility: hidden;
   opacity: 0;
   transition: all 0.3s 0.2s ease;
   transform: scale(0);
}
 .quick-subscribe .subscribe-sent {
  font-size: 22px;
   font-weight: 700;
   visibility: hidden;
   text-align: left;
   opacity: 0;
   transition: all 0.3s 0.2s ease;
 .quick-subscribe.show-field {
   max-width: 27.5rem;
}
 .quick-subscribe.show-field .subscribe-field {
   opacity: 1;
   visibility: visible;
}
 .quick-subscribe.show-field .subscribe-field label {
   opacity: 1;
   visibility: visible;
   transform: scale(1);
   transition-delay: 0s;
}
 .quick-subscribe.show-field .subscribe-field.has-value label {
   opacity: 0;
```

```
   visibility: hidden;
}
 .quick-subscribe.show-field .subscribe-button {
   visibility: hidden;
   opacity: 0;
}
 .quick-subscribe.show-field .subscribe-submit {
   visibility: visible;
   opacity: 1;
   transform: scale(1);
}
 .quick-subscribe.submitted .subscribe-button {
   display: none;
}
   transition-delay: 0s;
}
 .quick-subscribe.submitted .subscribe-sent {
   visibility: visible;
   opacity: 1;
}
.ps0{
  padding: 40px;
  background: ;
}
.ps11 h1{
  font-size: 50px;
}
.ps12{
  display: flex;
}
.ps121{
  width: 310px;
  height: 480px;
  }
.ps1212{
  padding-left: 10px;
  width: 100%;
}
```

```
.ps13,.ps133{
  font-weight: bold;
}
.ps15 button{
    background-color: grey !important;
  text-align: left;
  font-weight: bold;
  width: 100%;
  padding: 16px;
  font-size: 16px;
  border: none;
  cursor: pointer;
}
```

**PYTHON (django modules)**

**asgi.py**

```
"""
ASGI config for se project.
It exposes the ASGI callable as a module-level variable named ``application``.
For more information on this file, see
https://docs.djangoproject.com/en/dev/howto/deployment/asgi/
"""
import os
from django.core.asgi import get_asgi_application
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'se.settings')
application = get_asgi_application()
```

**settings.py**

```
"""
Django settings for se project.
Generated by 'django-admin startproject' using Django 3.2a1.
For more information on this file, see
https://docs.djangoproject.com/en/dev/topics/settings/
For the full list of settings and their values, see
https://docs.djangoproject.com/en/dev/ref/settings/
"""
from pathlib import Path
import os
```

```
# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent
# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/dev/howto/deployment/checklist/
# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-uzwcr1(9034o8ho0ecattk7-0z56l$k=m5f=$=nf8(=w9wupsb'
# SECURITY WARNING: don't run with debug turned on in production!
ALLOWED_HOSTS = []
# Application definition
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]
MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]
ROOT_URLCONF = 'se.urls'
TEMPLATES = [
    {
        'DIRS': [os.path.join(BASE_DIR,'templates')],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
```

```
        },
    },
]
WSGI_APPLICATION = 'se.wsgi.application'
# Database
# https://docs.djangoproject.com/en/dev/ref/settings/#databases
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}
# Password validation
# https://docs.djangoproject.com/en/dev/ref/settings/#auth-password-validators
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]
# Internationalization
# https://docs.djangoproject.com/en/dev/topics/i18n/
LANGUAGE_CODE = 'en-us'
TIME_ZONE = 'UTC'
USE_I18N = True
USE_L10N = True
USE_TZ = True
# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/dev/howto/static-files/
STATIC_URL = 'static/'
STATICFILES_DIRS = [
```

```
    BASE_DIR / "static",
]
# Default primary key field type
# https://docs.djangoproject.com/en/dev/ref/settings/#default-auto-field
DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'
```

**urls.py**

```
"""se URL Configuration
The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/dev/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.urls import path
from .views import movies,home,photos
urlpatterns = [
    path('',movies,name="movies"),
    path('home/',home,name="home"),
    path('photos/',photos,name="photos"),
]
```

**views.py**

```
from django.shortcuts import render,redirect
from django.http import HttpResponse,HttpResponseRedirect
import requests
from bs4 import BeautifulSoup
import json
from concurrent.futures import ThreadPoolExecutor
def getrequest(link):
   r = requests.get(
    url='https://proxy.scrapeops.io/v1/',
    params={
```

```
    ["summary"] = data['summary_text']
    newdata["topcast"] = topcast(soup3)
    newdata["writing"] = writingcredits(production)
    newdata["direction"] = directioncredits(production)
    newdata["production"] = productioncredits(production)
    newdata["music"] = musiccredits(production)
    newdata["reviews"] = reviews(imdbid)
    return render(request,'home.html',{'newdata':newdata})
def movies(request):
        return render(request,'index.html')
def getMovieDetails(imdbID):
    data = {}
    movie_url = "https://www.imdb.com/title/"+imdbID
    r = getrequest(movie_url)
    soup = BeautifulSoup(r.text, 'html.parser')
    jsonData = soup.find('script',{"type":"application/ld+json"})
    Moredata=[]
    jsonSourceObj=json.loads(jsonData.string)
    Moredata.append(jsonSourceObj)
    data["expanded"]=Moredata
    data["imdbID"] = imdbID
    title = soup.find('title')
    data["title"] = title.string
    data["Minutes"]=""
    try :
      data["Minutes"]=jsonSourceObj['duration']
    except :
      data["Minutes"]=""
    data["ratingValue"]=""
    try :
       data["ratingValue"]= jsonSourceObj['aggregateRating']['ratingValue']
    except :
      data["ratingValue"]=""
      data["ratingCount"] =""
    try :
       data["summary_text"]= jsonSourceObj['description']
    except :
      data["summary_text"]=""
```

```
            data['keywords']=jsonSourceObj['keywords']
    except :
        data['keywords']=""
    return data
def photosfun(link):
    imdb = "https://www.imdb.com"
    link2 = imdb+link['href']
    obj = getrequest(link2)
    soup = BeautifulSoup(obj.content, 'html.parser')
    try:
        di = soup.find('img',class_="sc-7c0a9e7c-0 hXPlvk")
    except:
        di = ""
    try:
        di = soup.find('img',class_="sc-7c0a9e7c-1 kJatiV")
    except:
        di = ""
    if di != None:
      return di['src']
    else:
      return ""
def photos(request):
    photos = request.GET.get('photoslink')
    piclist = []
    photopage = getrequest(photos)
    photosoup = BeautifulSoup(photopage.content, 'html.parser')
    picdiv = photosoup.find('div',class_="media_index_thumb_list").find_all('a')
    # newpicdiv = [picdiv[i] for i in range(10)]
    with ThreadPoolExecutor(max_workers=10) as executor:
            return render(request,'photos.html',{'piclist':piclist})
def reviews(ImdbId,sort="submissionDate",ratingFilter=10,dir="asc"):
  try :
  soup = BeautifulSoup(r.text, 'html.parser')
  try :
    reviews = soup.find_all('div',{'class' : 'imdb-user-review'})
  except :
    pass
  reviews_text =[]
```

```
  for review in reviews :
    review_imdb={}
  for i in oddtr:
      if(i.find('td',class_='character') != None):
          j = i.find('td',class_='character').find('a')
      if(j != None):
          oddchar.append(j.text)
      else:
          oddchar.append("None")
  bothchar = []
      bothchar.append(oddchar[i])
      bothchar.append(evenchar[i])
  oddnames = []
  evennames = []
  bothnames = []
  for i in oddtr:
      l = i.find_all('a')
      if(l):
          oddnames.append(l[1].text.strip())
  for i in eventr:
      l = i.find_all('a')
      if(l):
          evennames.append(l[1].text.strip())
  for i in range(len(evennames)):
      bothnames.append(oddnames[i])
      bothnames.append(evennames[i])
  return dict(zip(bothnames,bothchar))
def writingcredits(production):
    wri1 = production[1].find_all('a')
    wri2 = production[1].find_all('td',class_="credit")
    l1 = [i.text.strip() for i in wri1]
    l2 = [i.text.strip() for i in wri2]
    return l1,l2
def productioncredits(production):
    pro1 = production[2].find_all('a')
    pro2 = production[2].find_all('td',class_="credit")
    l1 = [i.text.strip() for i in pro1]
    l2 = [i.text.strip() for i in pro2]
```

```
    return dict(zip(l1,l2))
    application = get_wsgi_application()
PYTHON
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys
def main():
    """Run administrative tasks."""
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'se.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)
if __name__ == '__main__':
    main()
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys
def main():
    """Run administrative tasks."""
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'se.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)
```

# 6.TEST CASES

## Home Page

All the html elements are successfully redirecting to the particular interface or not.The Nav bar and Bootstrap components must be collectively paired with their respective objectives.

## Movie Trailer

Where all the movie related trailers are directly connected to the youtube source.Search that you have to get trailer as per the searched movie.It directly redirects to the youtube and we get the trailer as the user searched.

## Movie Images

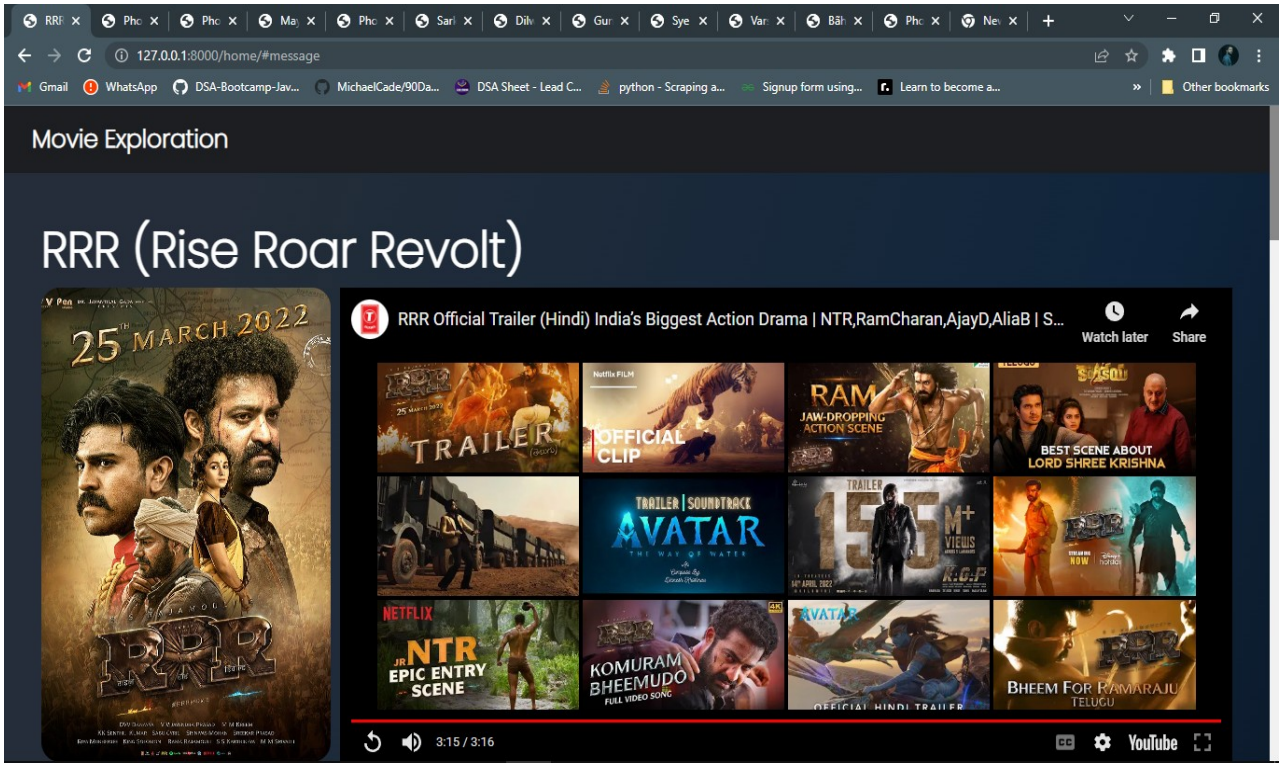
Where as the movie images are the mandatory to the movie related source.For the related movie searches we have to get the images exactly as per the searched image.User will be satisfied with the searched input.

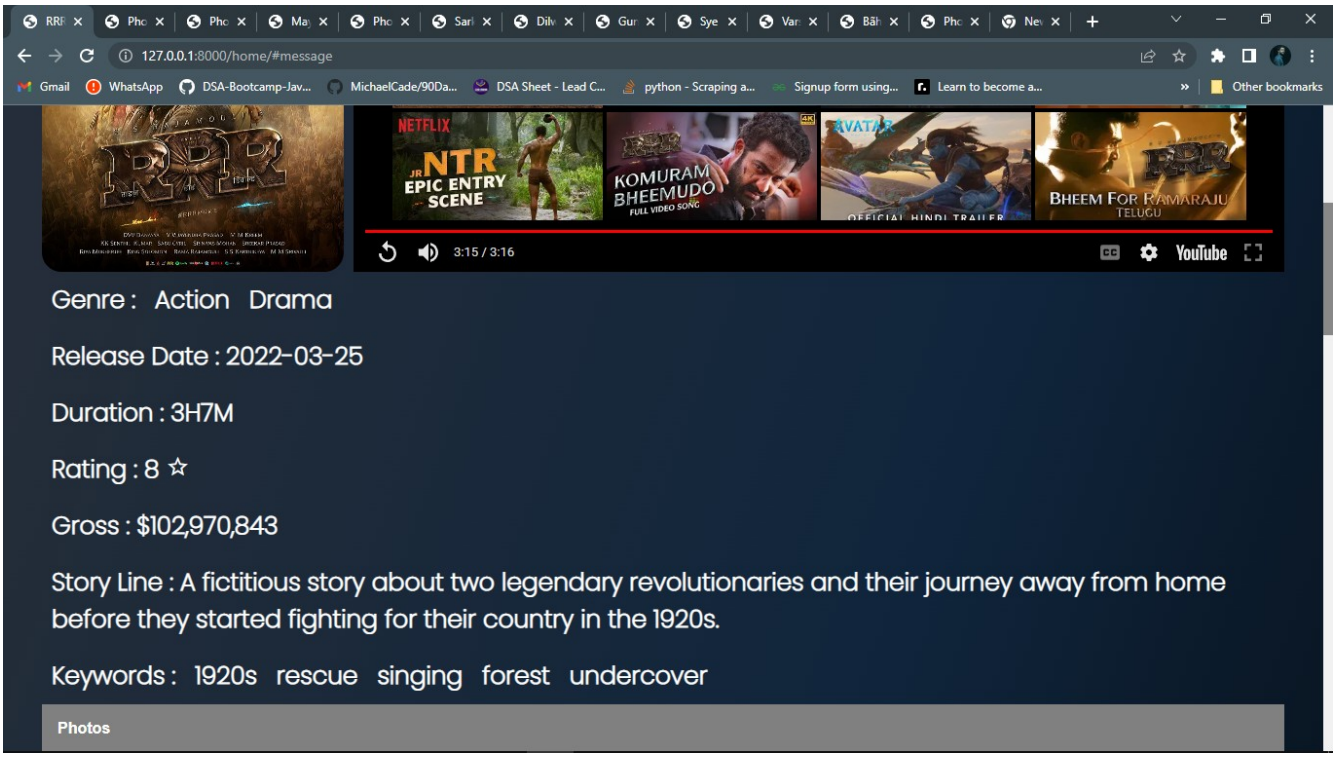
## Movie Details

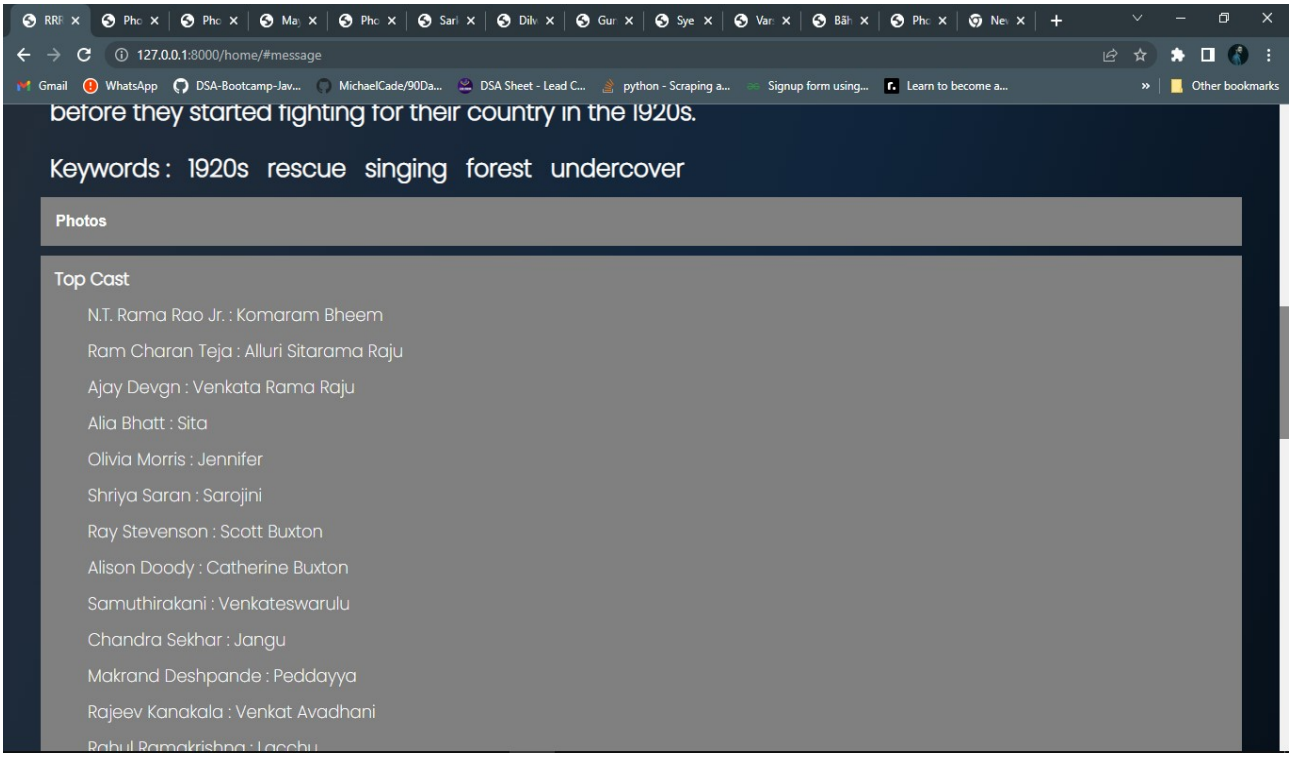
Movie details means user get the output like top cast crew rating reviews budget management production music and all the awards which are awarded to that movie.All the reviews are written by critics and reviewrs.

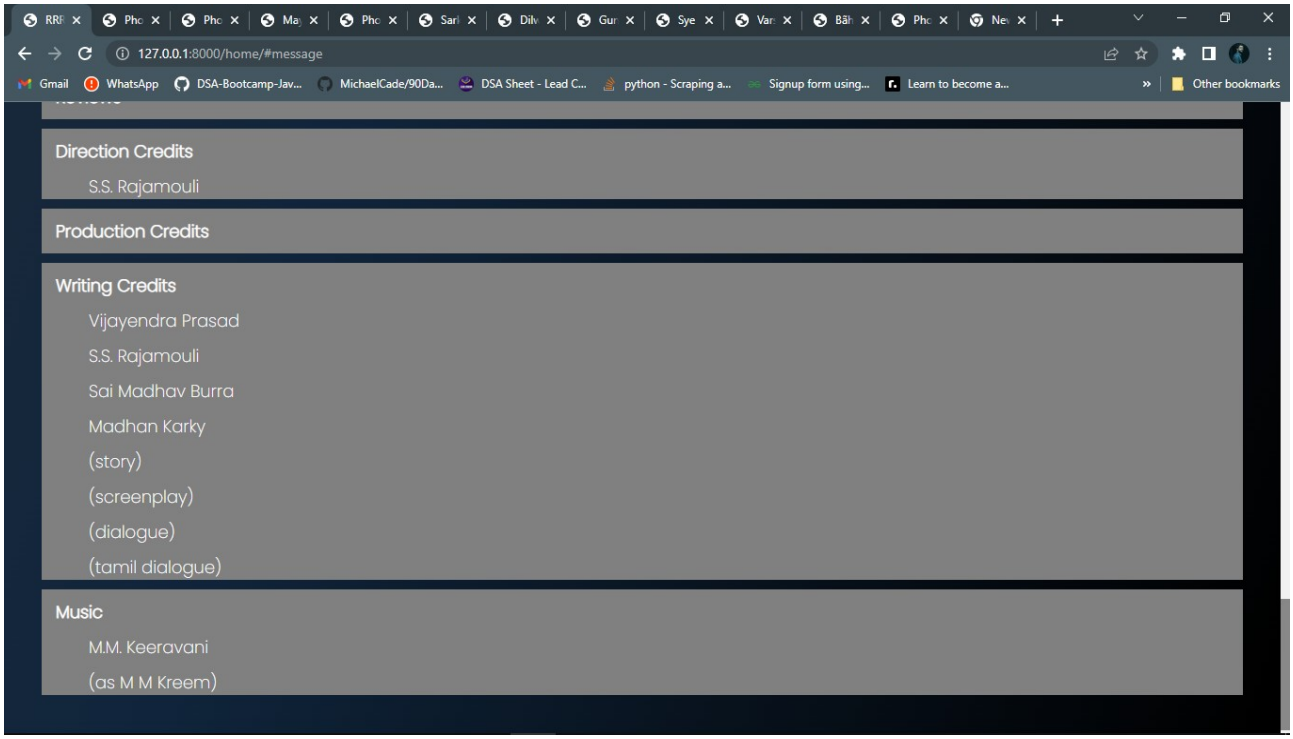
# 7.SCREENSHOTS

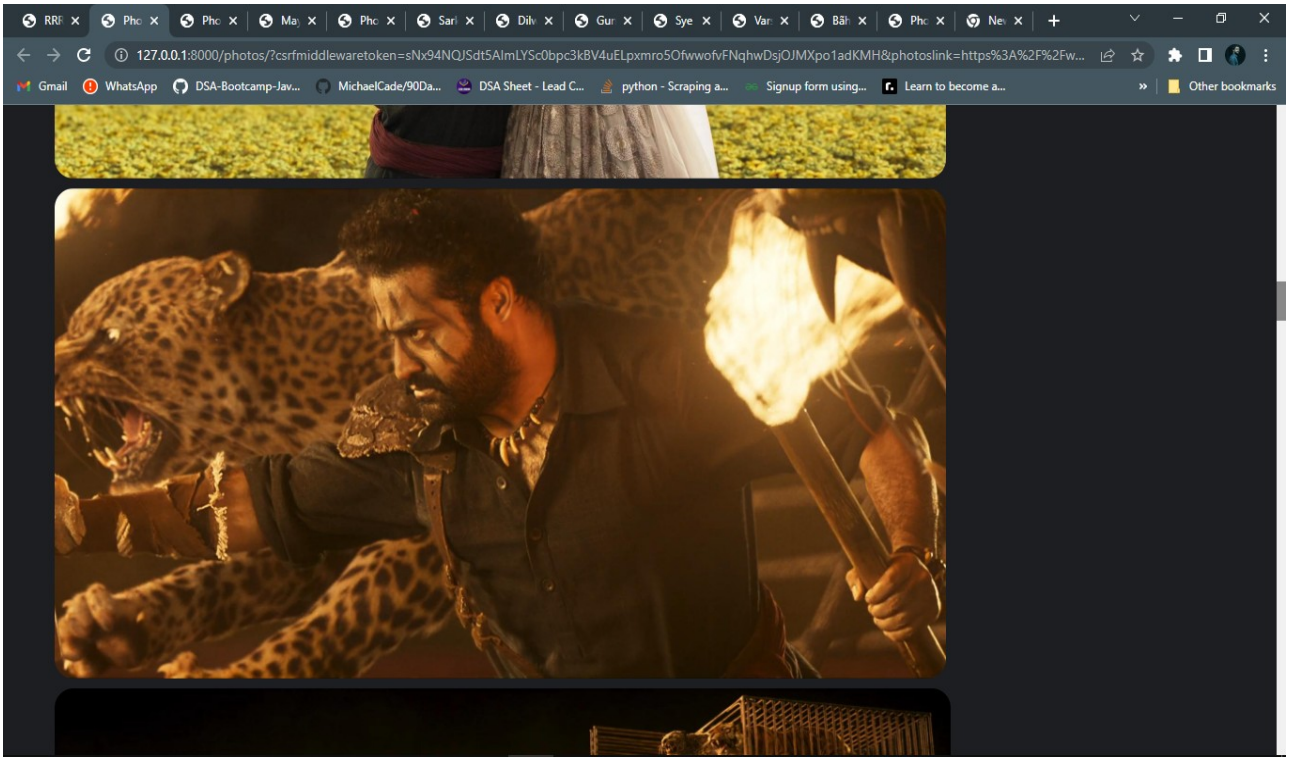
**Home page:**




127.0.0.1:8000/home/#message
Gmail
WhatsApp
DSA-Bootcamp-Jav...
MichaelCade/90Da...
DSA Sheet - Lead C...
python - Scraping a...
Signup form using...
Learn to become a...
Other bookmarks
NETFLIX
JR NTR EPIC ENTRY SCENE
KOMURAM BHEEMUDO FULL VIDEO SONG
AVATAR
BHEEM FOR RAMARAJU TELUGU
3:15 / 3:16
YouTube
Genre : Action Drama
Release Date : 2022-03-25
Duration : 3H7M
Rating : 8 ☆
Gross : $102,970,843
Story Line : A fictitious story about two legendary revolutionaries and their journey away from home before they started fighting for their country in the 1920s.
Keywords : 1920s rescue singing forest undercover
Photos


127.0.0.1:8000/home/#message
before they started fighting for their country in the 1920s.
Keywords : 1920s rescue singing forest undercover
Photos
Top Cast
N.T. Rama Rao Jr. : Komaram Bheem
Ram Charan Teja : Alluri Sitarama Raju
Ajay Devgn : Venkata Rama Raju
Alia Bhatt : Sita
Olivia Morris : Jennifer
Shriya Saran : Sarojini
Ray Stevenson : Scott Buxton
Alison Doody : Catherine Buxton
Samuthirakani : Venkateswarulu
Chandra Sekhar : Jangu
Makrand Deshpande : Peddayya
Rajeev Kanakala : Venkat Avadhani


127.0.0.1:8000/home/#message
Direction Credits
S.S. Rajamouli
Production Credits
Writing Credits
Vijayendra Prasad
S.S. Rajamouli
Sai Madhav Burra
Madhan Karky
(story)
(screenplay)
(dialogue)
(tamil dialogue)
Music
M.M. Keeravani
(as M M Kreem)


127.0.0.1:8000/photos/?csrfmiddlewaretoken=sNx94NQJSdt5AlmLYSc0bpc3kBV4uELpxmro5OfwwofvFNqhwDsjOJMXpo1adKMH&photoslink=https%3A%2F%2Fw...

## 8.CONCLUSION

Here is our project **MOVIE EXPLORATION** using python can be accessed by each every person.Here user can get the all the deatils of the movies at one place.This application is like the single stand point for the movies history.Here we provide genre,release date,duration,rating,reviews,topcast,crew,direction,production,management,music,awards,trailer,and some images.

User can get any language related movies like tollywood, hollywood, bollywood, sandalwood, kollywood etc.Genuine ratings and reviews are given.IMDB rating also mentioned .For every user who needs a quality information about their movie which they are going to wished will be displayed.Moreover any regional people can use this application because we get details about the each and every language movie.

This is the best one stop reliable application for movie exploraton.We can explore any type of movie using this sofyaware application.

## 9.FUTURE ENHANCEMENT

Movies have always been a primary source of entertainment for people, and this needs to be noted by all from the very beginning in this case. As means of entertainment, it can be found that movies have been there for us for most of the last century. But the growth has not been seen to be static in this case. There have been exponential developments in this case and have to be noted by all with due diligence. These developments need to be understood because they form a seminal part of the history of films at large. Traditionally, films mainly were seen from the big theatres, but that model gradually evolved, and streaming platforms came into existence. These completely changed how the movie industry works and need to be remembered by all with due efforts. There are distinct perspectives to be seen in this case, and in this application, we shall focus on analyzing and providing .

More amount of content in the future will be customized. The dependence will also be on regional varieties and tastes. It can be seen that due to niche areas of technology like big data, several desires of viewers can be kept track of easily, and this needs to be remembered by all. As the desires can be kept track of, the streaming sites can customize the movies' content accordingly, which is one of the major trends to expect in the short term and has to be noted with due diligence in this case.

There have been many genres of movies in which the streaming platforms have not yet ventured. But that will soon be replaced. More people will now view niche content. This will be one of the most trending aspects to note in the case of future streaming.

## 10.BIBILOGRAPHY

Dir. Linda Bonito. Paramount, 2002. Film." List performer names after the director’s name if you would like to do so. You might write "Fake Movie. Dir. Linda Bonito. Perf. Julia Pacman, Sharon Loinclout, Bugs Appleman. Paramount, 2002. Film."

An overview of different approaches to recommendations

Kunal Shah, Akshaykumar Salunke, Saurabh Dongare, Kisandas Antala SIT, Lonavala India 2017.The first instance of film censorship actually dates back to 1897. It was then that enoch j rector filmed a boxing match between James Corbett and Bob Fitzsimmons in Carson City, Nevada. At the time, boxing was illegal throughout the rest of the United States

In the 1980s, many blockbuster films featured violence and gore – a majority of these films belonged to steven spekberg. His films still fell under the guidelines of PG, as they did not push the boundaries into an R rating. After a public outcry.

In 2020 the movies are given rating but without having any perfect sorce without any perfect references

In 2021 movies given every particulars but trailer will be opened in different source which we have to open in different application.

In 2022 we are including movie details along with the trailer and images including every paticular of the movie.

## REFERENCES

- https://scrapeops.io/
- https://stackoverflow.com
- https://youtube.com
- https://m.imdb.com/
- https://www.coding/

## THANK YOU