**Movie Recommendation System**

###### A Project Report

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

#### BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

###### By

**ANKIT KALAUNI (23) & ADITYA KHANDELWAL (24)**



###### DEPARTMENT OF INFORMATION TECHNOLOGY

**KISHINCHAND CHELLARAM COLLEGE**

***(Affiliated to University of Mumbai)* MUMBAI, 400020 MAHARASHTRA**

###### 2021-2022

**PROFORMA FOR THE APPROVAL PROJECT PROPOSAL**

PNR No.: Roll No: 23 &24

1. **Name of the Student: ANKIT KALAUNI & ADITYA KHANDELWAL**
2. **Title of the Project:** MOVIE RECOMMENDATION SYSTEM
3. **Name of the Guide:** MR. LOKESH IYER 

#### Teaching experience of the Guide: 7 Years

1. **Is this your first submission?** Yes No

✓

##### Signature of the Student Signature of the Guide

Date: ………………… Date: ……………….

Signature of the Coordinator

Date: ………………

**KISHINCHAND CHELLARAM COLLEGE**

***(Affiliated to University of Mumbai)***

**MUMBAI-MAHARASHTRA-400020**

#### DEPARTMENT OF INFORMATION TECHNOLOGY



**CERTIFICATE**

This is to certify that the project entitled, " **MOVIE RECOMMENDATION SYSTEM** ", is Bonafede work of ANKIT KALAUNI & ADITYA KHANDELWAL bearing

Seat.No:23 & 24 submitted in partial fulfilment of the requirements for the award of the degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

###### Internal Guide Coordinator

External Guide

Date Stamp

**DECLARATION**

I hereby declare that the project entitled, “**Movie Recommendation System**” done at **Kishinchand Chellaram College**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF SCIENCE (the Evaluate INFORMATION TECHNOLOGY) t**o be submitted as a final semester project as part of our curriculum.

**Signature Name:**

**ANKIT**

**KALAUNI**

**Signature**

**Name: Aditya Khandelwal**

**ABSTRACT**

**ABSTRACT**

In this fast and complex world, several alternatives of products are available for a given product. Think of the examples above: streaming videos, social networking, online shopping. the list goes on. Recommendation systems help to personalize a platform and help the user find something they like.

Machine learning algorithms in recommender systems typically fit into two categories: content-based systems and collaborative filtering systems. Modern recommender systems combine both approaches. A content-Based Recommendation system is used to suggest “relevant” items to users. They predict future behavior based on past data. Content-based methods are based on the similarity of movie attributes. Using this type of recommender system, if a user watches one movie, similar movies are recommended.

From a business standpoint, the more relevant products a user finds on the platform, the higher their engagement. This often results in increased revenue for the platform itself. Various sources say that as much as 35–40% of a tech giant’s revenue comes from recommendations alone.

**ACKNOWLEDGEMENT**

**ACKNOWLEDGEMENT**

The success and outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along with the completion of my project. All that we have done is only due to such supervision and assistance and we would not forget to thank them. Special gratitude to our Principal Dr HEMLATA BAGLA for giving us this golden opportunity of doing this project. We respect and thank our HOD Mrs. RAKHI GUPTA and Mrs. Varkha Jewani for providing us with an opportunity to do the project under them and giving us all the support, guidance and also showing keen interest in our project which made us complete the project duly, We also thank them for guiding us all along till the completion of our project by providing all the necessary information for developing the system.

We would like to express our gratitude towards our parents and teachers for their kind co-operation and encouragement which help us in the completion of this project. We also thank the non-teaching staff of the college for their support. Lastly, we give ourselves credit and acknowledgement for the successful completion of the project.

#### Signature

**Name: Aditya Khandelwal**

**Signature**

**Name: Ankit Kalauni**

**TABLE OF CONTENTS**

**TABLE OF CONTENTS**

[**CHAPTER 1: INTRODUCTION 13-16**](#_gjdgxs)

* 1. [**Background 14**](#_30j0zll)
  2. [**Objectives 14**](#_1fob9te)
  3. [**Purpose, Scope and Applicability 15**](#_3znysh7)
     1. [Purpose 15](#_2et92p0)
     2. [Scope 15](#_44sinio)
     3. [Applicability 15](#_tyjcwt)
  4. [**Achievements 16**](#_2jxsxqh)

[**CHAPTER 2: SURVEY OF TECHNOLOGIES 17-21**](#_3dy6vkm)

* 1. [**Technological Adaptations 18**](#_1t3h5sf)
     1. [Hardware 18](#_4d34og8)
     2. [Software 19](#_2s8eyo1)
  2. **Comparative Study 21**

[**CHAPTER 3: REQUIREMENTS AND ANALYSIS 22-28**](#_17dp8vu)

* 1. **Problem Definition 23**
  2. **Requirement Specification 24**
  3. [**Planning and Scheduling 24**](#_3rdcrjn)
  4. [**Software and Hardware Requirements 25**](#_26in1rg)
     1. Hardware Requirements 27
     2. Software Requirements 27
  5. [**Preliminary Product Description 28**](#_35nkun2)
  6. [**Conceptual Models 28**](#_1ksv4uv)

**CHAPTER 4: SYSTEM DESIGN 29-42**

* 1. **Basic Modules 30**
  2. **Block Diagram 31**

###### Logic Diagrams 32

* + 1. Data Flow Diagram 32
    2. Activity Diagram 34
    3. Use Case Diagram

**TABLE OF FIGURES**

|  |  |
| --- | --- |
| **Figure No** | **Name** |
| 3.6 | Gantt Chart |
| 4.1.1. | Data Flow Diagram |
| 4.1.2. | Activity Diagram |
| 4.1.3. | Use case Diagram |
| 4.1.4. | Sequence Diagram |

**CHAPTER 1**

## CHAPTER 1 INTRODUCTION

#### BACKGROUND:

As the World Wide Web continues to grow at an exponential rate, the size and complexity of certain data or web site grow along with it. For the users of these websites, it becomes increasingly difficult and time-consuming to find the information they are looking for. To help users find the information that is following their interests a website can be personalized. Recommender systems can improve a website for individual users by dynamically adding “relevant” information.

When users browse through a website, they are usually looking for items they find interesting. Interest items can consist of several things. For example, movie information can be considered as interest items.

Every large collection needs a certain structure to make it easy for visitors to find what they are looking for. A website can be structured by dividing its web pages into content pages and navigation pages. The content pages provide the user with the interest items while the navigation pages help the user to search for the interest items.

One popular technique of recommendation/recommender systems is content-based filtering. Content here refers to the content or attributes of the products you like. So, the idea in content-based filtering is to tag products using certain keywords, understand what the user likes, look up those keywords in the database and recommend different products with the same attributes.

#### OBJECTIVES:

* Web-based application of content recommendation system, will show the searchedmovie information and recommend relevant movies to get the information about that movie

* Users can search for a movie and get information about the movie’s casts, trivia, descriptions, reviews

* Users can add reviews and upvote or downvote particular movies.

* No hard-coding in the system, simple code structure and pseudo code

* Easy to update and modify the new data into the system using Machine Learning Pipeline

* Deploy recommendation system once and use multiple times, even on unseen new data. i.e., on new movies

###### PURPOSE, SCOPE AND APPLICABILITY:

#### PURPOSE:

* + 1. A recommendation system helps the user to find useful and relevant information from the website which also increase user engagement on the website. It helps users find compelling content in large corpora. For example, the Google Play Store provides millions of apps, while YouTube provides billions of videos. More apps and videos are added every day. How can users find new compelling new content? Yes, one can use search to access the content. However, a recommendation engine can display items that users might not have thought to search for on their own.

#### SCOPE:

The recommendation system overcomes the idea of the hard-coded sites, which makes lots of loads in work and increases the breakdown of the websites as the hard-coded website needed to be updated frequently. On the other side websites which use Recommendation Systems can train the machine learning model once and use it as many times as it wants. it also works if the websites get updated. As if new movies were added to the website database. The developer can make the machine learning

Pipeline to preprocess a new dataset with one click. This will decrease the workload and make the system more efficient and faster.

#### APPLICABILITY:

#### The recommendation system filters out the information automatically for the user, so they do not have to search for it explicitly. Users can get the related article, information with a single click. They can find the information which can’t find directly from the searched term. This will increase the engagement of the user on the website, and users can explore the website and get the needed information, the applicability of this recommendation system is that it overcomes the old traditional website which uses hardcoding. Machine learning recommendation system has a simple data structure and can be easily modified and update, updating this recommendation system make it smarter based on information.

#### ACHIEVEMENTS:

In this project (Content-based recommendation system) we can achieve some benefits, which are listed below.

* Recommend relevant movies related to the searched movies on the navigation page
* Train model once and use it many times
* No hard-coding, the model can predict new movies which are added into the website database
* Easy to update Recommendation system without taking down the server
* Add reviews of the new users on the website and Recommendation system

# CHAPTER 2

### CHAPTER 2 SURVEY OF TECHNOLOGIES

#### TECHNOLOGICAL ADAPTATIONS:

#### HARDWARE:

##### A desktop or Laptop with the following Specifications is used in this project.

* + - * RAM: 4 GB
      * ROM: 1 TB
      * PROCESSOR: Intel Core i3 6th Gen
      * OS: Windows 10

#### SOFTWARE:

**Front End:**

HTML**:**

* It is easy to learn and easy to use.
* It is platform-independent.
* Images, videos, and audio can be added to a web page.
* Hypertext can be added to the text.
* It is a markup language.
* CSS allows users to view documents with their preferred fonts, colors, etc. by specifying them in a user style sheet.

**Feature of JavaScript.**

* Light Weight Scripting language.
* Dynamic Typing.
* Object-oriented programming support.
* Functional Style.
* Platform Independent.
* Prototype-based.
* Interpreted Language.

**Feature of Bootstrap.**

* Fewer Cross-browser bugs
* A consistent framework that supports major of all browsers and CSS compatibility fixes
* Lightweight and customizable
* Responsive structures and styles
* Several JavaScript plugins using the jQuery
* Good documentation and community support
* Loads of free and professional templates, WordPress themes and plugins

* 1. **Technologies used in the backend.**

**We have taken Flask for framework and Pickle to save trained machine learning model**

Features of Flask.

* It is very flexible and easy to learn
* It provides unit testing through its integrated support, built-in development server, fast debugger and restful request dispatching
* It is a lightweight framework so it is fast.

Features of Pickel.

* Pickle is the standard way of serializing objects in Python. You can use the pickle operation to serialize your machine learning algorithms and save the serialized format to a file.
* Later you can load this file to deserialize your model and use it to make new predictions.
  1. **Comparative Technologies of Project**

**2.3.1 Django**

* [Django](https://www.guru99.com/django-tutorial.html) is a web development framework for Python. This framework offers a standard method for fast and effective website development. It helps you in building and maintaining quality web applications. It enables you to make the development process smooth and time-saving.

* Flask is a micro-framework offering basic features of the web app. This framework has no dependencies on external libraries. The framework offers extensions for form validation, object-relational mappers, open authentication systems, uploading mechanism, and several other tools.

* Flask is a good choice if you want a lightweight codebase. The best feature of Django is Robust documentation. Flask framework is suitable for single applications. Django framework allows developers to divide a project into multiple page application

# CHAPTER 3

### CHAPTER 3

### REQUIREMENTS AND ANALYSIS

#### PROBLEM DEFINITION:

To create a Recommendation system Website where the user searches for the movies and reads about movie reviews, Casts, Trivia, and other details. It is important to collect data, maintain and clean it.

Sub Problems:

* To create a dynamic data webpage where the data retrieval can be done even after loading the webpage.
* To list the top related movie list in the webpage that is predicted by the recommendation system
* To create an efficient model with high accuracy and bias-variance trade-off

#### Requirements Specification:

##### The Basic requirements for the system are as follows:

* + The system must allow the users to search for a movie by title, directors or cast names.
  + The system must let the users see the related movies links and details.

The working pre-trained model on the cloud can read and predict the given input by the dynamic webpage and turn the output vector in seconds.

#### PLANNING AND SCHEDULING:

##### Tasks (Activities):

* + Do Initial Setup:

⇨ Create or scrape data from websites containing movies details.

⇨ Set up the environment to preprocess data. The preprocessing of the dataset must be done before training the model and deploying it for the system.

⇨ Initialize git for versioning.

* + Create an environment to start the training model for reviews.

⇨ Create dataset

⇨ Train model on train set of data

⇨ Do evaluation on the test set

* Create a trained model using Pickle.

=> build a home page for the search bar

* + - Add Features of the website

=> Implement recommendation page

* + - Add recommended movies list and reviews.
* Setup the TMDB API
* Develop cosine similarity model

=> save the model using Pickle

* Create frontend pages and utilities.

=> Integrate model with the frontend

=> design frontend page using CSS, Bootstrap and JS

#### Software and Hardware Requirements:

Define the details of all the software and hardware needed for the development and implementation of the project.

##### Hardware Requirement:

* No need for graphic cards since this is a Web Application. Requires an active internet connection.
* RAM capacity: any

##### Software Requirements:

* Since this is a Web Application, it is independent of the type to OS.
* Python, jupyter, Conda python evn
* Also does not require any testing tools or compiler, etc. Need to have a Browser and an active Internet connection.

#### Preliminary Product Description:

The objective of the project is to create a Movie Recommendation System where the users can search for any movies and details about the movie along will related movies that.

The main function of the website will be to let users search for a movie and users will get the details about the casts, directors, movie descriptions and recommendations of similar movies. It uses a machine learning model to find the best similar movies of your interest to binge-watch them all in one click.

* + 1. **Project Scheduling**

Scheduling the project tasks is an important project planning activity. It involves deciding which task should be taken up and when. In order to schedule the project activities; a software project manager needs to do the following:

• Identify all the tasks needed to complete the project.

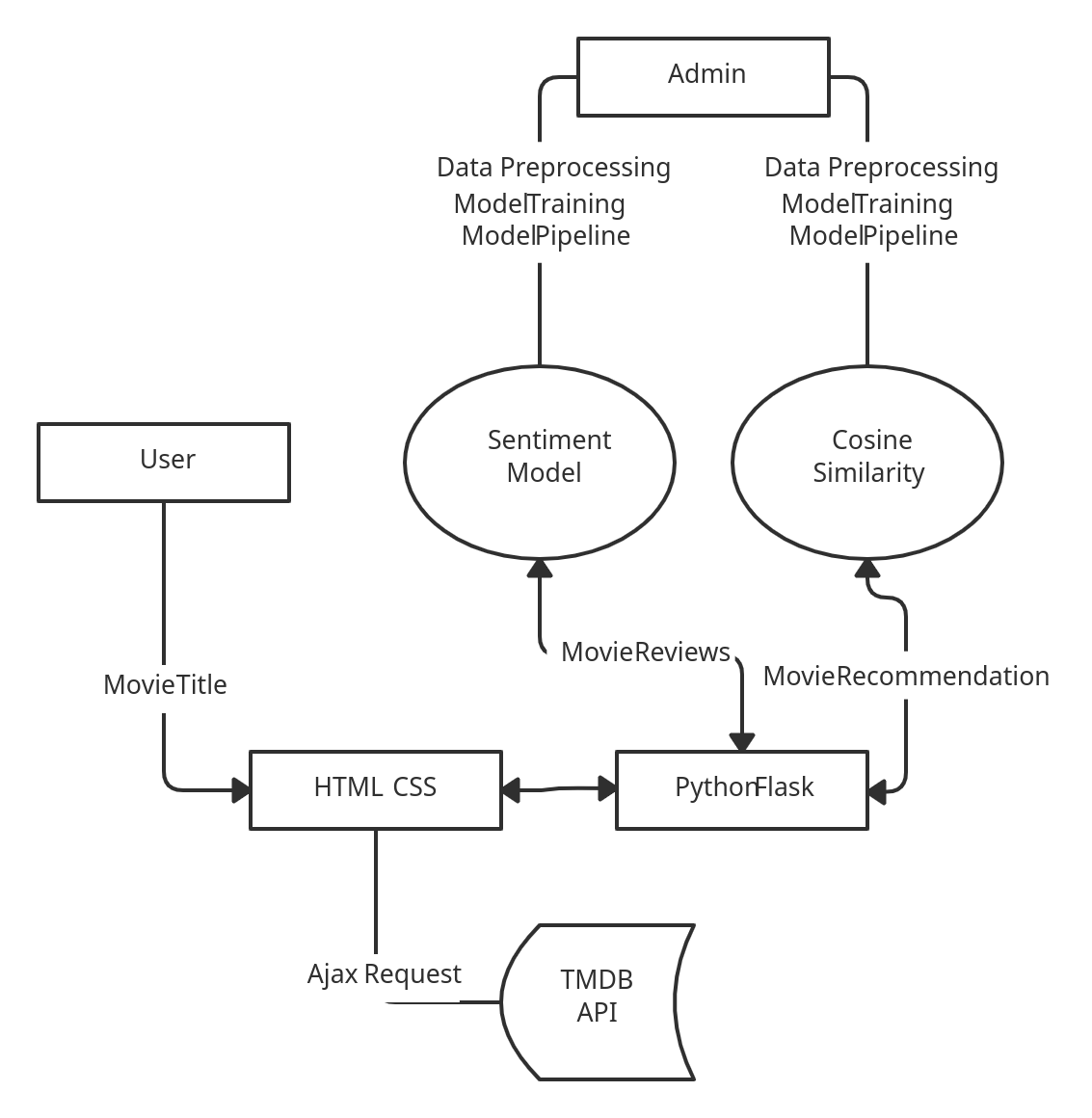
• Break down large tasks into smaller activities.

• Determine dependencies among different activities.

* 1. **Gantt Chart:**

|  |  |  |
| --- | --- | --- |
| **Task** | **Start** | **End** |
| Project References | 25-06-21 | 03-07-21 |
| Project Selection Discussion | 07-07-21 | 10-07-21 |
| Topic Approval | 11-07-21 | 11-07-21 |
| Abstract | 12-07-21 | 15-07-21 |
| Problem Statement | 20-07-21 | 22-07-21 |
| Survey of Technologies | 28-07-21 | 30-07-21 |
| Diagram Discussion | 08-08-21 | 14-08-21 |
| Requirement and Analysis | 28-08-21 | 29-08-21 |
| System Design | 29-08-21 | 07-09-27 |

#### flow diagram:

The Various models that can be made are Conceptual Models, Circuit Diagrams, Block Diagrams, etc.

# CHAPTER 4

## CHAPTER 4

**SYSTEM DESIGN**

### User interface design

**4.1.1. UML Diagrams:**

* **What is UML?**
  + UML stands for Unified Modelling Language.
  + UML is popular for its diagrammatic notations.
  + We all know that UML is for visualizing, specifying, constructing and documenting the components of software and non-software systems.
  + Hence, visualization is the most important part which needs to be understood and remembered.
  + Efficient and appropriate use of notations is very important for making a complete and meaningful model.
  + The model is useless, unless its purpose is depicted properly.

Hence, learning notations should be emphasized from the very beginning. Different notations are available for things and relationships.

UML diagrams are made using the notations of things and relationships.

Extensibility is another important feature which makes UML more powerful and flexible.

* **Why Do We Use UML?**

A complex enterprise application with many collaborators will require a solid foundation of planning and clear, concise communication among team members as the project progresses.

Visualizing user interactions, processes, and the structure of the system you're trying to build will help save time down the line and make sure everyone on the team is on the same page

* **What are the Types of UML Diagrams?**

1. Entity Relationship Diagrams.
2. Use Case Diagrams.
3. Sequence Diagrams.
4. Activity Diagrams

### BASIC MODULES:

* + - Initial Setup **=>** This is the starting of the project where we will be collecting data from open sources and by web scraping from different sites containing useful information. Initializing git version.
    - Model Preprocessing & Training **=>** Here we will preprocess the data (data cleaning/wrangling), after the data is ready to train, we will train the model and save the trained model as a file using pickle.
    - Setup TMDB API => Here we will access the TBDM API to get the details about the movie’s trivia, description and reviews.
    - Home Search Page **=>** Designing of the home page where the user can search for the title of a movie, casts name, genre, and also linking the TBDM API with a framework
    - Recommendation Page **=>** On this page user will get the searched movie details about its description, casts, genre, reviews along with its recommended movies list

### Data Design:

Data design will consist of how data is organized, managed and manipulated.

* + 1. Schema Design: The schema will consist of the following fields:
       1. Movie title
       2. Genres
       3. Director Name
       4. Actors Name

##### Data Integrity and Constraints:

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELDS** | **CONSTRAINTS** | **LIMIT (char**  **)** | **DATA TYPE** |
| director\_name | None | Min:2 Max:25 | String |
| genres | None | Min:2 Max:25 | String |
| movie\_title | Primary | Min:2 Max:25 | String |
| actor\_1\_name | None | Min:2 Max:25 | String |
| actor\_2\_name | None | Min:2 Max:25 | String |
| actor\_3\_name | None | Min2 Max:25 | String |
| comb | Primary | Min:12 Max:60 | String |

* 1. **LOGIC DIAGRAMS:**

Logic diagrams include Data Flow Diagrams, Activity Diagrams and Component diagrams.

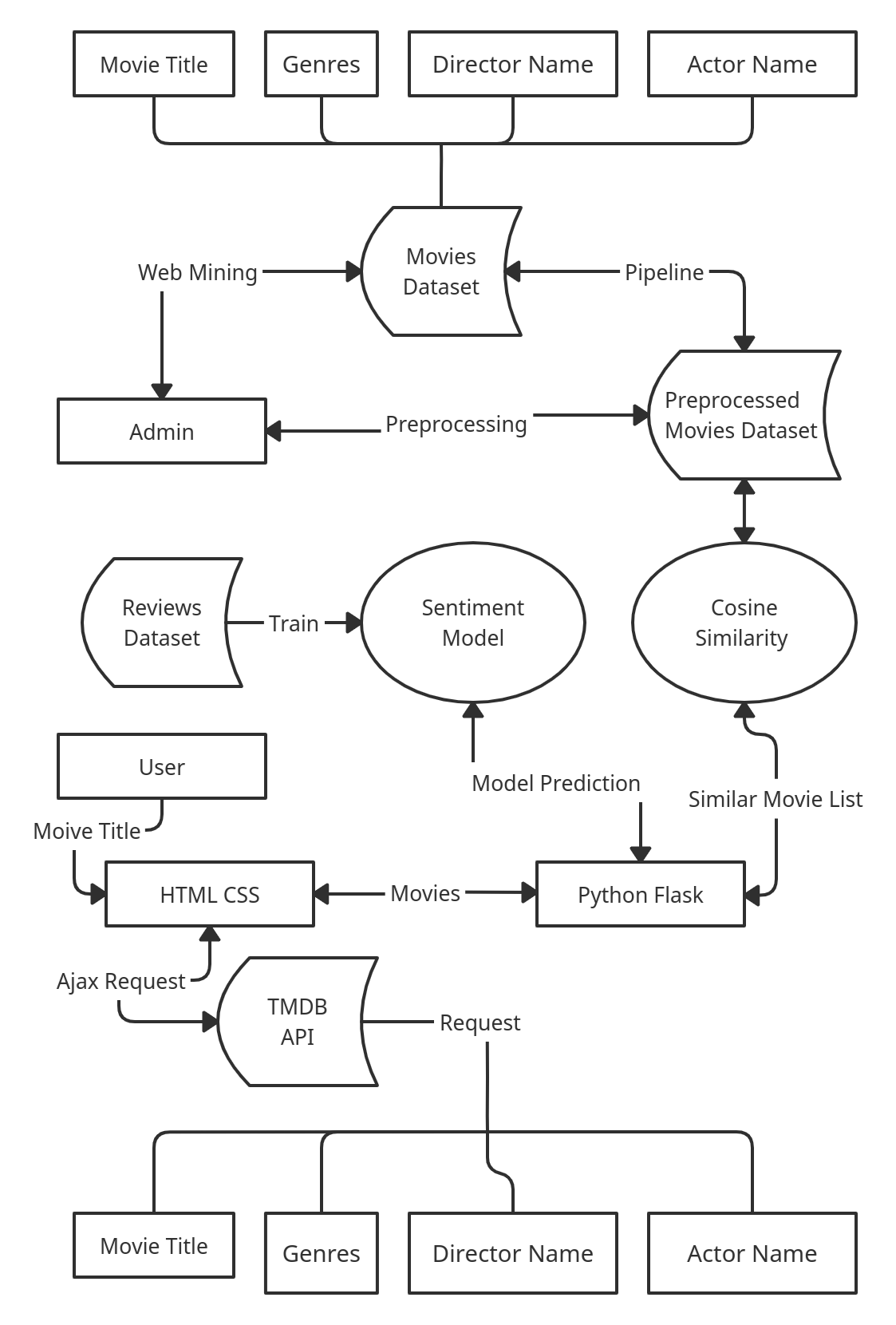
* + 1. **Data Flow Diagrams:**

A data-flow diagram is a way of representing a flow of data through a Process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. DFD consists of processes, flows, warehouses, and terminators.

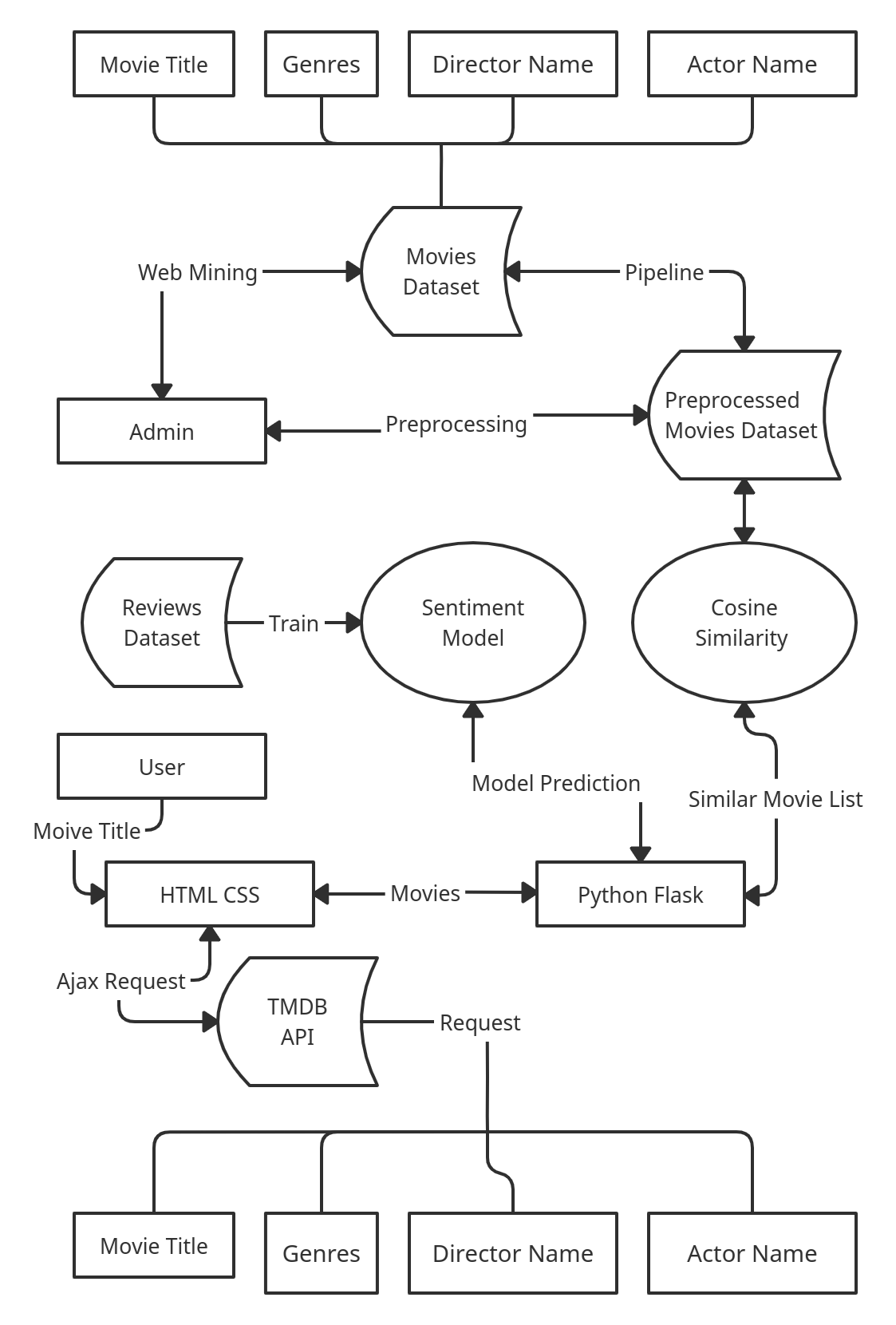
DFD consists of processes, flows, warehouses, and terminators.

###### Notations:

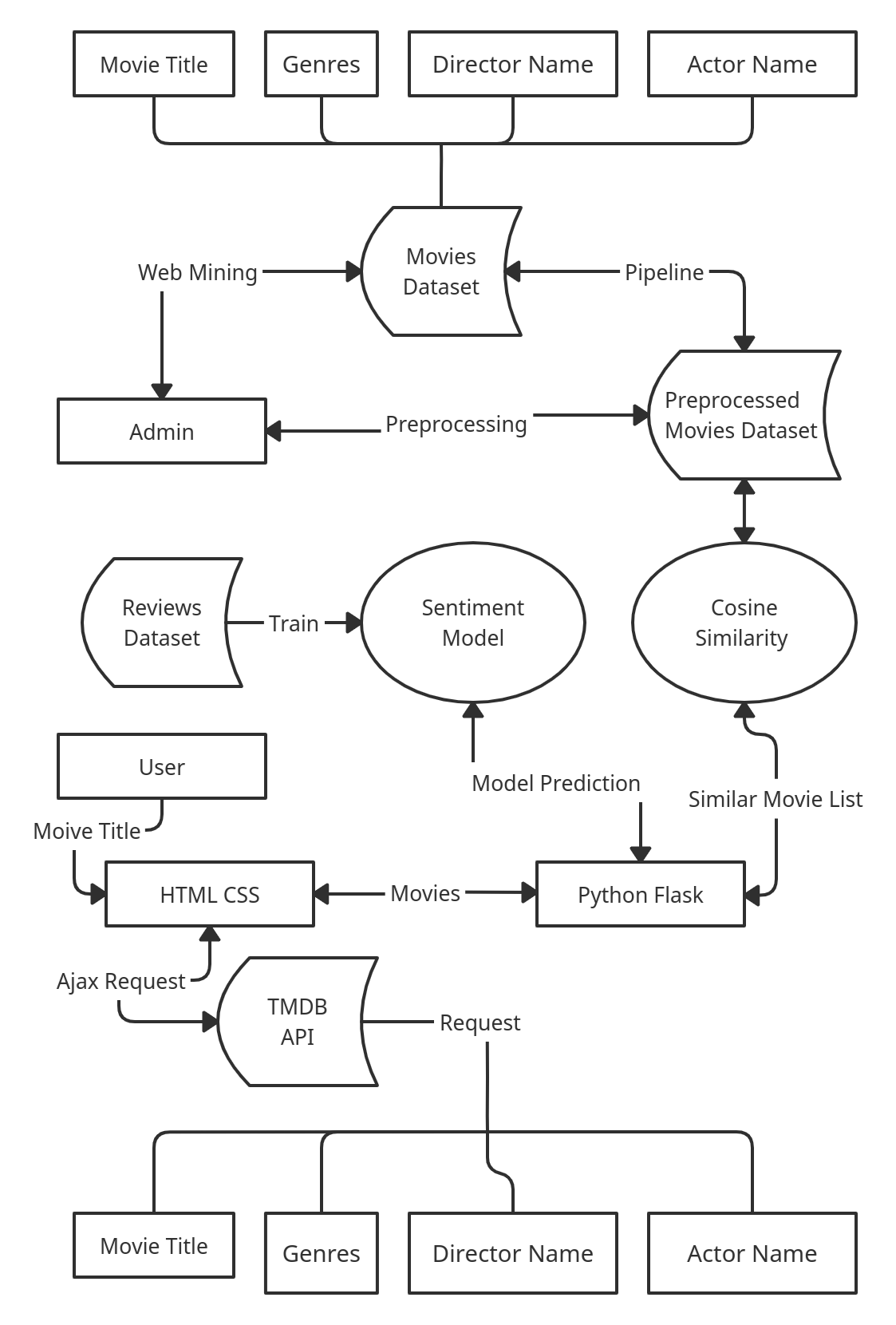
* + - 1. Process**: -**
         1. It shows a process that transforms data inputs into data outputs.
         2. Symbol



* + - 1. Data Flow:
         1. A curved line shows the flow of data into or out of a process or data store.
         2. Symbol



* + - 1. Data Store:
         1. A set of parallel lines shows a place for the collection of data items. A data store indicates that the data is stored which can be used at a later stage or by the other processes in a different order. The data store can have an element or group of elements.
         2. Symbol:



* + - 1. External Entity:
    1. External entity acts as a source of system inputs or sinks of system outputs.
    2. Symbol

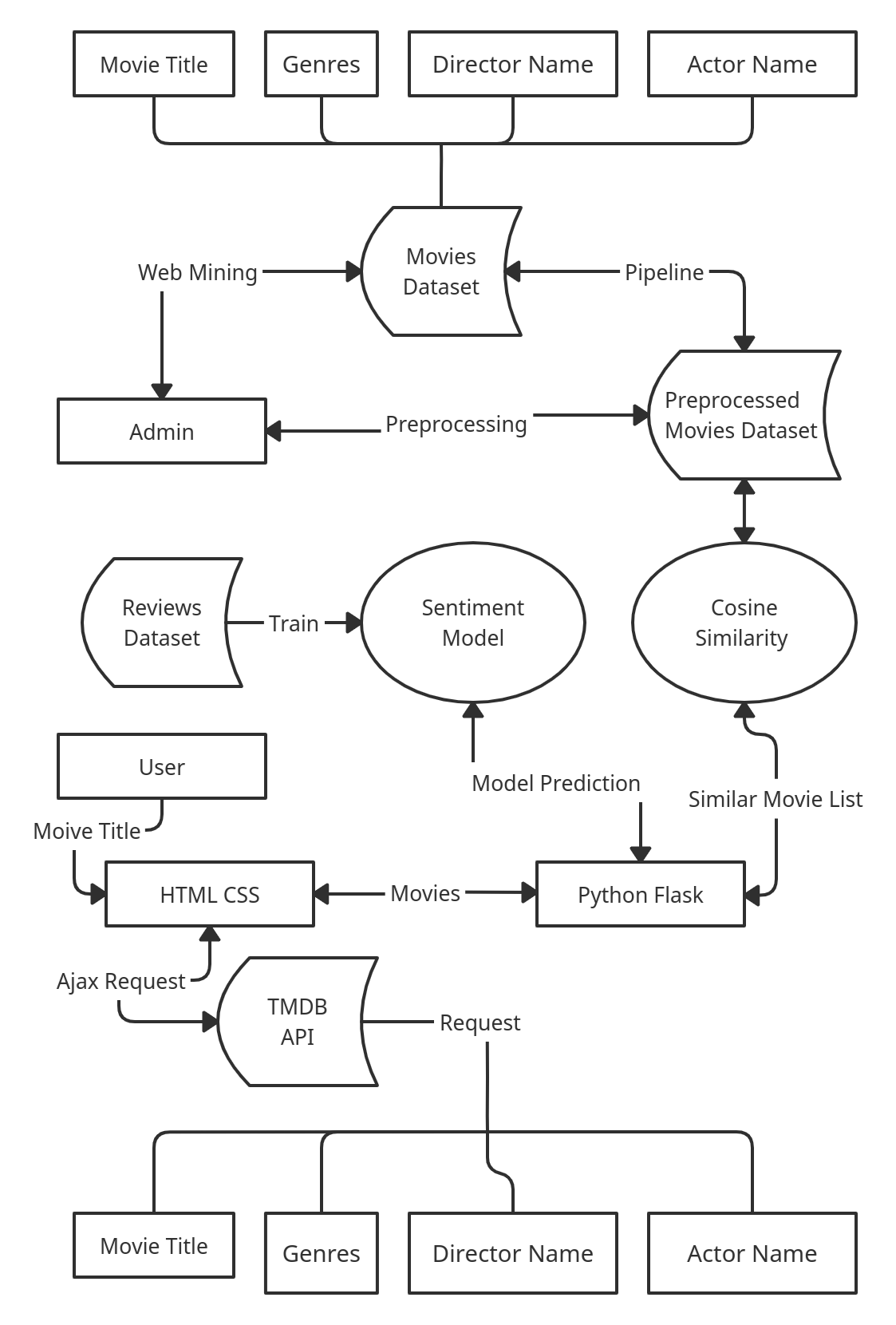
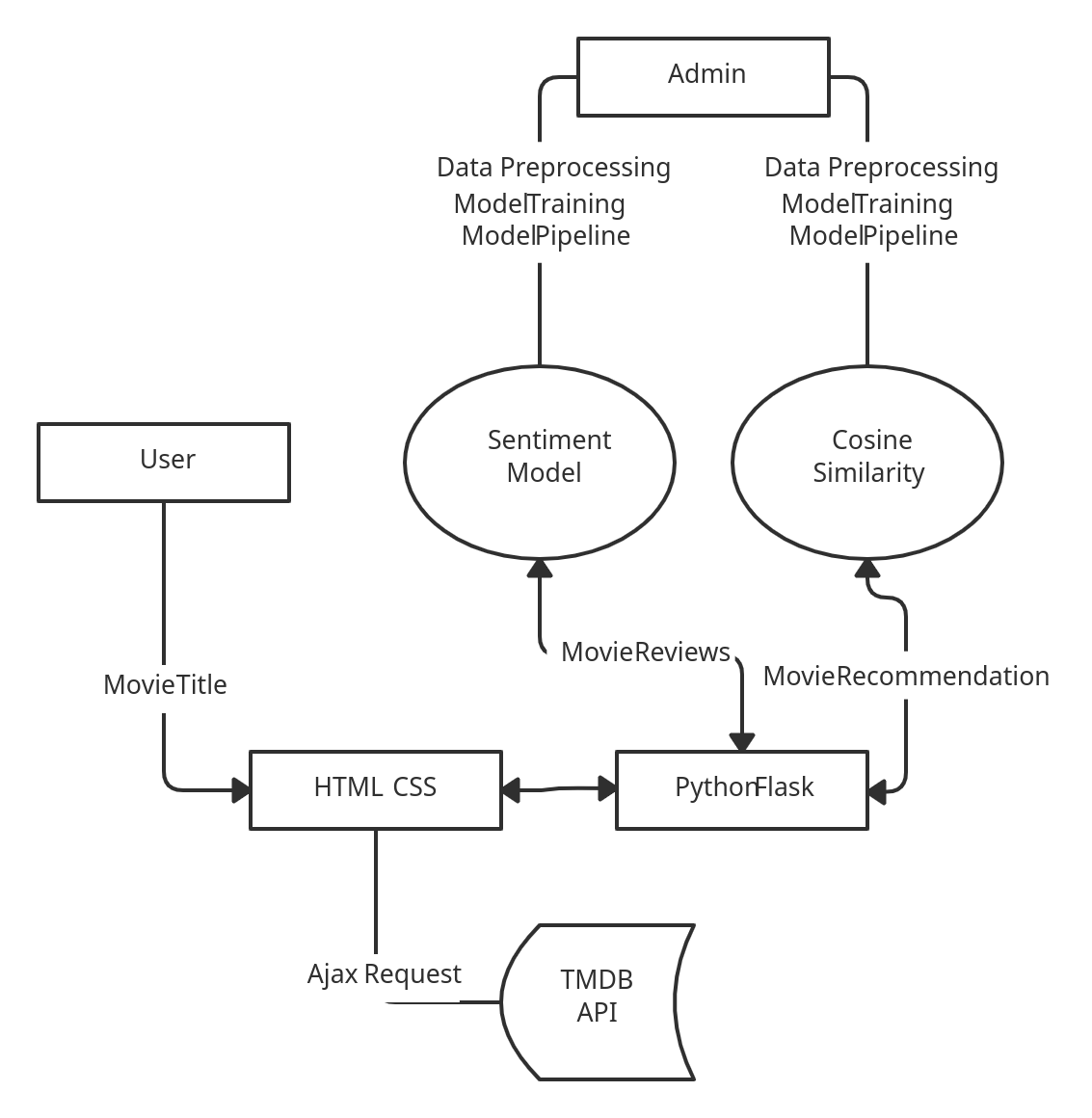
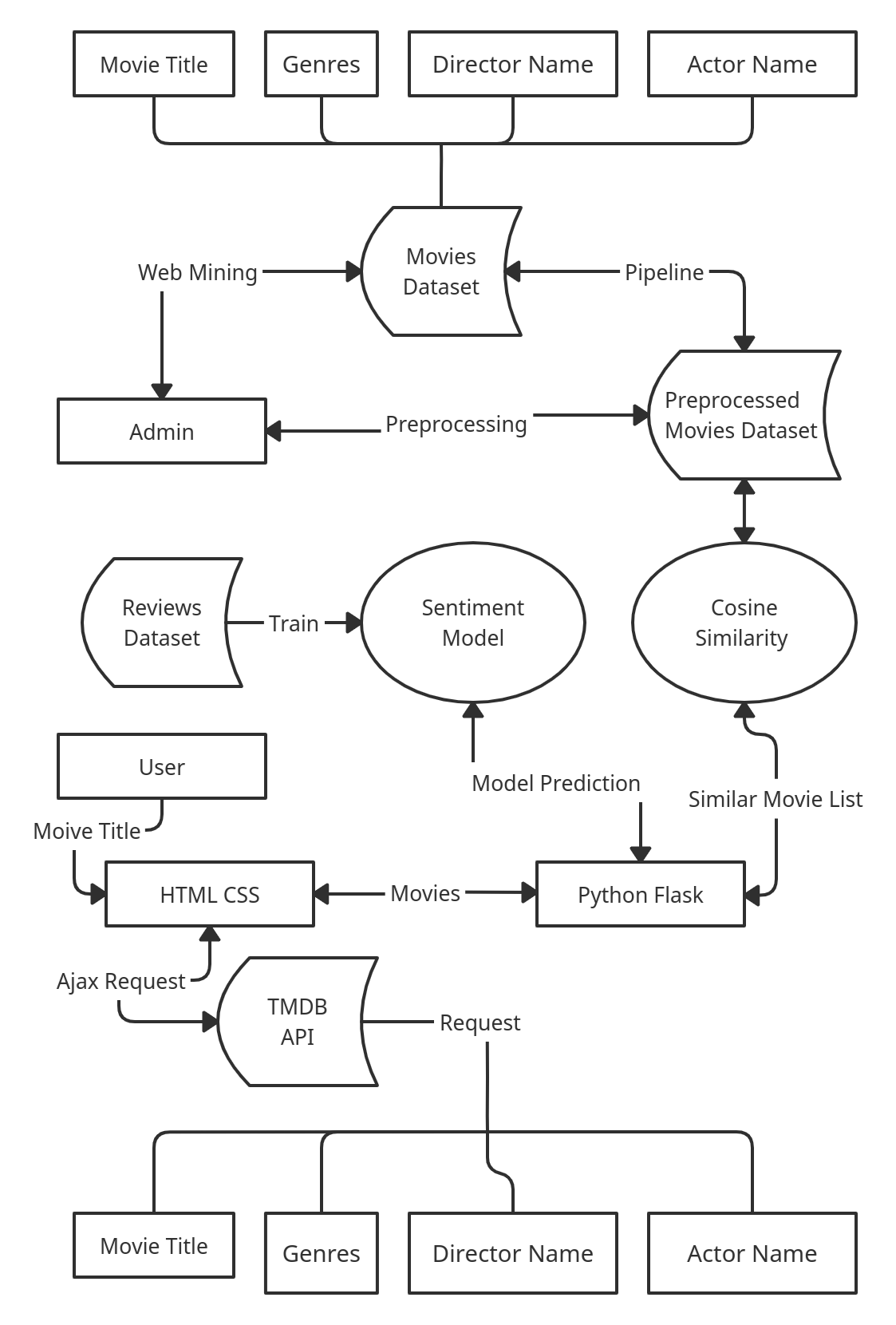


Figure 4.1





* + 1. **Activity Diagrams**

The activity diagram is another important diagram in UML to describe the dynamic aspects of the system activity it diagrams ly a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all types of flow control by using different elements such as fork, join, etc.

### Notations:

###### Initial State or Start Point:

A small filled circle followed by an arrow represents the initial action state or the start point for any activity diagram.



###### Activity or Action State:

An action state represents the non-interruptible action of objects. You can draw an action state in Smart Draw using a rectangle with rounded corners.

###### Action Flow:

The action flows, also called edges and paths, illustrate the transitions from one action state to another. They are usually drawn with an arrowed line.



###### Decisions and Branching:

A diamond represents a decision with alternate paths. When an activity requires a decision before moving on to the next activity, add a diamond between the two activities. The outgoing alternates should be labelled with a condition or guard expression. You can also label one of the paths "else."

###### Final State or End Point:

An arrow pointing to a filled circle nested inside another circle represents the final action state.



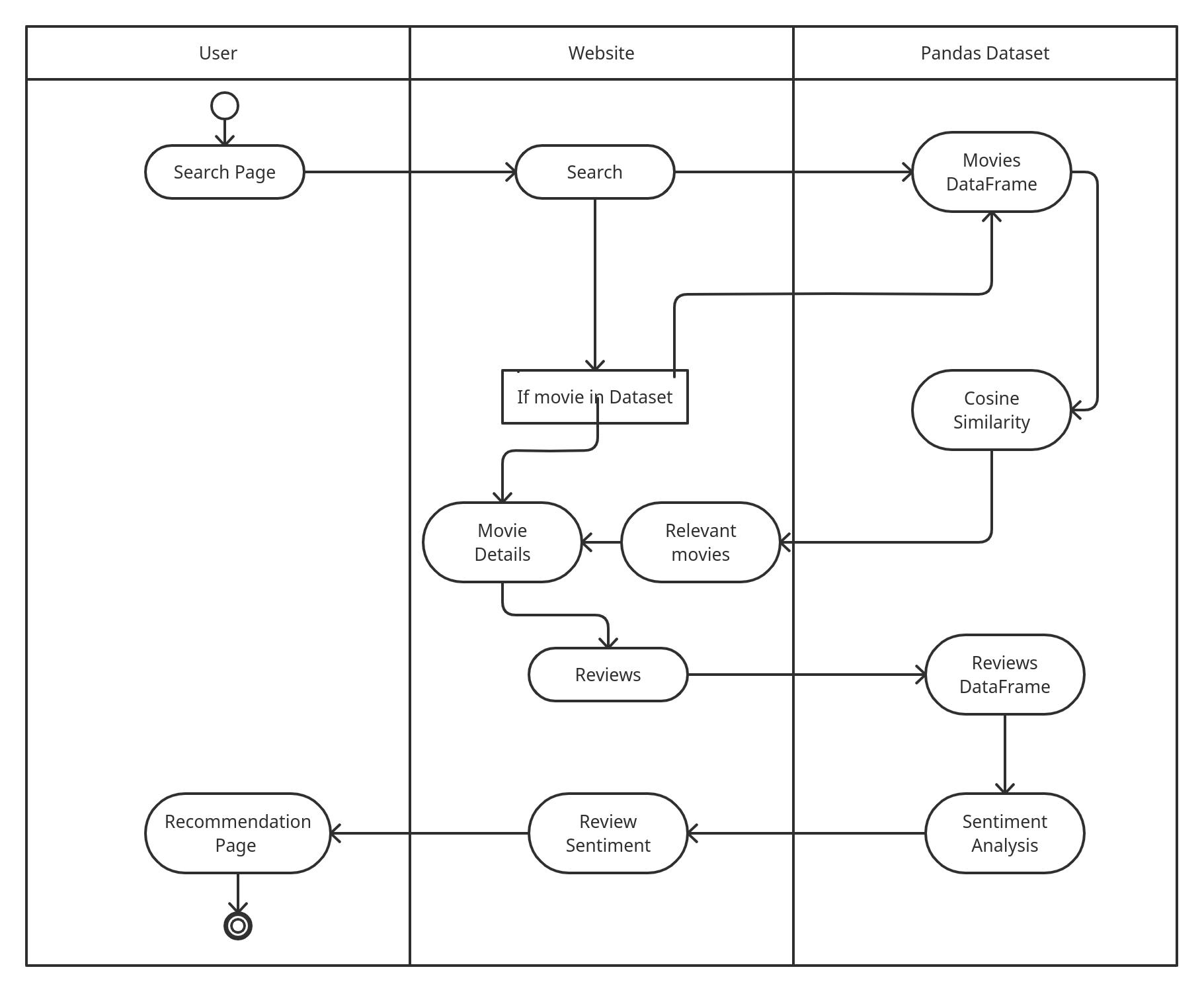


Figure 4.2

* + 1. **Use Case Diagram:**

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

#### Notations: -

1. **Actor: -**
   1. Actors are the entities that interact with a system. Although in most cases, actors are used to representing the users of the system, actors can be anything that needs to exchange information with the system. So, an actor may be people, computer hardware, other systems, etc.
   2. Symbol

#### Use Case: -

* 1. A use case represents a user goal that can be achieved by accessing the system or software application. In Visual Paradigm, you can make use of the sub-diagram feature to describe the interaction between user and system within a use case by creating a sub-sequence diagram under a use case.
  2. Symbol



#### Include: -

* 1. An include relationship specifies how the behavior for the inclusion use case is inserted into the behavior defined for the base use case.
  2. Symbol



1. Extend: -
   1. An extended relationship specifies how the behavior of the extension use case can be inserted into the behavior defined for the base use case.
   2. Symbol



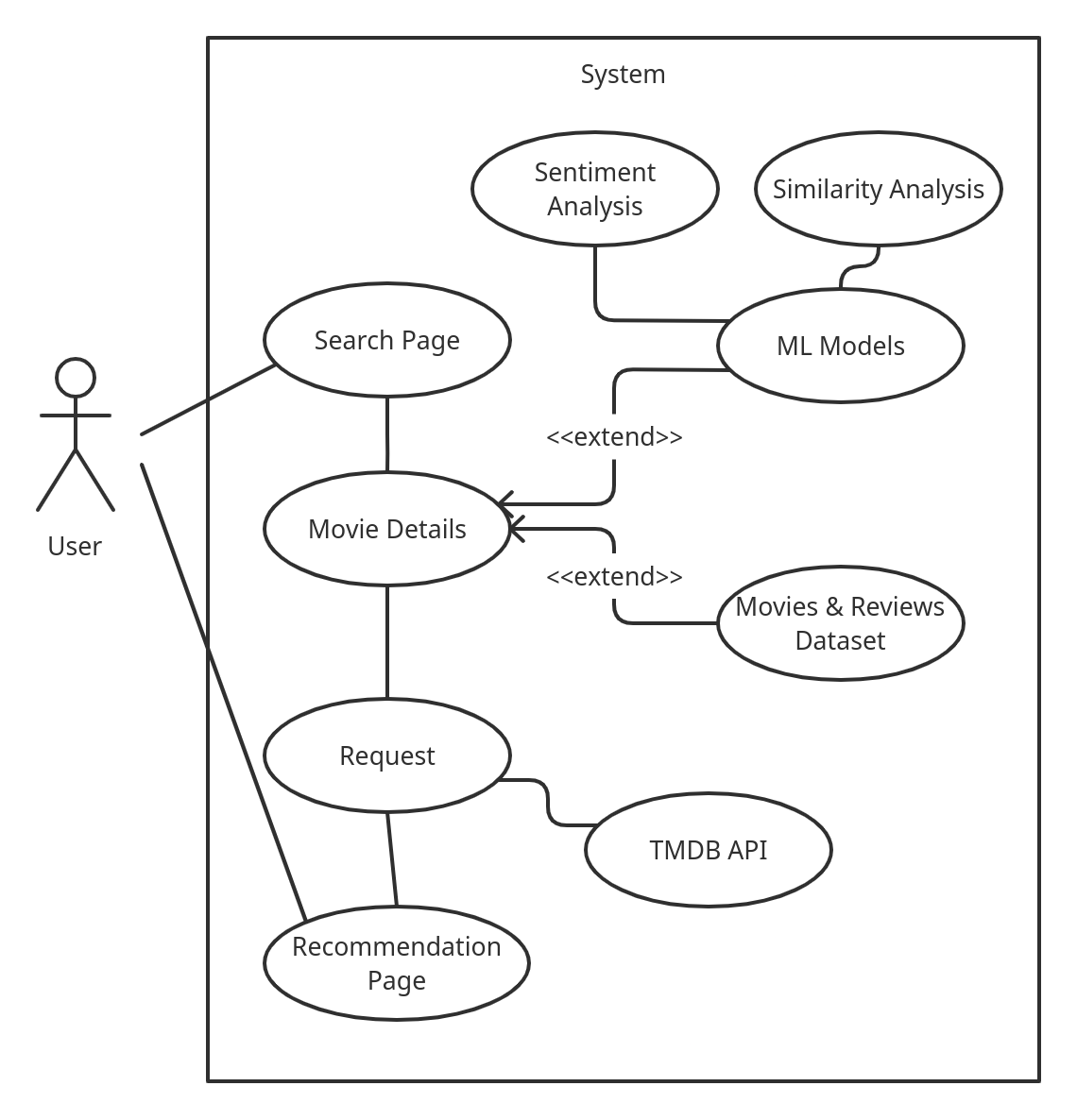


Figure 4.3

## CHAPTER NO 5

**CHAPTER 5**

### 5.1 IMPLEMENTATION APPROACHES

A planned approach for a successful project is of utmost importance as the saying goes like

* "Plans are of little importance, but planning is essential."
* The planning approach used in this project is AGILE METHODOLOGY AGILE

METHODOLOGY

* Agile is a methodology of early delivering business value with less bureaucracy.
* It is a collection of software development methodology which works on an iterative and incremental method building the software from the very initiation of the project development instead of building all the software at once
* Agile discovers what the customer and how to build the code and change things along the way.
* Agile works by breaking down the project into smaller chunks and then continuously delivering them in short two weeks’ cycles called iterations.
* In an agile development process, solutions are obtained by self-organizing • teams which collaborate, discuss, and constantly strive to improve their working
* process and become more effective in what they do.

STANDARDS USED IN THIS PROCESS.

* Individuals and interactions over processes and tools
* Working hardware over comprehensive documentation
* Customer collaboration over contract negotiation
* Responding to change over following a plan

STEPS IN AGILE METHODOLOGY

1. We not only implement requirements during an iteration, we also fix defects (Disciplined agile teams, particularly working at scale, may have a parallel testing effort during construction iterations where these defects are found)
2. **Define the business opportunity**: This includes exploring how the new functionality will improve your organization's presence in the market, how it will impact profitability, and how it will impact the people within your organization.
3. Assess the feasibility: During the Concept Phase you will want to do just enough feasibility analysis to determine if it makes sense to invest in the potential project. Depending on the situation you may choose to invest very little effort in considering feasibility
4. Actively working with stakeholders to initially model the scope of the system: To promote active stakeholder participation you should use inclusive tools, such as white boards to do this modelling – our goal is to understand the problem and solution domain, not to create mounds of documentation. The details of these requirements are modelled on a just in time (JIT) basis in model storming sessions during the development cycles.
5. Estimating the project: You'll need to put together an initial estimate for your agile project based on the initial requirements, the initial architecture, and the skills of your team. This estimate will evolve throughout the project.

### 5.2 CODE DETAILS AND CODE EFFICIENCY

**5.2.1 CODE EFFICIENCY**

Optimization is a program transformation technique, which tries to improve the code by making it consume less resources (i.e., CPU, Memory) and deliver high speed.

In optimization, high-level general programming constructs are replaced by very efficient lowlevel programming codes.

A code optimizing process must follow the rules given below

* The output code must not, in any way, change the meaning of the program.
* Optimization should increase the speed of the program and if possible, the program should demand a smaller number of resources.
* Optimization should itself be fast and should not delay the overall compiling process. Efforts for an optimized code can be made at various levels of compiling the process.
* At the beginning, users can change/rearrange the code or use better algorithms to write the code.
* After generating intermediate code, the compiler can modify the intermediate code by address calculations and improving loops.
* While producing the target machine code, the compiler can make use of memory hierarchy and CPU registers.
* Optimization can be categorized broadly into two types: machine independent and machine dependent.

#### MACHINE-INDEPENDENT OPTIMIZATION

In this optimization, the compiler takes in the intermediate code and transforms a part of the code that does not involve any CPU registers and/or absolute memory locations.

#### MACHINE-DEPENDENT OPTIMIZATION

This type of optimization is done after the target code has been generated and when the code is transformed according to the target machine architecture. It involves CPU registers and may have absolute memory references rather than relative references. Machine-dependent optimizers put efforts to take maximum advantage of memory hierarchy.

#### BASIC BLOCK IDENTIFICATION

We may use the following algorithm to find the basic blocks in a program:

* Search header statements of all the basic blocks from where a basic block starts:
* First statement of a program.
* Statements that are target of any branch (conditional/unconditional).
* Statements that follow any branch statement.
* Header statements and the statements following them form a basic block.
* A basic block does not include any header statement of any other basic block.

#### CONTROL FLOW GRAPH

Basic blocks in a program can be represented by means of control flow graphs. A control flow graph depicts how the program control is being passed among the blocks. It is a useful tool that helps in optimization by help locating any unwanted loops in the program.

#### DEAD-CODE ELIMINATION

Dead code is one or more than one code statements, which are:

* Either never executed or unreachable, • Or if executed, their output is never used.
* Thus, dead code plays no role in any program operation and therefore it can simply be eliminated.

### 5.3 TESTING APPROACH

**5.3.1 UNIT TESTING**

**WHAT IS UNIT TESTING?**

* Unit testing, a testing technique using which individual modules are tested to determine if there are any issues by the developer himself. It is concerned with functional correctness of the standalone modules.
* The main aim is to isolate each unit of the system to identify, analyze and fix the defects.
* There are different methods that can be used for testing. This chapter briefly describes the methods available.

#### BLACK-BOX TESTING

• The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

#### WHITE-BOX TESTING (ADOPTED APPROACH)

* White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called glass testing or open-box testing. In order to perform white-box testing on a component, a tester needs to know the internal workings of the code.
* The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

#### GREY-BOX TESTING

• Grey-box testing is a technique to test the developed software with having a limited knowledge of the internal workings of an application. Mastering the domain of a system always gives the tester an edge over someone with limited domain knowledge. Unlike black-box testing, where the tester only tests the systems working; in grey-box testing, the tester has access to architecture documents. Having this knowledge, a tester can prepare better test data and test scenarios while making a test plan.

#### 5.3.2 INTEGRATED TESTING

**WHAT IS INTEGRATION TESTING?**

• Upon completion of unit testing, the units or modules are to be integrated which gives raise to integration testing. The purpose of integration testing is to verify the functional, performance, and reliability between the modules that are integrated.

##### INTEGRATION STRATEGIES

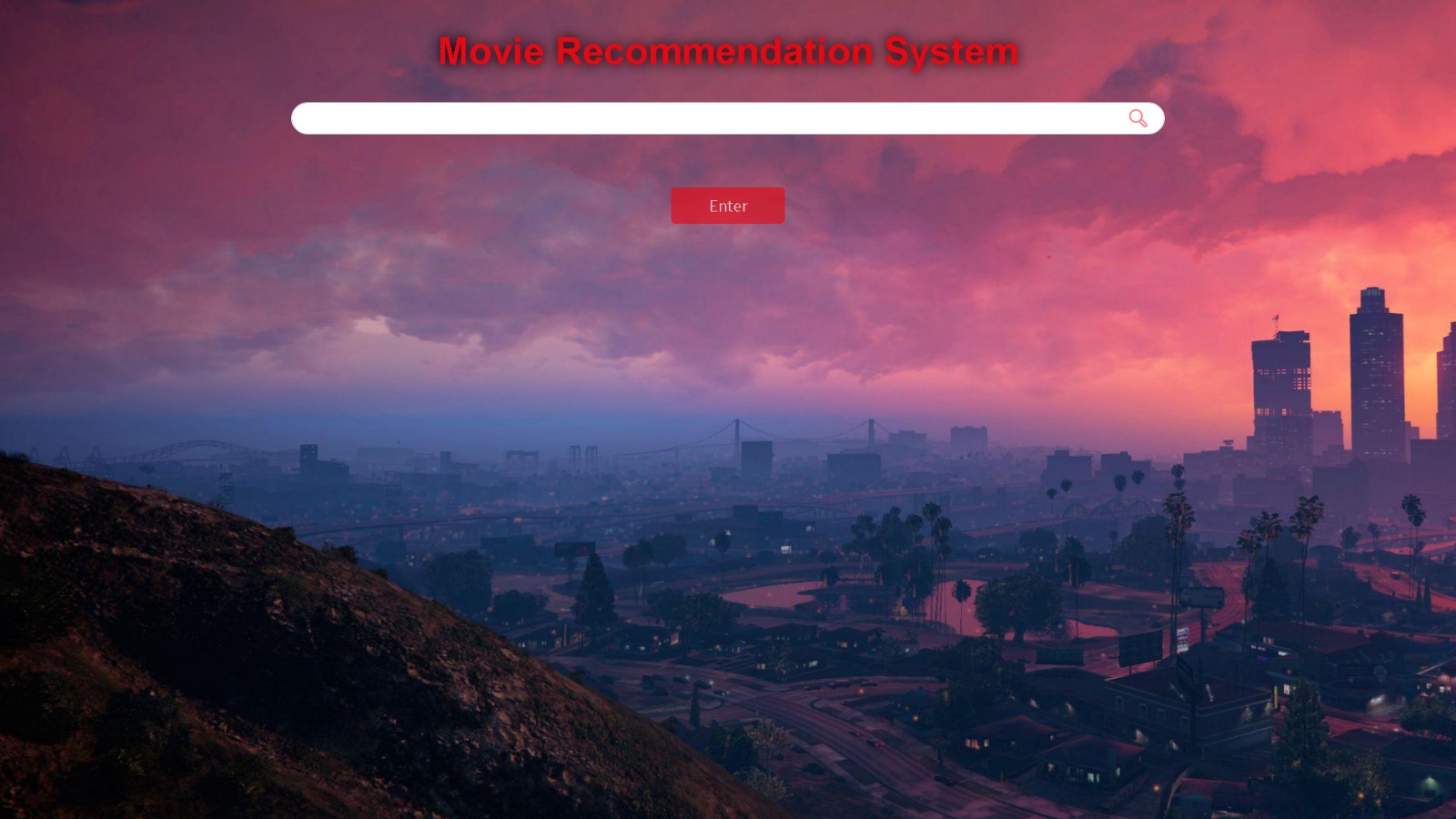
* Big-Bang Integration
* Top-Down Integration
* Bottom-Up Integration
* Hybrid Integration

**5.3.3 BETA TESTING**

**WHAT IS BETA TESTING?**

* Beta testing also known as user testing takes place at the end users site by the end users to validate the usability, functionality, compatibility, and reliability testing.
* Beta testing adds value to the software development life cycle as it allows the "real" customer an opportunity to provide inputs into the design, functionality, and usability of a product. These inputs are not only critical to the success of the product but also an investment into future products when the gathered data is managed effectively.
* There are number of factors that depends on the success of beta testing:
  + - Test Cost
    - Number of Test Participants
    - Duration of Test
    - Demographic coverage

## CHAPTER NO 6



## CHAPTER NO 7

**COST AND BENEFIT ANALYSIS**

The main objective behind cost and benefit Analysis is to assess the feasibility to determine whether the developed project has a reasonable chance of success.

A Requirement of the economic feasibility of the Project always requires a thorough cost benefit analysis.

**DEVELOPING A COST BENEFIT ANALYSIS IS A 3 STEP PROCESS.**

Step 1: Estimate the anticipated Development and Operational Costs.

1. Development costs are those that are incurred during the development of the new system. It includes: System Analyst Time, Programmer Time, User Time, possible hardware cost, possible software purchase cost and Possible outside service cost
2. Operational costs are those that will be incurred after the system is put into production. It includes: Computer cost, Communication cost, Operation staff cost, Incremental user cost, Maintenance costs and Server-side applications cost.

Step 2: The Second step is to estimate the anticipated financial benefits.

Financial benefits Are the expected annual saving or increase in revenue derived from the installation of new system.

Step 3: In the Third Step, the Cost/Benefit Analysis is Calculated Based

on Detailed Estimated Costs and Benefits.

The most frequent error that happens here Is Lack of thorough Definition of costs and Benefits.

**ANALYSIS, DESIGNING AND CODING COST:**

Analysis, Designing and Coding = No. of people involved in Project \* Charges per day

No. of people involved in Project = 2

Charges per day = 60

No. of Days = 75

Analysis, Designing and Coding = 2 \* 60 \* 75 = Rs.9,000/-

**TESTING AND MAINTENANCE COST:**

Testing and Maintenance Cost = No. of hours \* Charges per day \* No. of Days

No. of Hours = 2

Charges per day = 30

No. of Days = 18

Testing and Maintenance Cost = 2 \* 50 \* 25= Rs.1,080/-

## Overall Cost = Rs. 10,080/-

## CHAPTER NO 8

### 8.1 CONCLUSION

**8.1.1 SIGNIFICANCE OF THE SYSTEM:**

* This is a website that helps users to get there movie recommended and give them other recommendations.
* Users can type there preferred movie and get recommendation or can see our top movies as a search for their genre taste.
* Every user can check the movie cast and the details regarding the movie and some movie recommendation.
* They can check different set of sentimental reviews given to the movie.
* This website can be useful as it primarily focus on recommendation and movie details.

**8.2 LIMITATIONS OF THE SYSTEM:**

* Expanding the data will need better database management.
* Limited movies title available.
* No option for user profile.
* No option for adding titles by the users.
* Overall security of the Website can be improved.

**8.3 FUTURE SCOPE OF THE PROJECT:**

* Scaling up resources will allow more people to use the website.
* Implement the system variation (complex model for recommendation).
* Adding reviews to upcoming movies.
* Adding short videos about movies.
* Improving the overall security of the project will allow it to be used commercially.
* Adding weekly trending. Best of all time section and more.

**8.4. References**

* **Stack overflow:** <https://stackoverflow.com>
* **Stack overflow:** <https://www.geeksforgeeks.org/>
* **YouTube:** <https://www.youtube.com/>
* **jQuery:** <https://learn.jquery.com/>
* **Bootstrap 4:** <https://getbootstrap.com/docs/4.6/getting-started/introduction/>
* **W3 Schools:** <https://www.w3schools.com/>
* **Kaggle:** <https://www.kaggle.com/>