**Design & Development of Semester Course Registration System**

*A Project Submitted in Partial Fulfillment of the Requirements for the*

*Degree of*

Bachelor of Science in Computer Science and Engineering

*by*

**Md. Kawsar Hossain**

CSE04606237

&

**Khaled Mohammad Faisal**

CSE04606247

Supervised by: Adnan Ferdous Ashrafi

Senior Lecture



Department of Computer Science and Engineering

STAMFORD UNIVERSITY BANGLADESH

December 2021

i

**Abstract**

Web-based services offer users convenient access to and the ability to manipulate information

that is of concern to such services. Due to the high requirements in functionality and performance, these systems are often very large in terms of the size of the underlying software.

The use of Component-Based Design (CBD) for web-based software comes into play for this

very reason. Not only does Component-Based Software Engineering (CBSE) address the manageability issue of web-based software, but it also ensures greater consistency and high reusability of web-based components. These advantages in turn lead to better productivity and hence better quality of the overall design of the system. In order to exemplify the effectiveness of CBD, the design and implementation of a component-based online course registration system is proposed for this project. The application will allow students, department members, faculty members and the registry members to view and make changes to course registration related issues for a specific semester. The system will be developed using a Service-Oriented Architecture (SOA), which involves grouping components into web services.

Keywords: Web-based, Component-Based Design, Component-Based Software Engineering,

online course registration, PHP; MySQL; jQuery; Apache; Database; Database Management System; Front-end; Back-end.

ii

**Approval**

The project report “Design & Development of Semester Course registration System” submitted by Md. Kawsar Hossain ID: CSE04606237 & Khaled Mohammad Faisal ID: CSE04606247, to the Department of Computer Science & Engineering, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science (B.Sc.) in Computer Science & Engineering and as to its style and contents.

Board of Examiner’s Name, Signature and Date:

|  |  |  |
| --- | --- | --- |
| **……………………………...** | **……………………………..** | **……………………………..** |
| **(Board Member 1)** | **(Board Member 2)** | **(Board Member 3)** |
| Date: | Date: | Date: |

Supervisor’s Signature and Date:

**……………………………...**

**Supervisor Name**

Date:

iii

**Declaration**

We, hereby, declare that the work presented in this Project is the outcome of the investigation performed by us under the supervision of Adnan Ferdous Ashrafi, Lecturer, Department of Computer Science & Engineering, Stamford University Bangladesh. We also declare that no part of this Project and thereof has been or is being submitted elsewhere for the award of any degree or Diploma.

Signature and Date:

**……………………………...**

**Student Name: Md. Kawsar Hossain**

Date:

**……………………………...**

**Student Name: Khaled Mohammad Faisal**

Date:

iv

Dedicated to…

Beloved Parents and Teachers

v

**Acknowledgements**

First of all, we would like to express our highest gratitude to the almighty God for his kindness on us that made it possible for us to complete the study and preparation of this project. Starting from the initial idea to final execution, everyone from Stamford University supported us to build and create something around movie recommendation system. We would like to thank our supervisor, Adnan Ferdous Ashrafi for his guidance and having so many endless discussions. He helped us in many ways to build the project successfully.

vi

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **List of Figures ...........................................................................................................................** | | **1** |
| **List of Tables ............................................................................................................................** | | **2** |
| 1. **Introduction .................................................................................................................** | | **3** |
| 1.1 | Problem Statement ....................................................................................................... | 3 |
| 1.2 | Goals ............................................................................................................................ | 3 |
| 1.3 | Purpose of the work ..................................................................................................... | 3 |
| 1.4 | Objectives of the project .............................................................................................. | 4 |
| 1.5 | Scope of the project ..................................................................................................... | 4 |
| 1.6 | Use-cases of Recommendation systems ...................................................................... | 4 |
| 1.7 | Importance of Recommendation Systems ................................................................... | 4 |
| 1.8 | What can be recommended? ........................................................................................ | 5 |
| 1.9 | Benefits of a movie recommendation system .......................................................... | 5 |

|  |  |
| --- | --- |
| 1. **Previous Work ..............................................................................................................** | **6** |

|  |  |  |
| --- | --- | --- |
| 1. **System Methodology ...................................................................................................** | | **12** |
| 3.1 | Requirement .............................................................................................................. | 12 |
| 3.2 | Design ........................................................................................................................ | 12 |
| 3.3 | Implementation .......................................................................................................... | 12 |
| 3.4 | Requirement Analysis ............................................................................................... | 13 |
| 3.5 | Use Case Diagram ..................................................................................................... | 13 |
| 3.6 | Data flow Diagram .................................................................................................... | 16 |

vii

|  |  |  |  |
| --- | --- | --- | --- |
|  | 3.7 | All table with description: ......................................................................................... | 19 |
| **4** | **. Implementation ..............................................................................................................** | | **22** |
|  | 4.1 | Main Window Information ........................................................................................ | 22 |
|  | 4.2 | Admin login Information ........................................................................................... | 23 |
|  | 4.3 | Admin movie upload form ........................................................................................ | 24 |
| **5** | **. Conclusion: .....................................................................................................................** | | **25** |
|  | 5.1 | Future work: ................................................................................................................... | 25 |
|  | 5.2 | Limitations ..................................................................................................................... | 25 |
| **6** | **. References .......................................................................................................................** | | **26** |

Viii

**List of Figures**

|  |  |
| --- | --- |
| [Figure 1.10.1: Movie Rating ......................................................................................................](#page13) | [5](#page13) |
| [Figure 2.1.1: Content Based Filtering [1] ..................................................................................](#page14) | [6](#page14) |
| [Figure 2.2.1: Collaborative Based Filtering [1] .........................................................................](#page15) | [7](#page15) |
| [Figure 2.3.1: Hybrid Based Filtering [2] ...................................................................................](#page16) | [8](#page16) |
| [Figure 2.5.1: Youtube Recommendation System [4] ...............................................................](#page19) | [11](#page19) |
| [Figure 3.5.1: User Interaction ..................................................................................................](#page21) | [13](#page21) |
| [Figure 3.5.2: Admin Panel .......................................................................................................](#page21) | [13](#page21) |
| [Figure 3.5.3: User Registration Panel ......................................................................................](#page22) | [14](#page22) |
| [Figure 3.5.4: Admin Registration Panel ..................................................................................](#page22) | [14](#page22) |
| [Figure 3.5.5: User Watching Diagram .....................................................................................](#page23) | [15](#page23) |
| [Figure 3.5.6: Admin Movie Uploading Process ......................................................................](#page23) | [15](#page23) |
| [Figure 3.5.7: ER Diagram ........................................................................................................](#page24) | [16](#page24) |
| [Figure 3.6.1: Level 0 Data Flow Diagram ...............................................................................](#page24) | [16](#page24) |
| [Figure 3.6.2: Level One Data Flow Diagram ..........................................................................](#page25) | [17](#page25) |
| [Figure 3.6.3: Level Two Data Flow Diagram ..........................................................................](#page26) | [18](#page26) |
| [Figure 4.1.1: Main Window .....................................................................................................](#page30) | [22](#page30) |
| [Figure 4.2.1: Admin Login Page..............................................................................................](#page31) | [23](#page31) |
| [Figure 4.2.2: Admin Password Changing Page .......................................................................](#page31) | [23](#page31) |
| [Figure 4.3.1: Admin Movie Upload From ...............................................................................](#page32) | [24](#page32) |
|  |  |

**List of Tables**

[Table 3.7.1: Categories Table](#page27) [19](#page27)

[Table 3.7.2: Table For Movies Profile](#page27) [19](#page27)

[Table 3.7.3: Table For Rating Profile](#page28) [20](#page28)

[Table 3.7.3: Table for User Profile](#page28) [20](#page28)

[Table 3.7.4: Table For User-Visit-Category Profile](#page28) [20](#page28)

[Table 3.7.5: Table For Watch-List Profile](#page29) [21](#page29)

[Table 3.7.6: Table for Wish-List Profile](#page29) [21](#page29)

**1 Introduction**

System may be defined as a layered structure that depicts how programs involved would interrelate and communicate. In computers, System may also include actual programs, programming interfaces and tools for managing the larger system. The term system may be used differently in different contexts, but more or less the concept remains the same. Online student course registration system combines multiple systems to construct a combined framework. This framework consists of multiple modules, which further contain different systems along with the implementation of their defined constraints.

Basically, systems are implemented for facilitating complex manual processes and that is exactly what we are trying to achieve. System is implemented as per user requirement such as a manufacturing concern may install a plant for easing out manual processes. We have sought help from computer programming for automation of manual registration system. With the introduction of computers, every aspect of our lives has been revolutionized. When used judiciously, computers can help us save time, secure our personal information, access the required information whenever and wherever required. Keeping all these positive points in mind, we have developed an Online Student Course Registration System for easily managing the semester registration process for the student in an institution. Ours is an advisory based system. In state agricultural universities the course allocation is advisory based and more complicated. The courses are assigned according to the skill set and industry requirements. Hence, in current scenario, automated system is required for course registration of students.

***1.1*** ***Problem Statement***

For building a recommender system from scratch, we face several different problems. Currently there are a lot of recommender systems based on the user information, so what should we do if the website has not gotten enough users. After that, we will solve the representation of a movie, which is how a system can understand a movie. That is the precondition for comparing similarity between two movies. Movie features such as genre, actor and director are a way that can categorize movies. we get these questions:

* How to recommend movies when there are no user information.
* What kind of movie features can be used for the recommender system.
* How to calculate the similarity between two movies.

***1.2*** ***Goals***

* To design and implement an online registration system.
* To analyze the impact of an online registration system at Stamford University.

***1.3*** ***Purpose of the work***

The need for structured storage, modification, and maintenance of huge amounts of data has resulted in the emergence of the Database Management System (DBMS) as one of the core fields in the Computer Science industry. DBMS is the system of computer software that is aimed to provide a managing tool for maintaining the data, through various data models.

The purpose of implementing this project is to understand the data modeling concepts that is used in a real time scenario and to implement a fully functional database system which interacts with a front-end interface.

***1.4*** ***Objectives of the project***

The current research aims at reducing the workload all the entities involved in the registration procedure for the students. The current manual system faces different challenges as to maintaining data of each student manually. Hard copy registers are maintained currently to verify student details. From students’ point of view, they must fill the forms manually and then get them verified from concerned officials, which is a very time-consuming process. The objectives of this proposed web application system are:

* To computerize student and faculty database.
* To maintain data consistency and integrity.
* Automate the registration process without any physical human interaction
* Making the registration process accessible anywhere to the student.
* Allowing faculty to acknowledge registration requests from anywhere.

***1.5*** ***Scope of the project***

With the requirement of registration process for every semester, it becomes all the more important to simplify a process which is highly repetitive. The achievement of the above objectives can help the institution in managing the resources efficiently. The automated process will lead to time saving and eradication of common errors.

***1.6 Use-cases of Recommendation systems***

Recommendations are not a new concept. Even when e-commerce was not that prominent, the sales staff in retail stores recommended items to the customers for the purpose of upselling and cross-selling, and ultimately maximize profit. The aim of recommendation systems is just the same. Another objective of the recommendation system is to achieve student loyalty by providing flexibility of course registration. This also helps in increasing student engagement.

***1.7*** ***Importance of Recommendation Systems***

Recommendation systems help the student course registration on new semester.

***1.8*** ***What can be recommended?***

Lots of thing can be recommended student to save their time and priority of their choice.

***1.9*** ***Benefits of a course registration system***

* User-friendly interface.
* Responsive Design
* 24/7 hours availability
* Basic computer knowledge required
* The effective searching system included with categories.
* Effective and easy Registration Process.
* Secure login system.
* No cost for online service.
* Saves time, and energy.
* Rich Documentations.

**2 Previous Work**

There are several online registration systems that have been made in different context and to suit the need arising. Gunawardena et al in December 2008, researched on the online course registration for the faculty of engineering in university of Peradeniya and in their research, they generated a solution named Online Registration. The project was divided into three namely, Online Registration which is the start-up project of their system and depending on Data Access and Web security. Data Access would actuate the data transaction between the client and the server and Web Security would configure the database and handle the authentication and authorization. The research described their experience in designing, developing, and deploying an online course registration system at the Faculty of Engineering in University of Peradeniya. The system has not only reduced the burden of all parties involved in the course registration process, but also improved the process by reducing errors. Though their system managed to reduce the errors significantly it lacked the continued interaction with the student in the updates of the student’s registration status updates. It also lacked accountability as some stakeholders in the registration process do not have access to the system.

**3 System Methodology**

This proposed system is the web application that student can easily enroll new course on new semester. Our web application is design to helping student and university to help course registration mechanism.

***3.1*** ***Requirement***

1. Server
2. Resource (Such as course list, faculty list, student list)
3. A programming language like php

***3.2*** ***Design***

In our website we have home button, logging/registration button, users, students etc.

***3.3*** ***Implementation***

We have completed this project in many small parts. Our main goal is given to student a better system for registration. We have design our website part by part. Here some main topic how we complete the project.

1. Design: At first, we create a basic website using html, css and bootstrap.
2. Server Design: Then we create registration system server in MySQL. There are many tables under the server recommendation. Like students, courses, users etc.
3. Connection: Then we connect the recommendation server via php.

The proposed recommendation web application is design to manage

**Admin**

* + Admin login
* Add user
* Manage User

**Chairmen/Advisor**

* Login
* Offer Course

12

* Offer List
* Expired Offer List
* Student List
* Pending/Approved Registration List

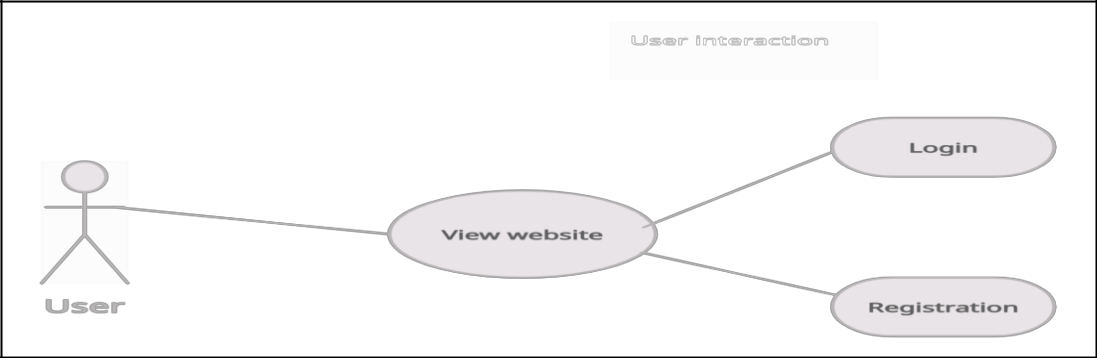
***3.4*** ***Requirement Analysis***

The requirement analysis process includes two types of diagram

* Use case diagram
* E-R diagram

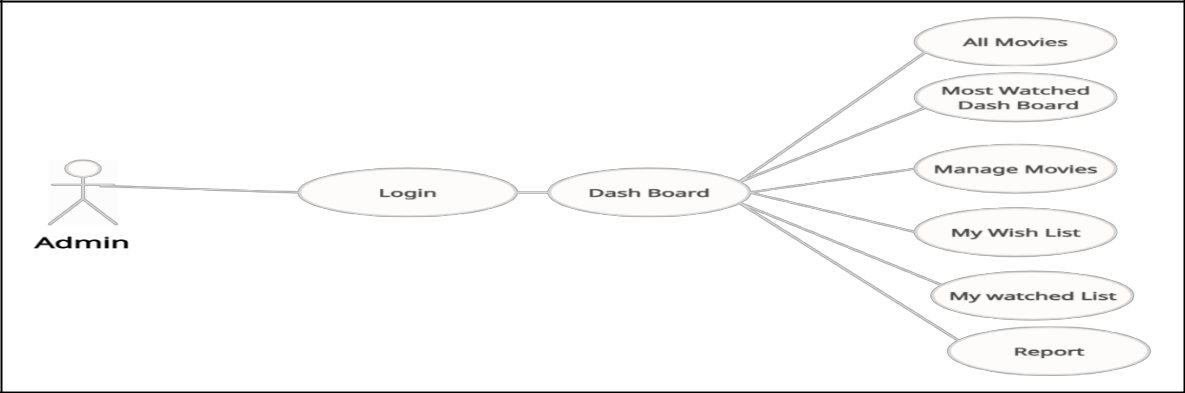
***3.5*** ***Use Case Diagram***

Use case diagram that connect between user and application



**Figure 3.5.1:** User Interaction

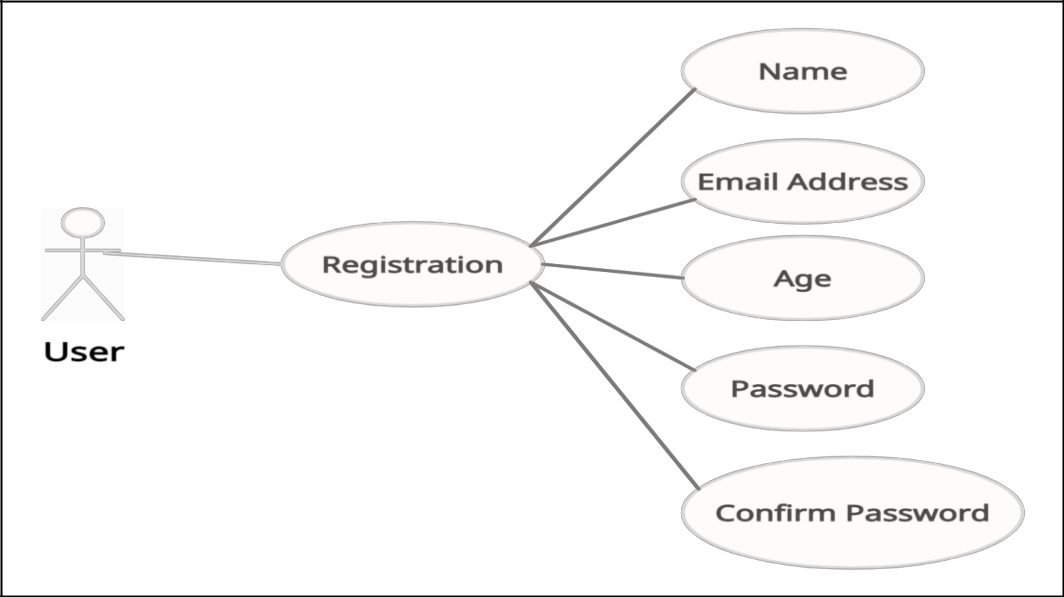
Admin panel overview



**Figure 3.5.2:** Admin Panel

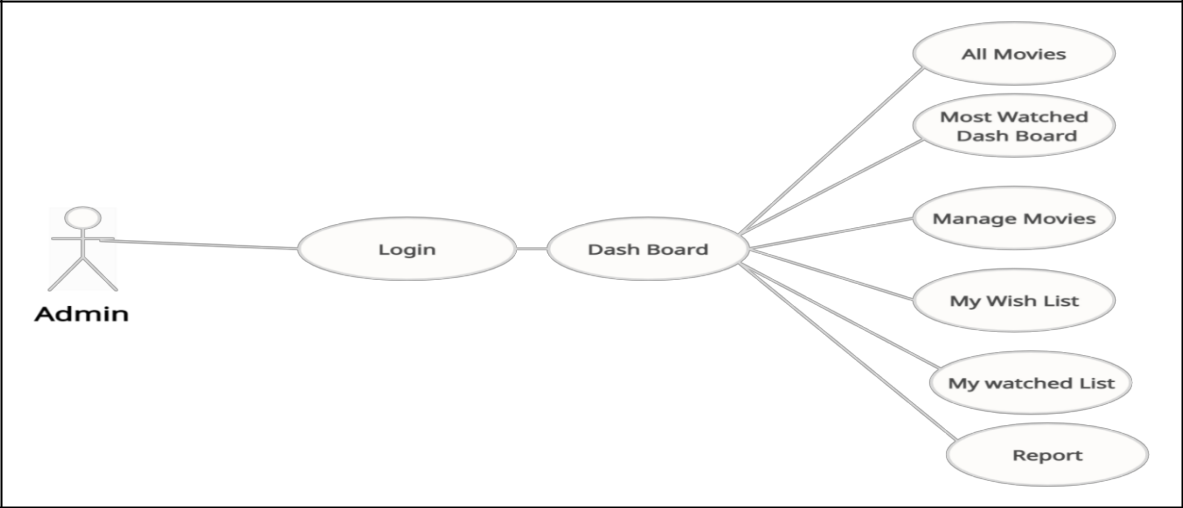
13

This is user registration profile. User create his/her name, Email, Age, Password and confirm password.



**Figure 3.5.3:** User Registration Panel

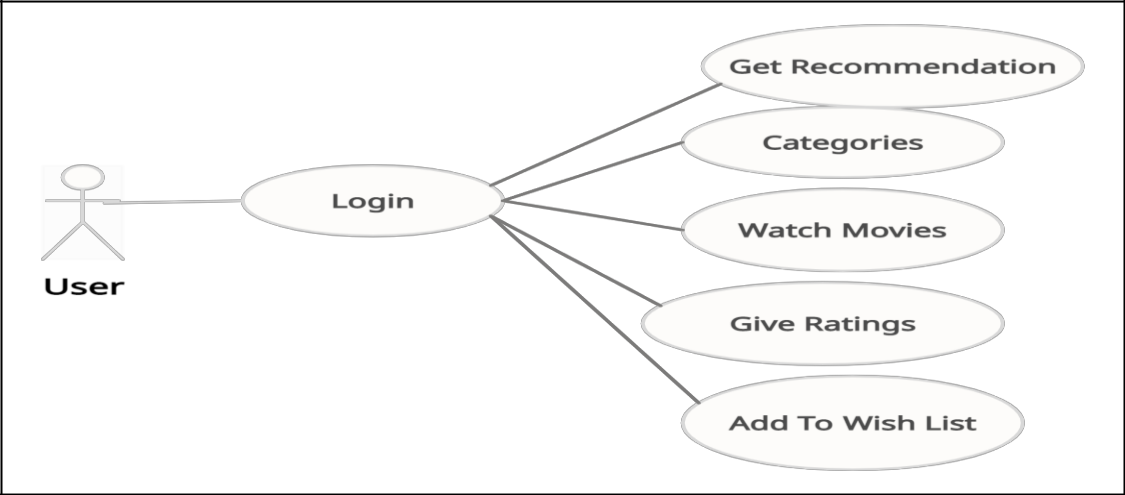
This is Admin registration profile. Admin create his/her name, Email, Age, Password and confirm password.



**Figure 3.5.4:** Admin Registration Panel

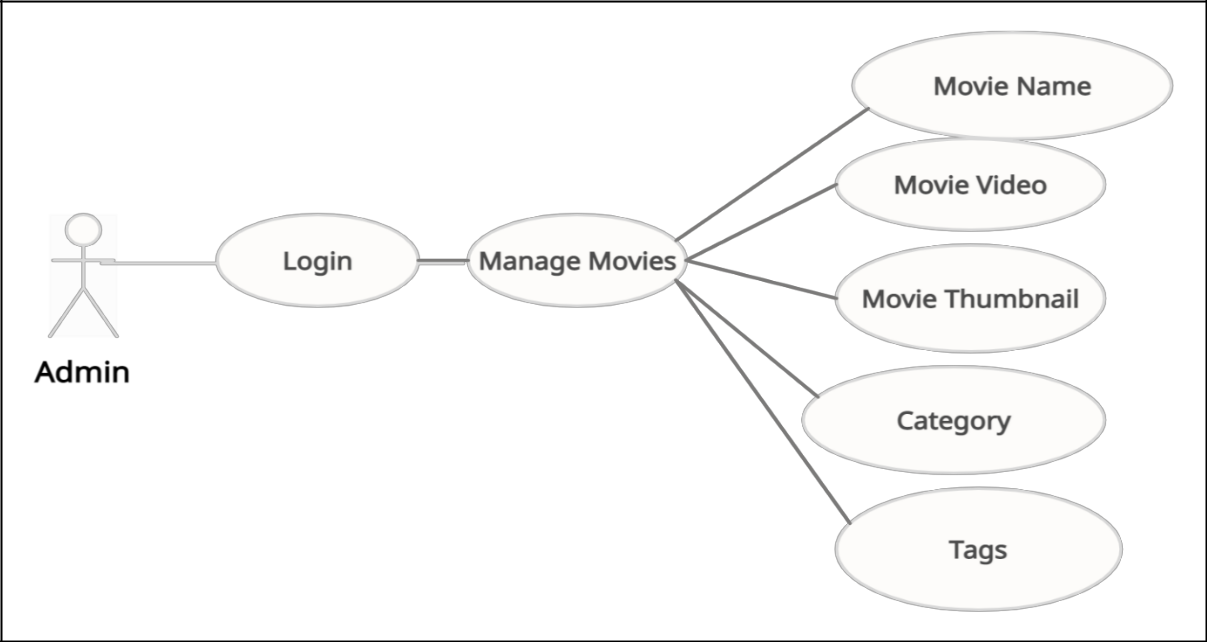
14

User watching movies



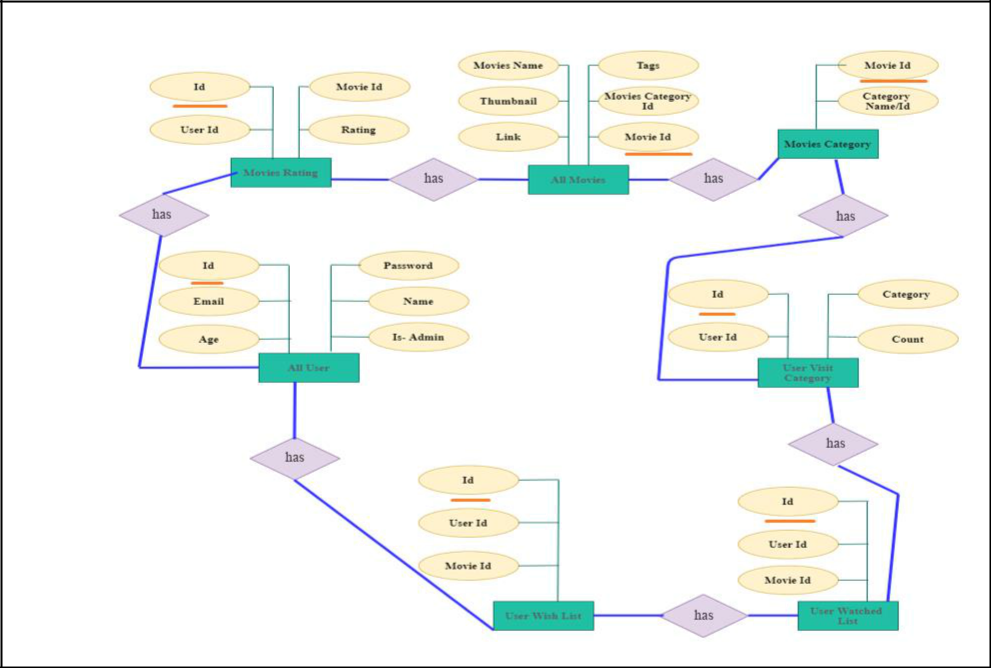
**Figure 3.5.5:** User Watching Diagram

Admin movie uploading process



**Figure 3.5.6:** Admin Movie Uploading Process

15



**Figure 3.5.7:** ER Diagram

***3.6*** ***Data flow Diagram***

We have three data flow diagram that describe how data passes.

Here its level 0 data flow diagram.

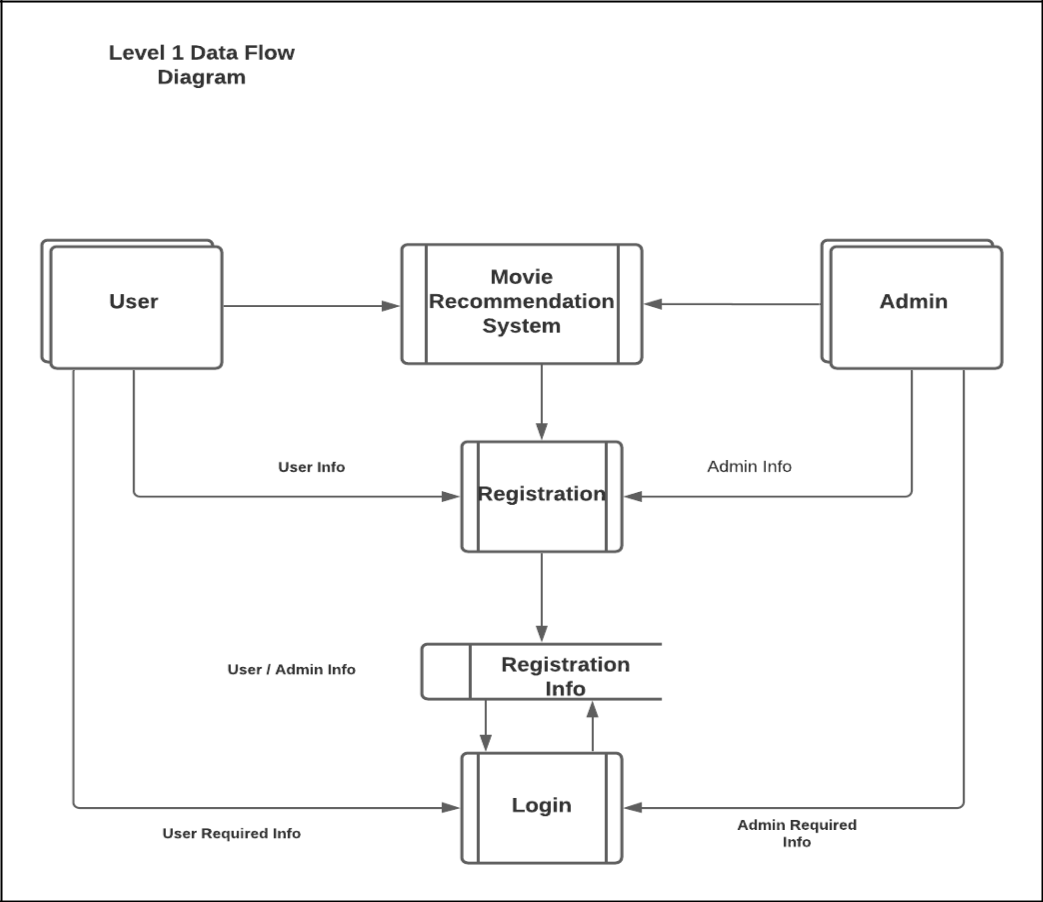


**Figure 3.6.1:** Level 0 Data Flow Diagram

We have two entity and a process. We have admin and user that’s are entity and the process name is Movie Recommendation System. Admin and user data flow through to the Movie Recommendation System.

16

Here is level 1 data flow diagram:

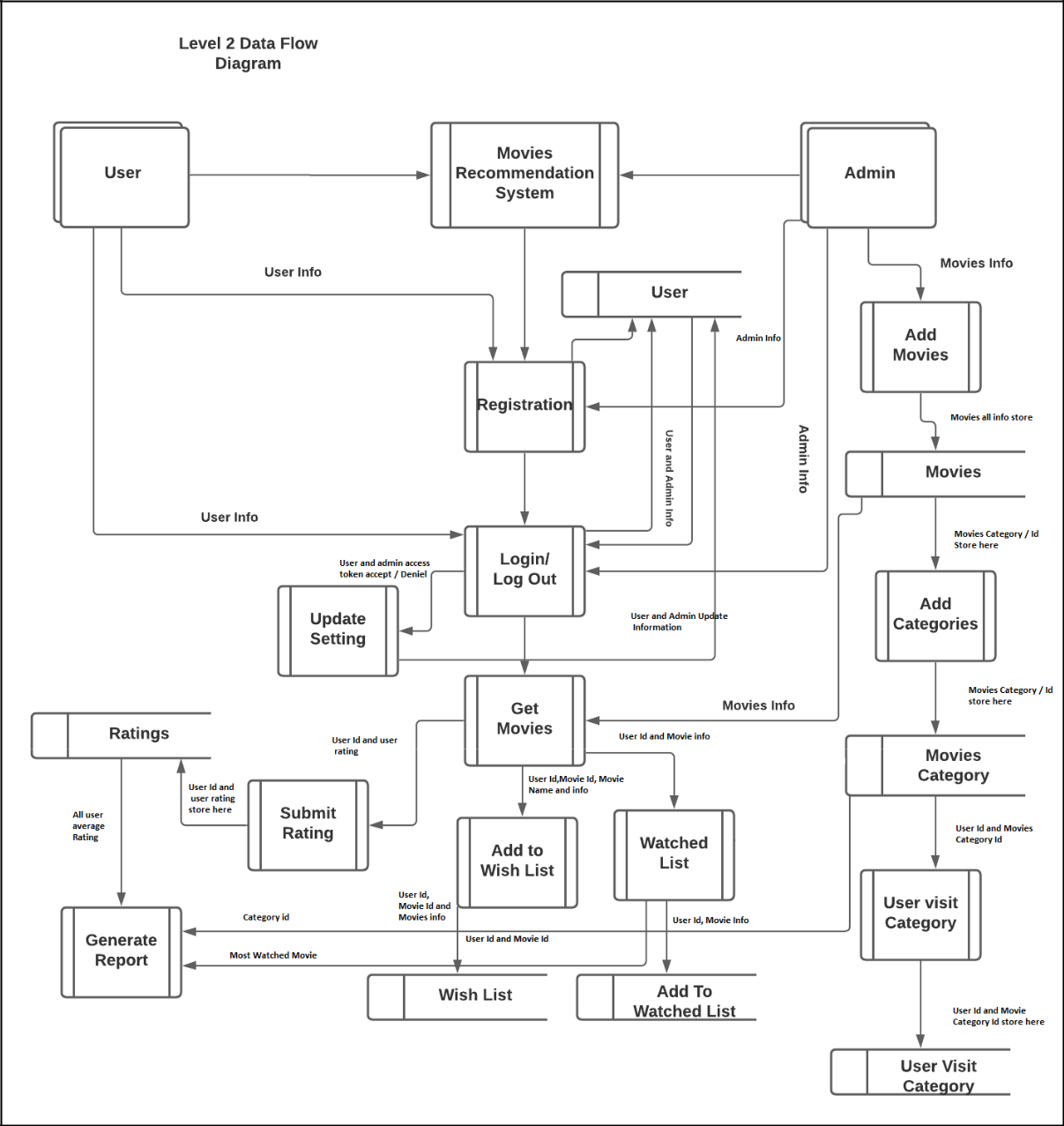


**Figure 3.6.2:** Level 1 Data Flow Diagram

We have two entities and we will breakdown our main process MRS. From MRS our data flow to the Registration process. Now User and Admin both give User and Admin info for registration process. Then successfully registration data go to the Registration Info. After completing registration process user and admin whenever they wish to login into the server thy would go through the login process. When user and admin wish to login into the server they give email and password to confirm their entry. This data go to the data store and match their provided data; If the given data is match then they login into the MRS successfully else system show invalid details. This process will happen every time when they wish to log in. We use this process to secure our server.

17

Here is level 2 data flow diagram:



**Figure 3.6.3:** Level 2 Data Flow Diagram

**User Process:**

After successfully login user can get movies, movies categories and recommendation via ‘’Get Movies’’ process. User get all movies info from ‘Movies’ table. Then User Can watch movies, Add to wish list and also can give rating the movies. User wish list is store in “wish list” table, Watched movies store in “Watched list” table and Rating store in “Rating table”. When a register user visit a movie our system also save movie categories in “Category” table.

18

**Admin Process:**

Admin can add many movies and all movies are save in “Movies” table. Admin also select Movies name, Movies Categories, Movies Thumbnail, Movies Name and Movies Genre.

***3.7*** ***All table with description:***

**Name of the table**

: Category Table

**Number of fields**

**:** 2

**Primary Key**

**:** id

**Table 3.7.1: Categories Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description |  | Type | Length |
| ID | Every movie has a category and every categories movie have a | | Integer | 11 |
|  | unique id |  |  |  |
| NAME | Every Category have a unique name. Example Horror, action. | | Text | 100 |
| **Name of the table** | | : Movies Table |  |  |
| **Number of fields** | | **:** 6 |  |  |
| **Primary Key** | | **:** id |  |  |

**Table 3.7.2: Table for Movies Profile**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Description | Type | Length |
|  |  |  |  |
| ID | Every movie has unique id. | Integer | 11 |
|  |  |  |  |
| NAME | Every movie must have a name. | Text | 100 |
|  |  |  |  |
| THUMBNAIL | Movie have a poster. | Varchar | 100 |
|  |  |  |  |
| LINK | Link to watch movie in website. | Varchar | 200 |
|  |  |  |  |
| TAGS | It is predefining text, that’s how user find a movie when | Text | 500 |
|  | they use search button |  |  |
|  |  |  |  |
| CATEGORY | Our system has 4 categories. So, every movie has a category | Integer | 11 |
| \_ID |  |  |  |
|  |  |  |  |

19

|  |  |
| --- | --- |
| **Name of the table** | : Rating Table |
| **Number of fields** | **:** 4 |
| **Primary Key** | **:** id |

**Table 3.7.3: Table for Rating Profile**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** |  | **Description** | | **Type** | **Length** |
| ID |  | It’s a primary key. | | Integer | 11 |
| User-ID |  | When a user log in. He/she get a user-id. | | Integer | 11 |
| Movie-ID |  | Movie also have id and that’s are unique id. | | Integer | 11 |
| Rating |  | After watching movie user give a rating to the movie and in | | Integer | 11 |
|  |  | this rating row save all the rating. | |  |  |
| **Name of the table** | | | : User |  |  |
| **Number of fields** | | | **:** 6 |  |  |
| **Primary Key** | |  | **:** id |  |  |

**Table 3.7.4: Table for User Profile**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Field | |  | Description | | Type | | Length | |
| ID | |  | User gets a unique id when they registration. | | Integer | | 11 |  |
| Email | |  | Email of a specific user by the using it user can log in. | | Varchar | | 100 |  |
| Age | |  | User age. | | Integer | | 11 |  |
| Password | |  | Password of a user. | | Varchar | | 100 |  |
| Name | |  | Name of a specific user | | Varchar | | 100 |  |
| IS-Admin | |  | Only for admin | | Tinyint | | 1 |  |
| **Name of the table** | | | | : User-Visit-Category |  |  |  |  |
| **Number of fields** | | | | **:** 4 |  |  |  |  |
| **Primary Key** | | | | **:** id |  |  |  |  |
|  |  |  |  | **Table 3.7.5: Table for User-Visit-Category Profile** | | |  |  |
|  |  | |  |  |  |  |  |  |
|  | Field | |  | Description | Type |  | Length |  |
|  | ID | |  | Movie id | Integer |  | 11 |  |
|  | User-ID | | | User id for collecting user movie watched list | Integer |  | 11 |  |
|  | Category- | | | For which category movie user watched. | Integer |  | 11 |  |
|  | Id | |  |  |  |  |  |  |
|  | Count | |  | How many times a user watched a movie. | Integer |  | 11 |  |
|  |  |  |  | 20 |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name of the table** | | : Watch-List | |  |  |
| **Number of fields** | | **:** 3 | |  |  |
| **Primary Key** | | **:** id | |  |  |
|  |  |  | **Table 3.7.6: Table for Watch-List Profile** | |  |
|  |  |  |  |  |  |
|  | Field |  | Description | Type | Length |
|  | ID |  | Its user id | Integer | 11 |
|  | User-ID |  | User id for saving the user watched list | Integer | 11 |
|  | Movie-ID |  | Its movie id | Integer | 11 |
| **Name of the table** | | : Wish-List | |  |  |
| **Number of fields** | | **:** 3 | |  |  |
| **Primary Key** | | : User**-** id | |  |  |

**Table 3.7.7: Table for Wish-List Profile**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Description | Type | Length |
| ID |  | Integer | 11 |
| User-ID | User id | Integer | 11 |
| Movie-ID | Movie id for saving the movie in user wish list. | Integer | 11 |

21

**4 Implementation**

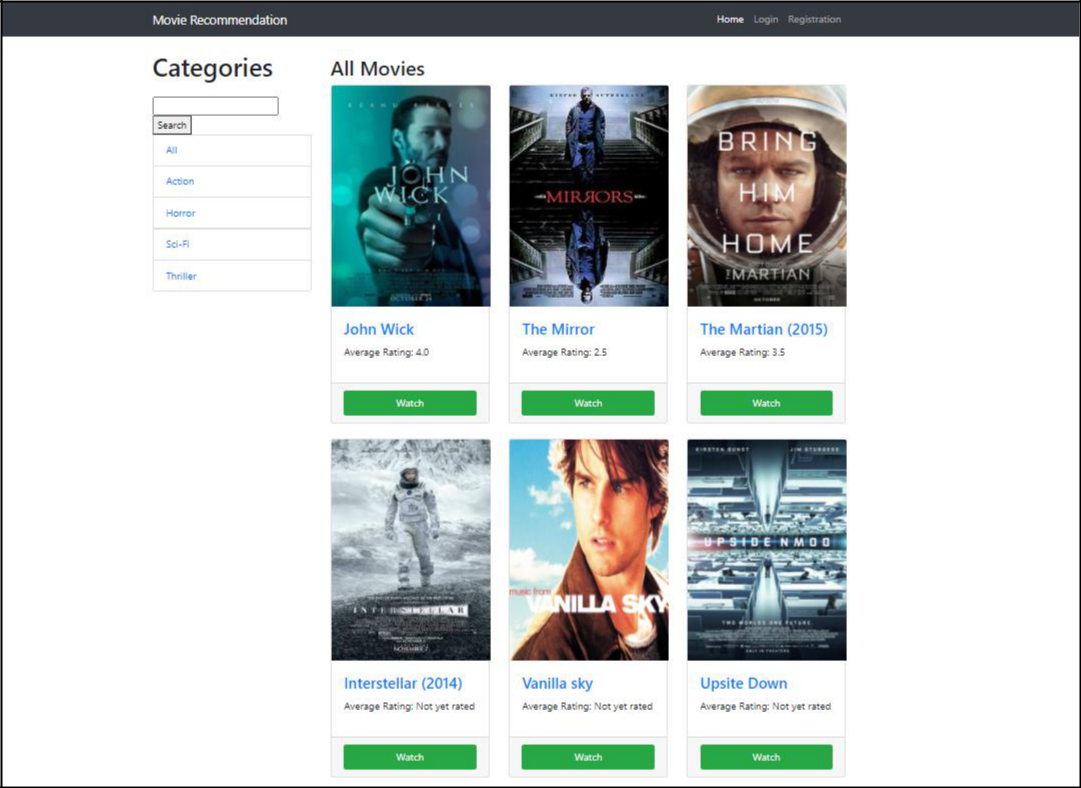
User interface (UI) design is the process designers use to build interfaces in software /web application or computerized devices, focusing on looks. Designers aim to create interfaces which users find easy to use and pleasurable. UI design refers to graphical user interfaces and other forms. Here we explain User interface and Backend process.

**User interface:**

***4.1*** ***Main Window Information***

*Propose:*

This page is used mainly user choose admin or user.



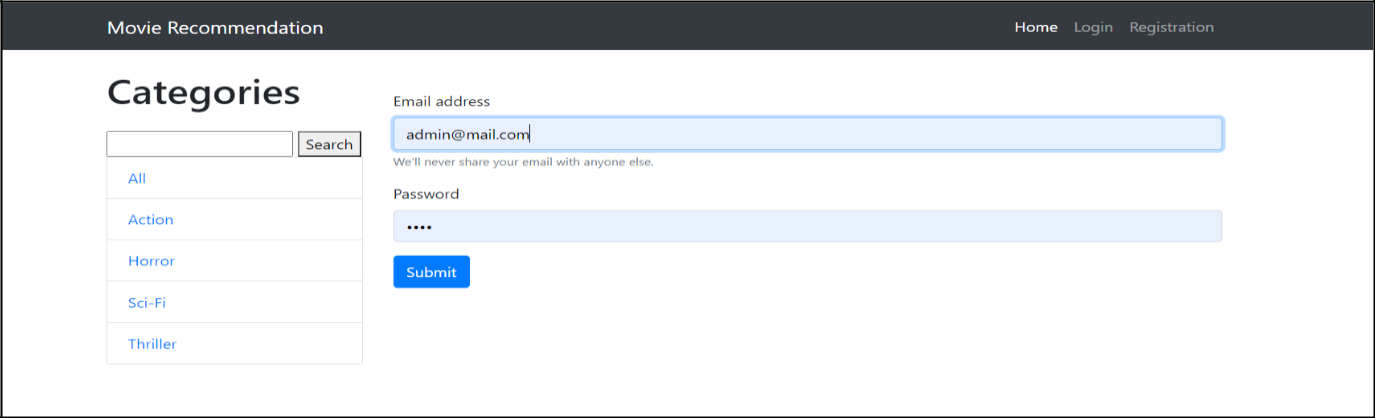
**Figure 4.1.1:** Main Window

22

***4.2*** ***Admin login Information***

*Propose:*

This page is used for admin login.



**Figure 4.2.1:** Admin Login Page

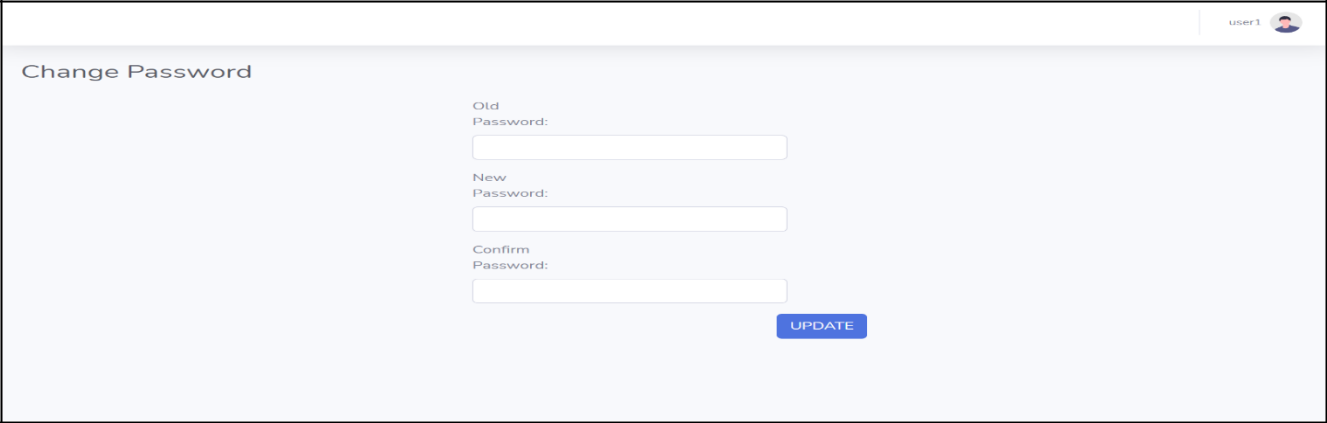
**User operation:**

By using this form, we can use admin account which is protected with password. If the username field is empty, it will return a message “Invalid Details”. When we give input to the username, password field and click on the “Submit” button it will match the details with previous data. If data is matched then admin sign in to the server.

**Admin change Password**

*Propose:*

This page is used for changing password if needed.



**Figure 4.2.2:** Admin Password Changing Page

23

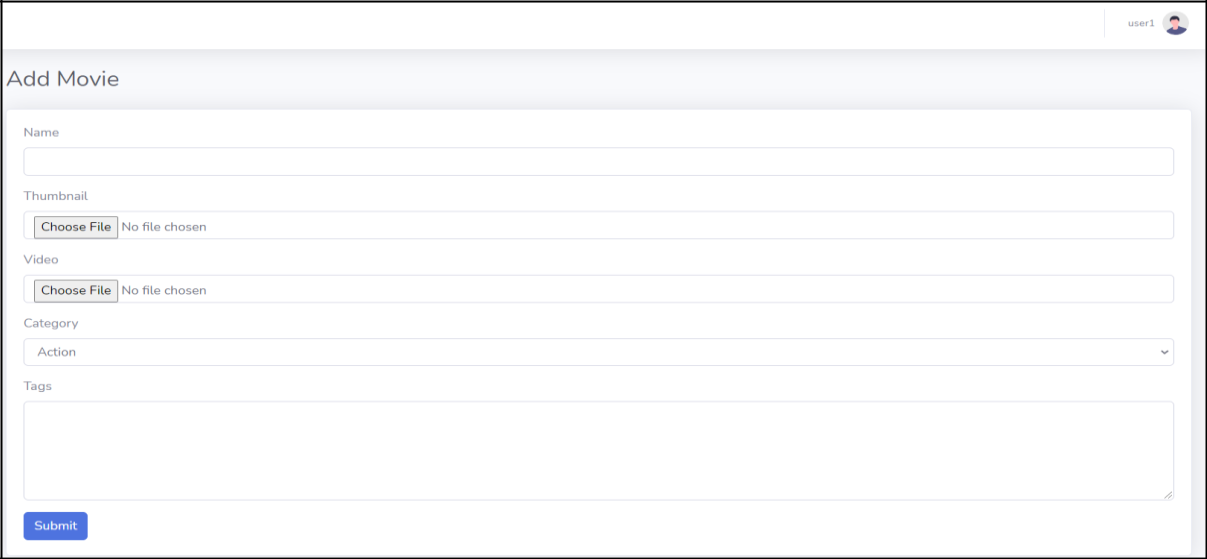
**User operation:**

1. Input the old password of an admin which is before use
2. Input the new password of an admin which is now use
3. Input again the confirm password of an admin which is before use in new password box.

***4.3 Admin movie upload form***

*Propose:*

This form is mainly use for upload a movie.



**Figure 4.3.1:** Admin Movie Upload From

**Backend Process:**

For User login we used “$\_Post” method. Because POST method does not have any restriction on data size to be sent. The POST method can be used to send ASCII as well as binary data.

Session is an alternative way to make data accessible across the various pages of an entire website is to use a PHP Session. A session creates a file in a temporary directory on the server where registered session variables and their values are stored. This data will be available to all pages on the site during that visit.

24

**5 Conclusion:**

Recommender system has become more and more important because of the information overload. For content-based recommender system specifically, we attempt to find a new way to improve the accuracy of the representative of the movie. In this project, we designed a new web recommender system for movie based on user behavior. The movies are a complex object and emotions are a human reaction and it is difficult to combining them together. In this paper, we tried to integrate movie recommendation by content-based filtering. This system provides better recommendations to users as it helps the users to understand the relation between their behavior and the recommendations.

***5.1 Future work:***

There are plenty of way to expand on the work done in this project. We tried to developed a movie recommended system for the user for their entertainment purpose only. In future we can use AI or deep learning for better understanding a user. Selecting the movies to finding most viewed using colors by system. Using more than two recommendation techniques to get best of the movies. Design a new algorithm to solving the movie recommender system. New movie upload notification to user. We can add a new movie added notification for user. User wish lists movie recommendation. We can many tags for finding movies. We will continue update our algorithms in our site. Like we can use hybrid-based recommendation. Nowadays is it very popular. We will use deep learning in our site to more informative. Deep learning is now very important for any recommendation type website.

***5.2 Limitations***

Doesn't take into account what others think of the item, so low quality item recommendations might be happen. Extracting data is not always easy. The content genera or tags can be incorrect or do not pick up enough detail. When we upload a movie, user don’t get a notification for new movies update. We have limited movie category and we could not add a movie in various category. We don’t have a foreign language movie category and don’t have web series category. Nowadays web series and foreign language and web series is very popular. We don’t have AI in our website, It can change user experience far better. We don’t have payment method system in our site. We don’t have Widevine L1 certification so mobile user can’t consume FHD content in our site.

25

**6 References**

1. A. Jain, "Movie Recommendation System — Content Filtering", Medium, 2019. [Online].

Available:https://medium.com/data-science-101/movie-recommendation-system-content-

filtering-7ba425ca0920. [Accessed: 08- Feb- 2020].

1. "Brief on Recommender Systems", *Medium*, 2021. [Online]. Available: https://towardsdatascience.com/brief-on-recommender-systems-b86a1068a4dd. [Accessed: 09- Mar- 2021].
2. "Recommendation System using K-Nearest Neighbors |Use Case in Python", *Analytics Vidhya*,2020.[Online].Available:https://www.analyticsvidhya.com/blog/2020/08/recommendation-system-k-nearest-neighbors/. [Accessed: 10- Sep- 2020].
3. F. Barros, "The Netflix Recommender System: Algorithms, Business Value, and

Innovation", *Academia.edu*, 2015. [Online]. Available: https://www.academia.edu/33391850/The\_Netflix\_Recommender\_System\_Algorithms\_Busi ness\_Value\_and\_Innovation. [Accessed: 11- Sep- 2020].

[5] A. Kumar, "YouTube's Recommendation Engine: Explained | Hacker

Noon", *Hackernoon.com*, 2020. [Online]. Available: https://hackernoon.com/youtubes-

recommendation-engine-explained-40j83183. [Accessed: 11- Sep- 2020].

1. J. Alvarez, "Hulu’s Recommendation System « Research « Chema's Home

Page", *Josemalvarez.es*, 2012. [Online]. Available: http://www.josemalvarez.es/web/2012/04/23/hulus-recommendation-system/. [Accessed: 12-Sep- 2020].

26