Learning Management System with Machine Learning Course Recommendation

Project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted By:

Ravi Prakash (20102134)

Mohammed Fayez Khan (20102148)

Submitted To:

Dr. Jasmine Saini

(Project Supervisor)

विद्या तत्व ज्योतिसमः

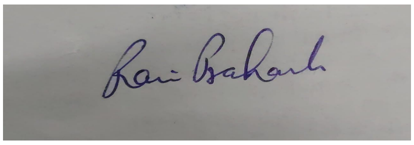
JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

SECTOR-62

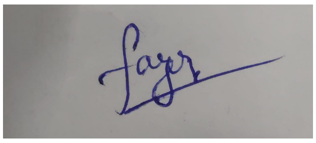
TABLE OF CONTENTS

CERTIFICATE

This is to certify that the work which is being presented in B. Tech Minor Project Report entitled "**Learning Management System with Machine Learning Course Recommendation**", submitted by "**Ravi Prakash & Mohammed Fayez Khan**", in partial fulfillment of the requirements for the award of degree of **Bachelor of Technology** in **Electronics & Communication Engineering** and submitted to the Department of Electronics & Communication Engineering of Jaypee Institute of Information Technology, Noida is an authentic record of our own work carried out during a period from July 2023 to December 2023 under the supervision of **"Dr. Jasmine Saini"**, ECE Department. The matter presented in this report has not been submitted by us for the award of any other degree elsewhere.

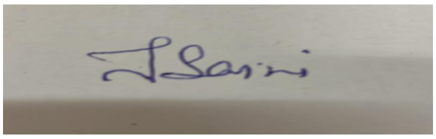


Ravi Prakash (20102134)



Mohammed Fayez Khan (20102148)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.



(Signature of Supervisor)

Dr. Jasmine Saini
ASSOCIATE PROFESSOR

Date: 30th November 2023

ACKNOWLEDGEMENT

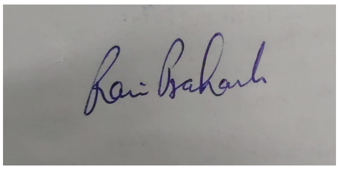
It gives us a great sense of pleasure to present the report of the Project Work that is "**Learning Management System with Machine Learning Course Recommendation"** undertaken during B. Tech Fourth Year. On the very outset of this report, I would like to extend our sincere and heartfelt obligation towards the ones who have helped me in this endeavor. Without their active guidance, help, cooperation & encouragement,we would not have made headway in the project.

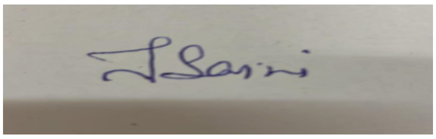
It is my great fortune that I have got opportunity to carry out this project work under the supervision of **Dr.Jasmine Saini**, in the Department of Electronics and Communication Engineering, Jaypee Institute of Information Technology (JIIT), A 10, A Block, Industrial Area, Sector 62, Noida, Uttar Pradesh - 201309. She would take some time out from his busy schedule and guide us to the right track. Time to time she would remind us about how important it is to write the report correctly. I express my sincere thanks and deepest sense of gratitude to my guide for her constant support, unparalleled guidance and limitless encouragement.

Last but not the least; we would acknowledge my teammate for their contribution in the completion of the project. Any omission in this brief acknowledgement does not mean lack of gratitude.

ABSTRACT

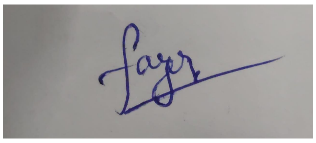
This project introduces an innovative Course Recommendation System, leveraging machine learning techniques to provide personalized and intelligent course suggestions for users. The system is seamlessly integrated into a web platform built with React and Next.js, ensuring a smooth and intuitive user experience. The recommendation system adopts a hybrid approach, combining collaborative filtering, which analyzes user behavior, with content-based filtering, which considers course attributes. This fusion results in a dynamic learning experience, tailoring course suggestions to individual preferences and prior knowledge. The web application features user-friendly functionalities, including streamlined registration, a comprehensive course catalog, and secure payment processing through Stripe. The project also integrates Mux for video content delivery, ensuring an immersive and uninterrupted learning environment. Despite the challenges faced in implementing skill assessment examinations, the system aims to empower users with tailored learning paths. The project lays the foundation for future enhancements, such as refining recommendation algorithms and leveraging additional data sources for improved personalization. Overall, this Course Recommendation System signifies a pioneering step towards reshaping online education, fostering adaptive and engaging learning journeys.





Ravi Prakash (20102134)

Dr. Jasmine Saini



Mohammed Fayez Khan (20102148)

Date: 30th November 2023

List of Figures

CHAPTER 1: INTRODUCTION

In the rapidly evolving landscape of online education, our college project endeavors to pioneer a transformative advancement by developing an innovative Learning Management System (LMS).

Central to our initiative is the integration of a cutting-edge machine learning-based course recommendation system. This Learning Management System, equipped with personalized course recommendations, represents a paradigm shift in the way users engage with educational content.

1.1 Significance of the Project:

Addressing Diverse Learning Needs:

Traditional Learning Management Systems often follow a one-size-fits-all approach, overlooking the diverse needs and preferences of individual learners. The project seeks to rectify this by implementing a personalized course recommendation system, ensuring that each user's unique learning journey is acknowledged and catered to.

Adaptive Learning Experience:

The core objective of the project is to create an adaptive learning experience that evolves with each user. By harnessing the capabilities of machine learning, our system tailors course recommendations based on individual preferences, prior knowledge, and learning styles. This adaptability ensures that users receive content that aligns precisely with their academic aspirations and goals.

Empowering Users:

The project is driven by the vision of empowering users to take charge of their educational paths. Through personalized course recommendations, learners can explore subjects of interest, discover new areas of study, and seamlessly progress through their academic journey with content that is both challenging and relevant.

Commitment to Modernising Education:

The project signifies our commitment to modernize e-education. It acknowledges that traditional educational approaches may not fully cater to individual needs. We aim to redefine how education is delivered and experienced, especially in the digital age.

Personalization and Empowerment:

Central to the project is the idea of personalization. We understand that every learner is unique. Our machine learning-based recommendation system offers personalised course recommendations based on skills, preferences, and aspirations, empowering learners to take control of their educational journey.

Adapting to the Digital Era:

In today's digital world, education should be flexible and accessible. The project combines technology and education to provide a learning experience that is adaptable and continuous, transcending physical boundaries and rigid schedules.

A Promise of Progress:

Our college project represents a promise of progress in the field of online education. We aim to create a more inclusive, effective, and engaging learning experience. By leveraging technology and educational expertise, we hope to empower learners from diverse backgrounds to achieve their full potential.

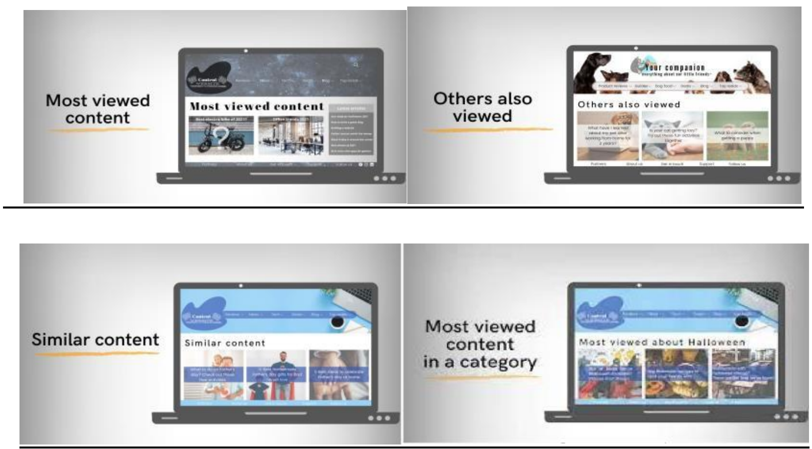


Fig 1.1 : Representation of a recommendation system

1.2 Key Features of the Learning Management System:

Advanced Course Recommendation System:

The heart of our LMS is a sophisticated machine learning-based course recommendation system. This system analyzes user behavior, preferences, and skill levels to provide tailored course suggestions, promoting a more engaging and effective learning experience.

User-Centric Design:

Our LMS is meticulously designed with the user in mind. The interface is intuitive, ensuring a seamless navigation experience. Users have the freedom to explore a comprehensive course catalog and receive personalized recommendations that align with their academic and career aspirations.

Dynamic Learning Paths:

Embracing the philosophy that learning is a dynamic and evolving process, our LMS facilitates the creation of personalized learning paths. Users can adapt their curriculum based on their progress, ensuring that the educational journey remains relevant and stimulating.

Data-Driven Insights:

The incorporation of data-driven insights allows users to track their learning progress, identify strengths and areas for improvement, and make informed decisions about their educational choices. This data-centric approach fosters a sense of ownership and accountability in the learning process.

In summary, the project seeks to modernize education by offering a personalized learning experience. It reflects our commitment to shaping the future of learning, where each learner can embark on a unique educational journey.

CHAPTER 2: BACKGROUND

In the field of education, there is a growing interest in technologies that support teaching and learning activities. For this purpose, ERS are strategic solutions to provide a personalized educational experience. Research in this sense has attracted the attention of the scientific community and there has been an effort to map and summarize different aspects of the field in the last 6 years.

2.1 Literature Review:

In Drachsler et al. (2015) :

A comprehensive review of technology enhanced learning recommender systems was carried out. Different aspects were analyzed about recommenders' approach, source of information and evaluation. Additionally, a categorization framework is presented and the study includes the classification of selected papers according to it. [8]

Klašnja-Milićević et al. (2015):

Conducted a review on recommendation systems for e-learning environments. The study focuses on requirements, challenges, (dis)advantages of techniques in the design of this type of ERS. An analysis on collaborative tagging systems and their integration in e-learning platform recommendersis also discussed. [9]

Ferreira et al. (2017):

Investigated particularities of research on ERS in Brazil. Papers published between 2012 and 2016in three Brazilian scientific vehicles were analyzed. [10]

Rivera et al. (2018):

Presented a big picture of the ERS area through a systematic mapping. The study covered a larger set of papers and aimed to detect global characteristics in ERS research. Aiming at the same focus, however, setting different questions and repositories combination, Pinho, Barwaldt, Espíndola, Torres, Pias, Topin, Borba and Oliveira (2019) performed a systematic review on ERS. In these works, it is observed the common concern of providing insights about the systems evaluation methods and the main techniques adopted in the recommendation process. [13]

Nascimento et al. (2017):

Carried out a SLR covering learning objects recommender systems based on the user's learning styles. Learning objects metadata standards, learning style theoretical models, e-learning systems used to provide recommendations and the techniques used by the ERS were investigated. [12]

Tarus et al (2018) and George and Lal (2019):

Concentrated their reviews on ontology-based ERS. Tarus et al. (2018) examined research distribution in a period from 2005 to 2014 according to their years of publication. Furthermore, the authors summarized the techniques, knowledge representation, ontology types and ontology representations covered in the papers. George and Lal (2019), in turn, update the contributions of Tarus et al. (2018), investigating papers published between 2010 and 2019. The authors also discuss how ontology-based ERS can be used to address recommender systems traditional issues, such as cold start problem and rating sparsity. [14]

Ashraf et al. (2021):

Directed their attention to investigate course recommendation systems. Through a comprehensive review, the study summarized the techniques and parameters used by this type of ERS. Additionally, a taxonomy of the factors taken into account in the course recommendation process was defined. Salazar et al. (2021), on the other hand, conducted a review on affectivity-based ERS. Authors presented a macro analysis, identifying the main authors and research trends, and summarized different recommender systems aspects, such as the techniques used in affectivity analysis, the source of affectivity data collection and how to model emotions. [7]

Khanal et al. (2019):

Reviewed e-learning recommendation systems based on machine learning algorithms. A total of 10 papers from two scientific vehicles and published between 2016 and 2018 were examined. The study focal point was to investigate four categories of recommenders: those based on collaborative filtering, content-based filtering, knowledge and a hybrid strategy. The dimensions analyzed were the machine learning algorithms used, the recommenders' evaluation process, inputs and outputs characterization and recommenders' challenges addressed. [11]

2.2 Common approaches used:

Content-Based Filtering:

One approach to course recommendation involves analyzing the content of courses and matching them to a student's profile. This method relies on features such as course descriptions, prerequisites,and keywords to make recommendations.

Collaborative Filtering:

Another common technique is collaborative filtering, which recommends courses based on the preferences and behaviors of similar students. This can be user-based, where recommendations come from similar students, or item-based, where courses similar to those previously taken by thestudent are suggested.

Hybrid Systems:

Many recommendation systems use a combination of content-based and collaborative filtering techniques to enhance the accuracy and coverage of recommendations. Hybrid systems aim to overcome the limitations of individual methods.

2.3 Issues to be Fixed:

Cold Start Problem:

One major issue is the "cold start"[14] problem, where the system struggles to provide accurate recommendations for new students who have limited or no historical data. Conventional systems may face challenges in suggesting relevant courses for these users.

Limited Diversity:

Some recommendation systems may lack diversity in their suggestions, leading to a narrow range of courses being recommended. This can result in a limited exploration of different academic areas and may not support a holistic educational experience.

Dynamic Changes:

Conventional systems might not adapt well to dynamic changes in a student's academic or personal interests. As a student progresses through their academic journey, their goals and preferences may evolve, requiring the recommendation system to be flexible and responsive.

Data Privacy Concerns:

Course recommendation systems often rely on collecting and analyzing sensitive student data. Ensuring the privacy and security of this data is crucial, and conventional systems may face challenges in addressing privacy concerns.

Scalability:

As the number of courses and students increases, scalability becomes a concern. Conventional recommendation systems may struggle to efficiently handle large datasets, impacting the speedand performance of the recommendation process.

User Engagement and Explainability:

Understanding why a specific course is recommended is essential for user trust. Conventional systems may not always provide transparent and easily understandable explanations for their recommendations, leading to lower user engagement.

Lack of Real-Time Adaptation:

Traditional recommendation systems may not be designed to adapt in real-time to sudden changes in a student's circumstances, such as new academic interests or shifts in personal goals.

2.4 Problem statement:

To overcome the lack of personalization in course recommendations, I propose the development and integration of a personalized recommendation system that incorporates a specialized test. This test will be designed to assess the student's prior knowledge and proficiency in various subject areas.

2.4.1 Implementation:

The project is an intersection of modern web development technologies and the cutting-edge capabilities of machine learning. It's about taking the traditional LMS concept and infusing it with AI-powered insights to provide a truly personalized learning journey. Users will discover a comprehensive LMS that doesn't just offer courses but empowers them with intelligent recommendations derived from their skills and preferences.

To bring the power of machine learning into our LMS, we'll deploy a hybrid recommendation system directly into the website's architecture. This model leverages both collaborative and content-based filtering techniques to refine course suggestions. Collaborative filtering draws insights from user behavior and preferences, while content-based filtering considers course attributes. This hybrid approach ensures that our recommendations are not only personalized but also enriched with a deep understanding of both user behavior and course content. This intelligent system will seamlessly integrate into the LMS, continually enhancing the learning experience for our users.

User Registration: The educational journey commences with user registration, during which basic information is collected. [2]

Course Catalog: A comprehensive course catalog empowers users to

Explore and select courses that align with their interests and aspirations. [2],[6]

Payment Processing: Stripe ensures the secure and efficient handling of payments, assuring users of the safety of their financial information. [2]

Skill Testing Exam: Users undergo a skill assessment examination, the results of which serveas the cornerstone for generating highly personalized course recommendations. [6]

Recommendation System: The hybrid recommendation system combines collaborative filtering, considering user behavior, and content-based filtering, considering course attributes, to fine-tune course suggestions, resulting in an intelligent and dynamic learning experience. [2],[4],[6]

Video Content Delivery*:* Mux is seamlessly integrated to facilitate uninterrupted streaming ofvideocontent, ensuring a rich and immersive learning experience. [2],[5],[6]

2.5 Benefits of the Personalized Recommendation System:

Tailored Learning Experience:

Students receive course recommendations that align with their existing knowledge, providing a more personalized and effective learning experience.

Efficient Skill Development:

The system facilitates targeted skill development by recommending courses that bridge the gap between a student's current proficiency and higher levels of understanding.

Improved Student Engagement:

Personalized recommendations enhance student engagement by offering courses that are challenging yet achievable, fostering a sense of accomplishment.

Adaptability to Changing Needs:

The system's real-time adaptation ensures that recommendations stay relevant as a student's academic interests and understanding evolve.

In conclusion, the introduction of a personalized recommendation system, incorporating a customized test for proficiency assessment, addresses the inherent issue of lack of personalization in conventional course recommendation systems. This innovative approach not only enhances the learning experience for students but also contributes to a more dynamic and adaptive educational ecosystem.

CHAPTER 3: PROJECT DEVELOPMENT

3.1 Methodology Explanation:

User Registration:

This initial step involves users registering on the platform by providing basic information. This information could include details such as name, email, and possibly academic background. User registration is crucial for creating personalized profiles and understanding individual preferences. [2]

Course Catalog:

A comprehensive course catalog is made available to users, allowing them to explore and select courses based on their interests and aspirations. This catalog serves as the foundation for the recommendation system, offering a diverse range of courses for users to choose from. [2], [6]

Payment Processing (Stripe):

To ensure a secure and efficient financial transaction process, the system integrates Stripe for payment processing. Stripe provides a trusted and reliable platform for handling payments, assuring users of the safety of their financial information. This step is crucial for users who wish to enroll in paid courses. [2]

Recommendation System (Hybrid):

The recommendation system employs a hybrid approach, combining collaborative filtering and content- based filtering. Collaborative filtering considers the behavior and preferences of similar users, while content-based filtering takes into account the attributes of courses. This hybrid model fine-tunes course suggestions, resulting in an intelligent and dynamic learning experience. It ensures that recommendations are not only based on user behavior but also align with the content.[2],[4],[6]

3.2 Technologies / Software used :

Frontend Framework : Next.js and React

1. **React**:

Description:

React is a JavaScript library for building user interfaces, developed by Facebook. It allows developers to create interactive and dynamic user interfaces efficiently. React operates on a component-based architecture, making it easy to manage and update different parts of an application independently. It's widely used for single-page applications where user interactions play a significant role.

Key Features:

Virtual DOM: React uses a virtual DOM to efficiently update the user interface, improving performance by minimizing unnecessary re-rendering.

Reusable Components: React components are modular and can be reused across different parts of theapplication, promoting code efficiency and maintainability.

Declarative Syntax: The declarative approach makes it easier to understand and predict how the UIwill behave, enhancing developer productivity.

2. Next.js:

Description:

Next.js is a React framework that simplifies the development of server-rendered React applications. It adds functionality like server-side rendering and routing to React applications, making it easier to build scalable and performant web applications. Next.js is known for its ease of use, allowing developers to focus on building features rather than configuring complex setups.

Key Features:

Server-Side Rendering (SSR): Next.js enables SSR, which improves page load times by renderingpages on the server before sending them to the client.

Automatic Code Splitting: The framework automatically splits the JavaScript code into smaller chunks, facilitating faster loading times for users.

Built-in Routing: Next.js comes with a built-in routing system that simplifies navigation within theapplication.

3. Payment Prowess: Stripe

Description:

Stripe is a widely used and trusted platform for online payment. It provides a set of APIs and tools enabling businesses to securely accept and manage payments over the internet. Stripe is known for its simplicity, developer-friendly documentation, and robust security measures.[2]

Key Features:

Secure Transactions: Stripe prioritizes the security of financial transactions, using encryption andother security protocols to protect sensitive payment information.

Developer-Friendly: Stripe offers well-documented APIs and libraries for various programming languages, making it easy for developers to integrate payment processing into their applications.

Scalability: Stripe is designed to scale with businesses of all sizes, from startups to large enterprises,providing a reliable and flexible payment infrastructure.

Customizable Checkout: Businesses can customize the checkout experience to align with their brand,providing a seamless and consistent user experience.

Integration Rationale:

- React's component-based architecture aligns well with the modular and reusable design principlesrequired for a dynamic user interface.
- Next.js simplifies server-side rendering and routing, enhancing the performance and scalability of theapplication.
- Stripe ensures the secure and efficient handling of payments, instilling trust in users regarding the

safetyof their financial information.

- Stripe's developer-friendly approach streamlines the integration of payment processing into the application, saving development time and effort.

4. Prisma:

Description:

Prisma is an open-source database toolkit that simplifies database access and management through a type-safe and auto-generated query builder. It supports multiple databases, including MySQL, PostgreSQL, and SQLite. Prisma provides a modern and developer-friendly approach to database interactions, making it easier to work with databases in application development.

Key Features:

Type-Safe Queries: Prisma generates type-safe query builders based on the database schema, reducingthe risk of runtime errors related to database interactions.
Auto-Generated Migrations: Prisma facilitates database schema migrations by automaticallygenerating migration scripts based on changes in the application's data model.
Real-Time Data Sync: Prisma supports real-time data synchronization, allowing applications to receiveinstant updates when changes occur in the database.
Declarative Data Modeling: Prisma uses a declarative approach to define data models, making itintuitive for developers to represent and interact with database entities.

Description:

Mux is a platform that provides infrastructure and tools for video streaming. It is designed to simplify the process of delivering high-quality video content over the internet. Mux offers features such as video encoding, streaming, and analytics, making it a comprehensive solution for businesses and developers aiming to deliver seamless and engaging video experiences.

Key Features:

Video Encoding: Mux supports efficient video encoding, converting video files into formats suitablefor streaming while maintaining high quality.
Adaptive Streaming: Mux enables adaptive streaming, allowing the delivery of video content at different quality levels based on the viewer's internet connection, ensuring a smooth playback experience.
Real-Time Analytics: Mux provides analytics tools that offer insights into video performance, viewer engagement, and other relevant metrics, helping content providers optimize their video delivery strategies.
Live Streaming: Mux supports live streaming, making it suitable for various use cases, including live events, webinars, and interactive content.

Integration Rationale:

- Mux's seamless integration ensures that the platform can efficiently deliver video content to users, enhancing the overall learning experience.
- Features like adaptive streaming and real-time analytics contribute to a rich and immersive videodelivery process, allowing for a personalized and responsive viewing experience.

5. MySQL:

Description:

MySQL is a widely used open-source relational database management system (RDBMS). It is known for its reliability, performance, and scalability. MySQL is suitable for various applications, ranging from small projects to large-scale enterprise systems.

Key Features:

ACID Compliance: MySQL adheres to the ACID (Atomicity, Consistency, Isolation, Durability)properties, ensuring the reliability and integrity of transactions.
Scalability: MySQL can handle large datasets and is scalable to accommodate the growth of applications over time.

Community Support: As an open-source database, MySQL has a large and active community,providing support, resources, and a wide range of plugins and extensions.
Security: MySQL offers robust security features, including user authentication, encryption, andaccess control, to protect sensitive data.

Integration Rationale:

- Prisma, paired with MySQL, forms a powerful combination for efficient and type-safe database interactions in the application.
- The use of MySQL ensures data reliability, scalability, and security, while Prisma's developer-friendly features simplify the process of interacting with the database, reducing the complexity of data management in the application.

Hybrid Recommendation System - Collaborative and Content-Based Filtering

1. Collaborative Filtering:

Definition:

Collaborative filtering is a recommendation technique that relies on the preferences and behaviors of similar users. The underlying assumption is that users who have similar preferences in the past willcontinue to have similar preferences in the future. The system identifies patterns among users andrecommends items based on what similar users have liked or interacted with.[2],[4]

Working Mechanism:

The system builds a user-item matrix, where each cell represents the interaction (e.g., ratings) between a user and an item (course). It then identifies users with similar preferences through techniques such as user-based or item-based collaborative filtering. For a user with a limited history, the system recommends courses based on what similar users have liked. For a user with an extensive history, the system identifies patterns in their behavior to make personalized recommendations.

Advantages:

Collaborative filtering is effective in capturing user preferences without relying on explicit knowledge of item features.
It can recommend items that are popular among similar users but not necessarily known to target users.

2. Content-Based Filtering:

Definition:

Content-based filtering recommends items based on their features and the user's preferences. In the context of courses, this involves analyzing the attributes of courses (e.g., subject, difficulty level, topics covered) and matching them to the user's profile.[1],[2]

Working Mechanism:

The system creates a profile for each user and a representation of each course based on its features. It then recommends courses that match the user's profile and preferences. This approach is particularly useful when a user has a well-defined set of preferences or when explicit item features are crucial.

Advantages:

Content-based filtering can make accurate recommendations for new users (cold start problem) basedon their profile and preferences.
It allows for a level of personalization that takes into account specific attributes of items that auser may be interested in.

Hybrid Recommendation System:

Integration of Collaborative and Content-Based Filtering:

The hybrid recommendation system combines the strengths of both collaborative and content-based filtering to provide more accurate and diverse recommendations. It addresses the limitations of individual methods and enhances the overall recommendation quality.[2]

Workflow:

The system begins by using collaborative filtering to identify users with similar preferences. It then leverages content-based filtering to fine-tune recommendations based on the specific attributes of courses that align with the user's profile.

The collaborative filtering component helps overcome the cold start problem and captures user preferences, while the content-based filtering component adds a layer of personalization by considering the characteristics of courses.

Benefits:

Increased Accuracy: By leveraging both collaborative and content-based approaches, the system canprovide more accurate and personalized recommendations.

Diversity: The hybrid model ensures a broader range of course suggestions, addressing the risk ofrecommending only popular or similar items.

Robustness: It enhances the robustness of the recommendation system by mitigating the limitationsassociated with individual methods, such as the cold start problem or the lack of item features.

Fig 3.2.1 : Stacks Used (React, Mux, Stripe, Prisma)

Conclusion:

The hybrid recommendation system, combining collaborative and content-based filtering, is a powerful approach to address the challenges of conventional recommendation systems. By intelligently fusing user preferences and item attributes, the system can offer precisely tailored course suggestions, creating a more personalized and effective learning experience for users.

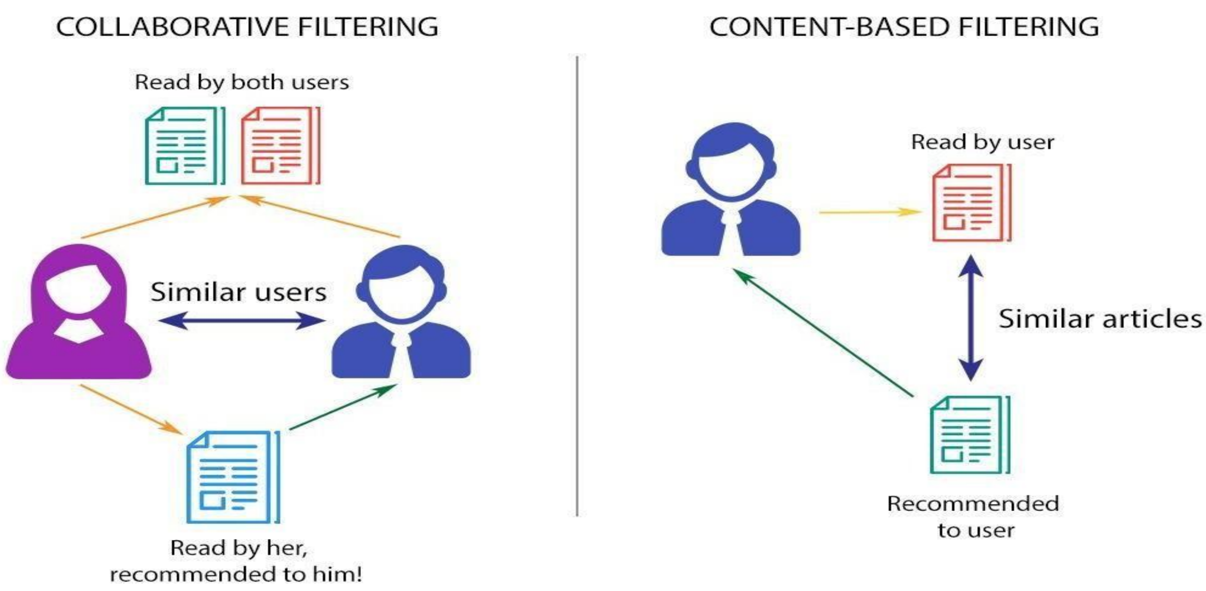


Fig 3.2.2 : Filtering Techniques

3.3 Flowchart of the Machine Learning process :

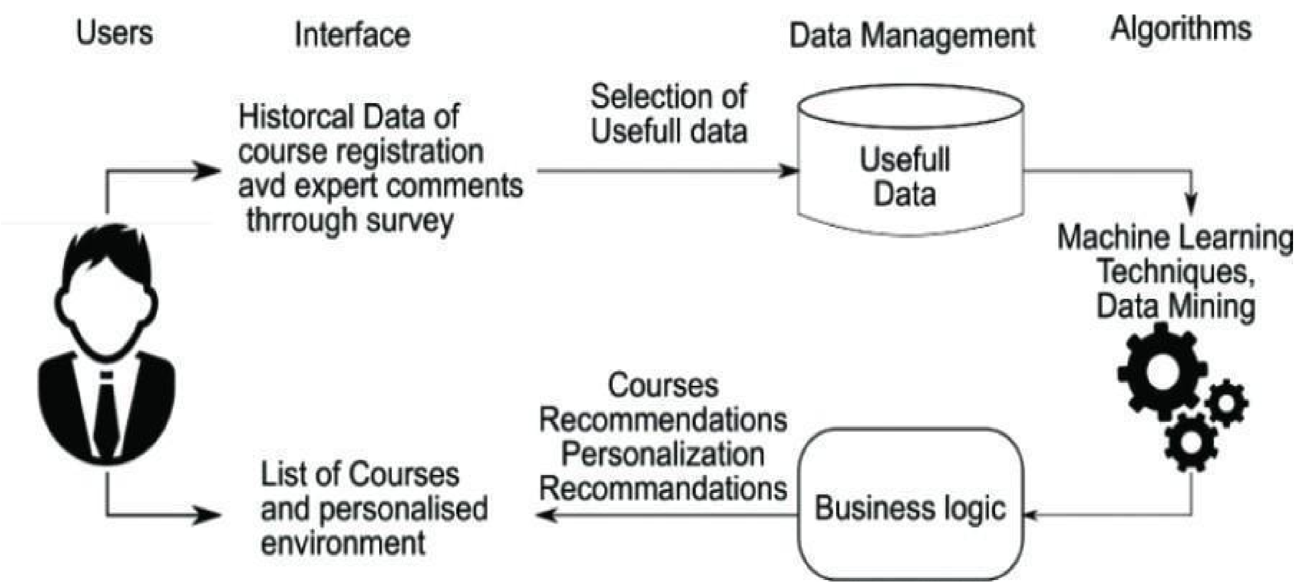


Fig 3.3.1 : Flowchart of the machine learning process

CHAPTER 4 : IMPLEMENTATION

4.1 Web Page Implementation

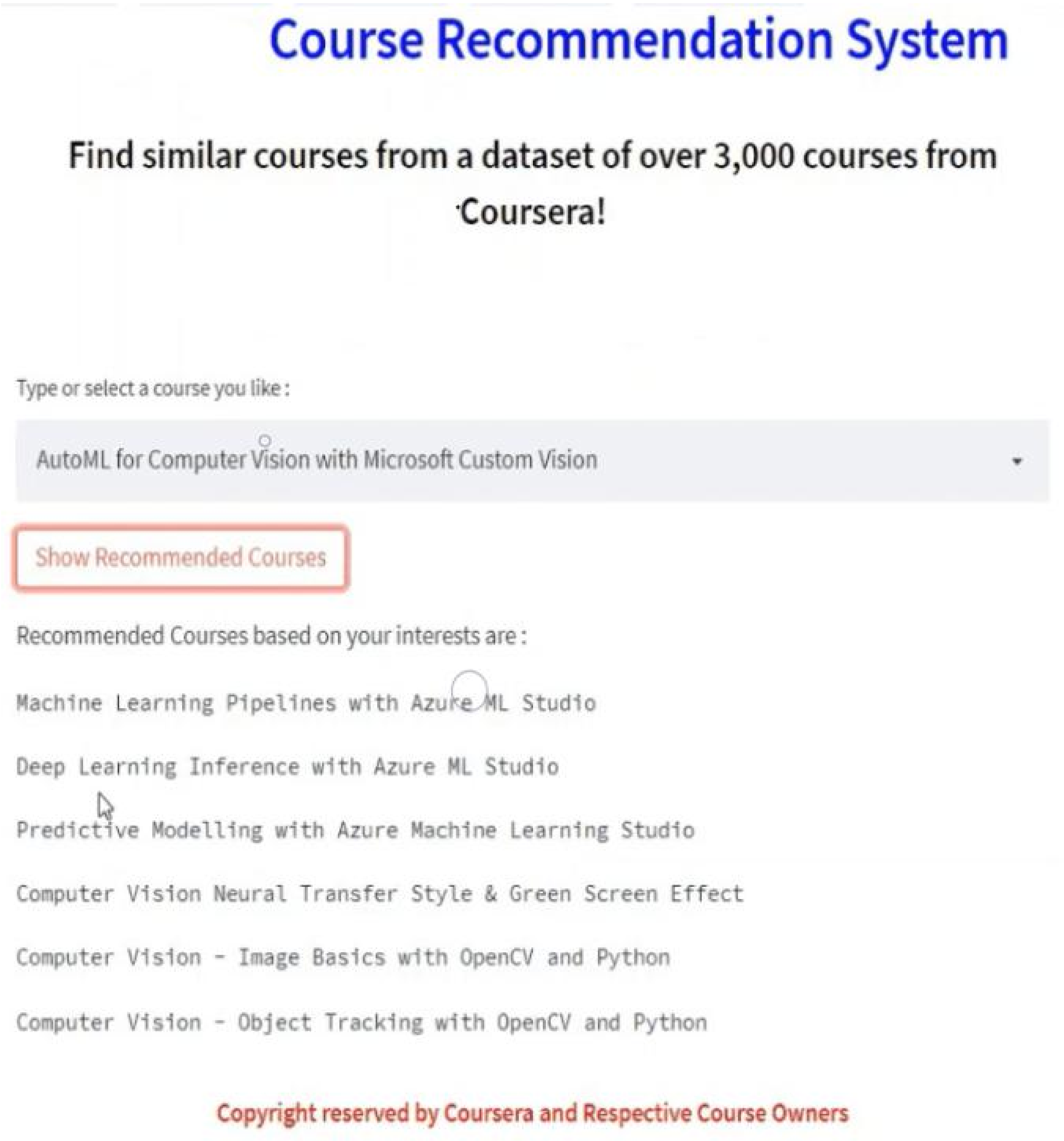


Fig 4.1.1: Similar Courses

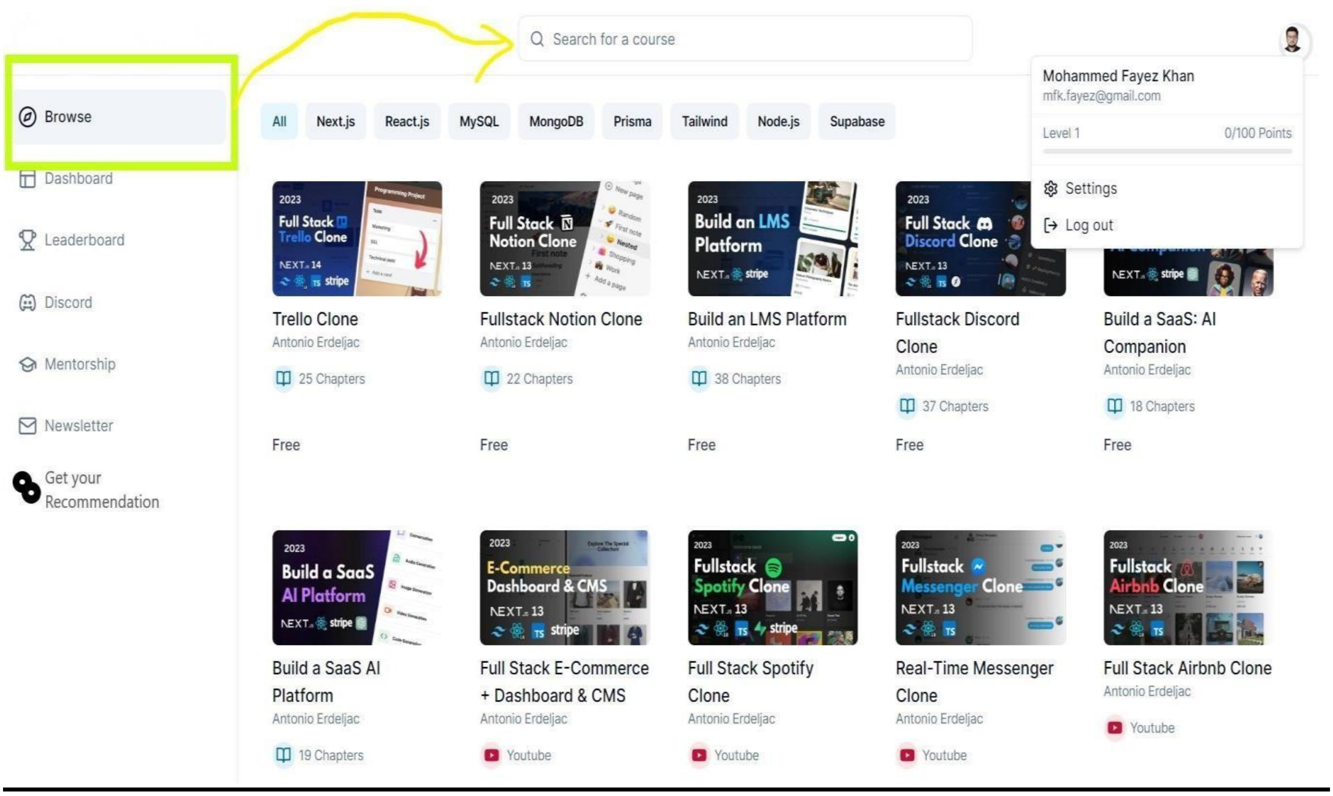


Fig 4.1.2: Search Bar

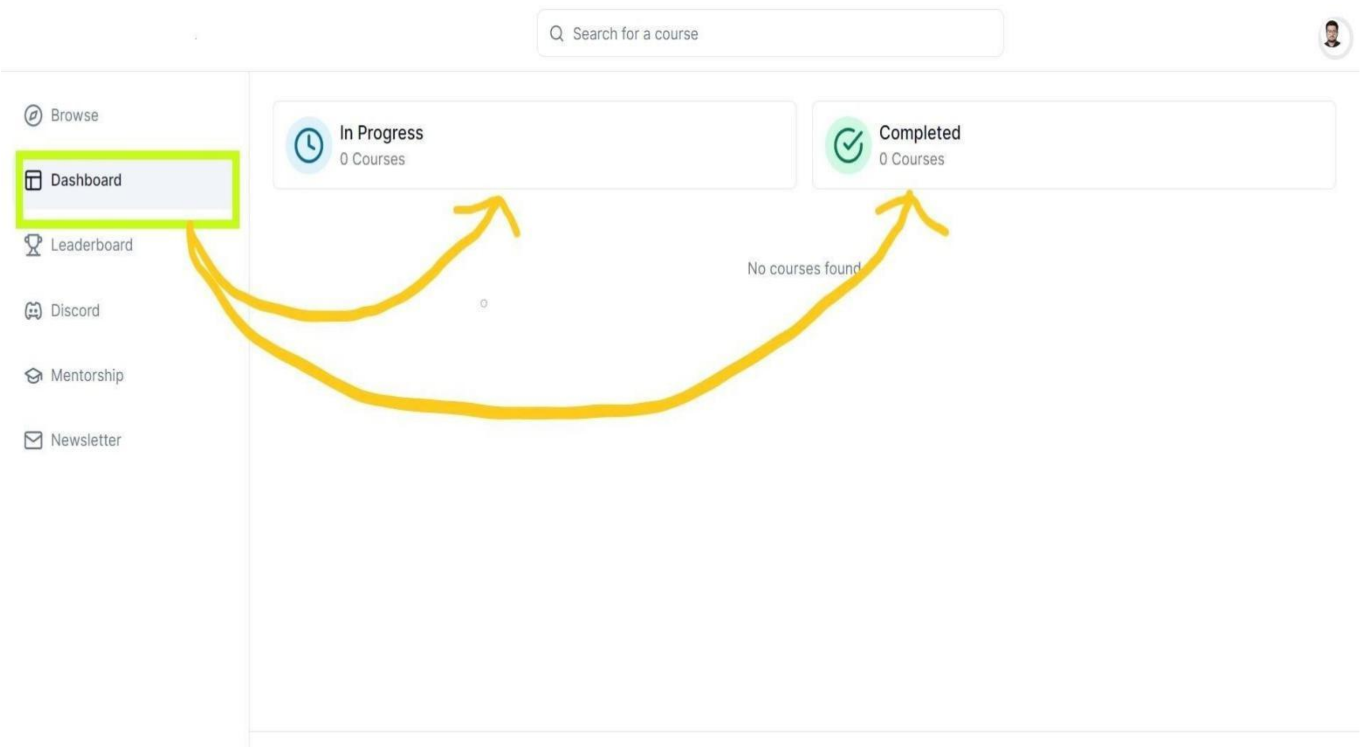


Fig 4.1.3: Dashboard

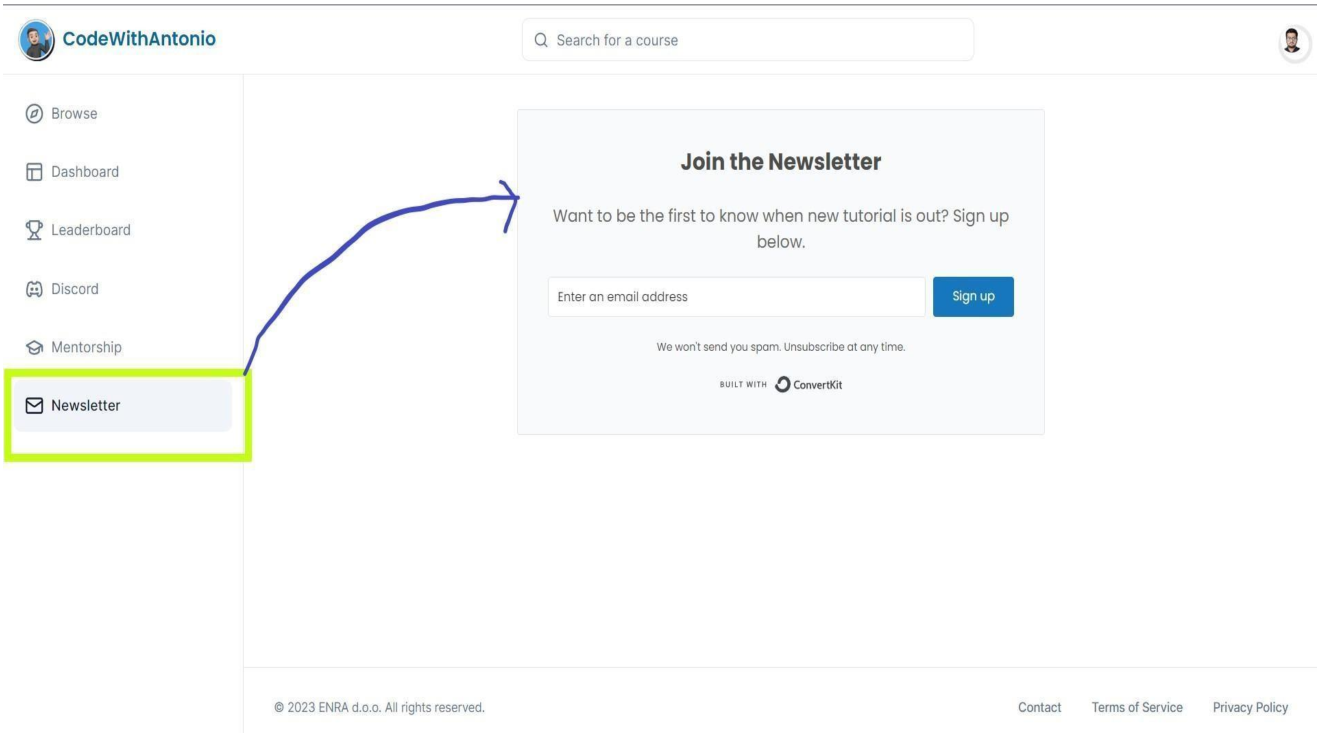


Fig 4.1.4: Newsletter

4.2 Machine Learning Implementation

Basic Data Analysis :

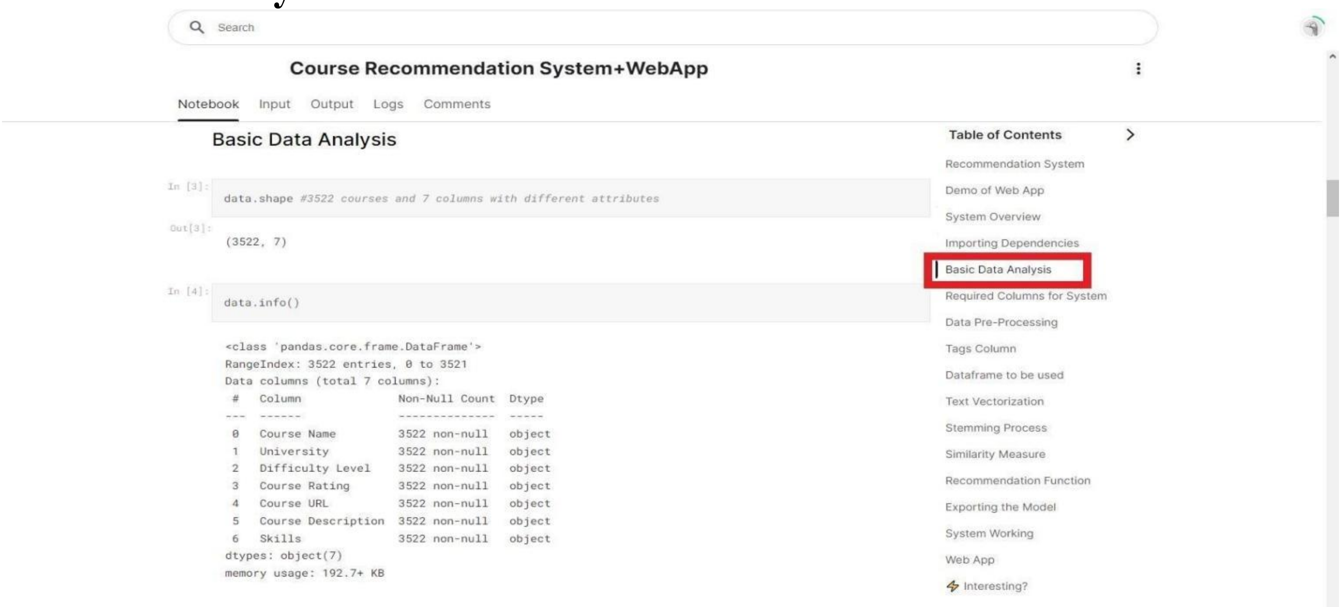


Fig 4.2.1: Basic Data Analysis

Here, The dataset comprises 3522 entries with 7 columns, primarily consisting of categorical data. The columns include 'Course Name,' 'University,' 'Difficulty Level,' 'Course Rating,' 'Course URL,' 'Course Description,' and 'Skills.' All columns have non-null values, indicating a complete dataset. The 'Difficulty Level' and 'Course Rating' columns are of the 'object' data type, suggesting the need for potential conversion to numerical types for further quantitative analysis. The dataset provides a comprehensive overview of various courses, their attributes, and associated information, forming a foundational basis for more in-depth data analysis and exploration.

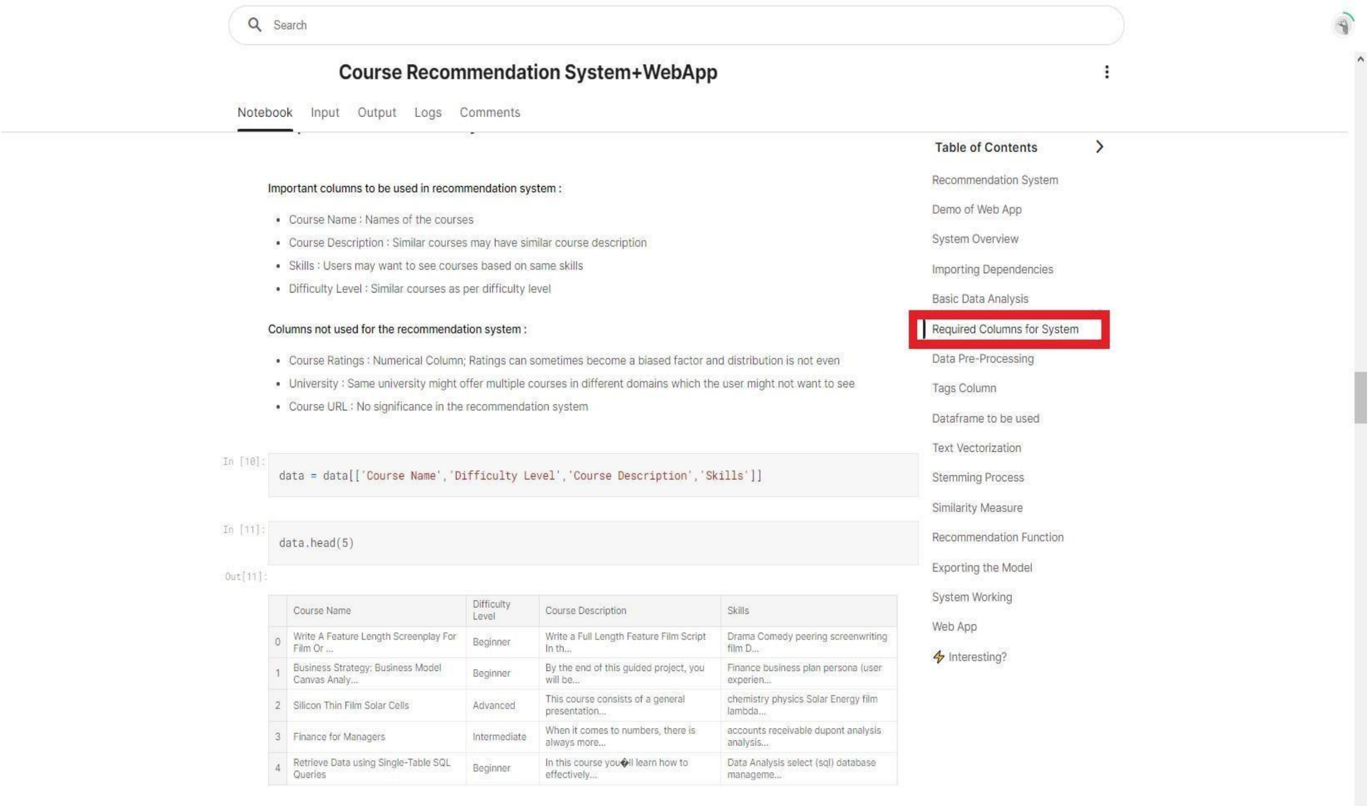
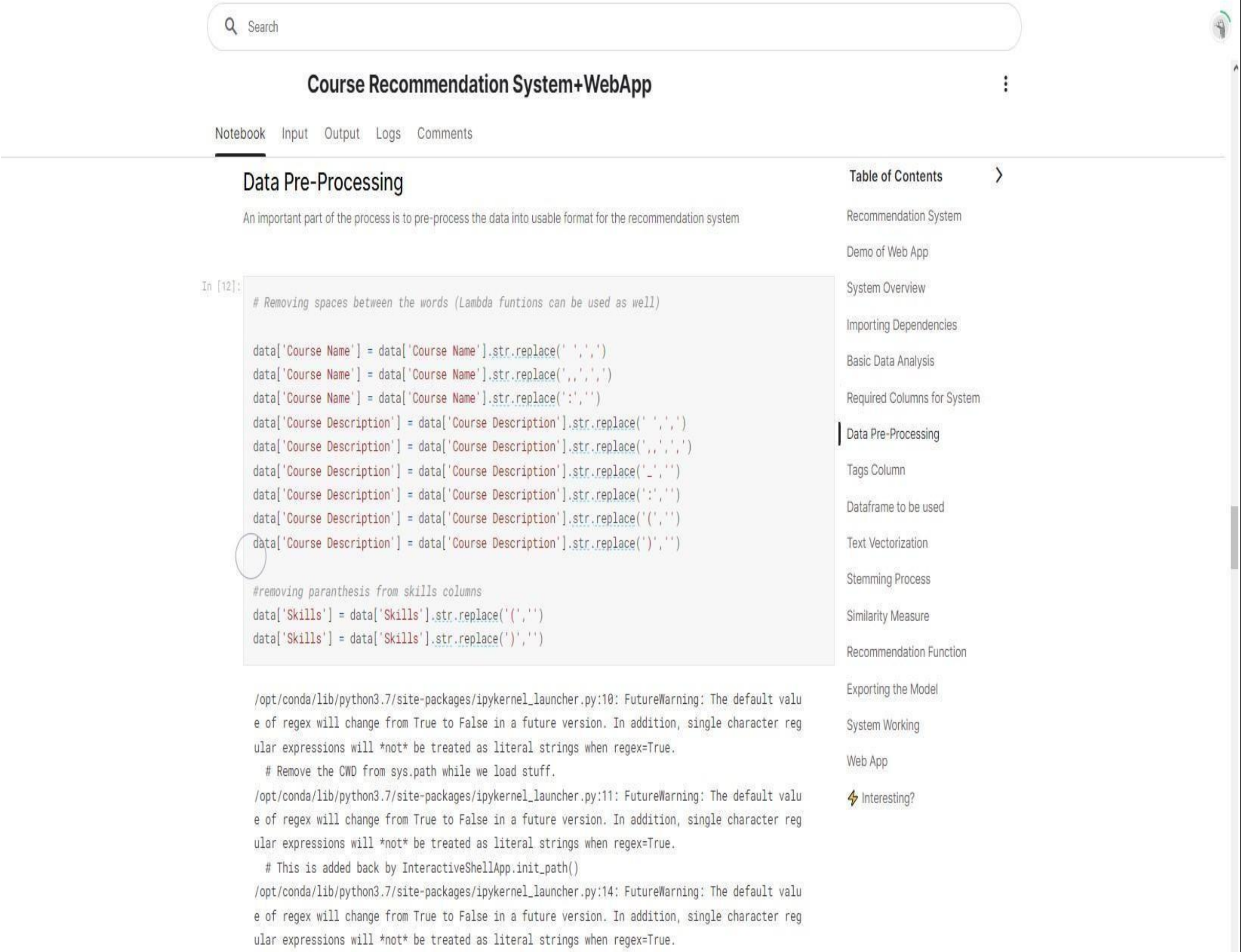


Fig 4.2.2: Required Column for system

Data Pre-Processing :

The dataset undergoes a preprocessing step to remove spaces between words in the 'Course Name' and 'Course Description' columns. Additionally, the removal of special characters, such as commas, colons, underscores, and parentheses, has been applied to enhance consistency and cleanliness in the textual data. The 'Skills' column has also been processed to eliminate parentheses. These transformations aim to standardize the format of text-based information, preparing the dataset for more effective analysis and feature extraction.



Search

Course Recommendation System+WebApp

Notebook Input Output Logs Comments

Data Pre-Processing

An important part of the process is to pre-process the data into usable format for the recommendation system

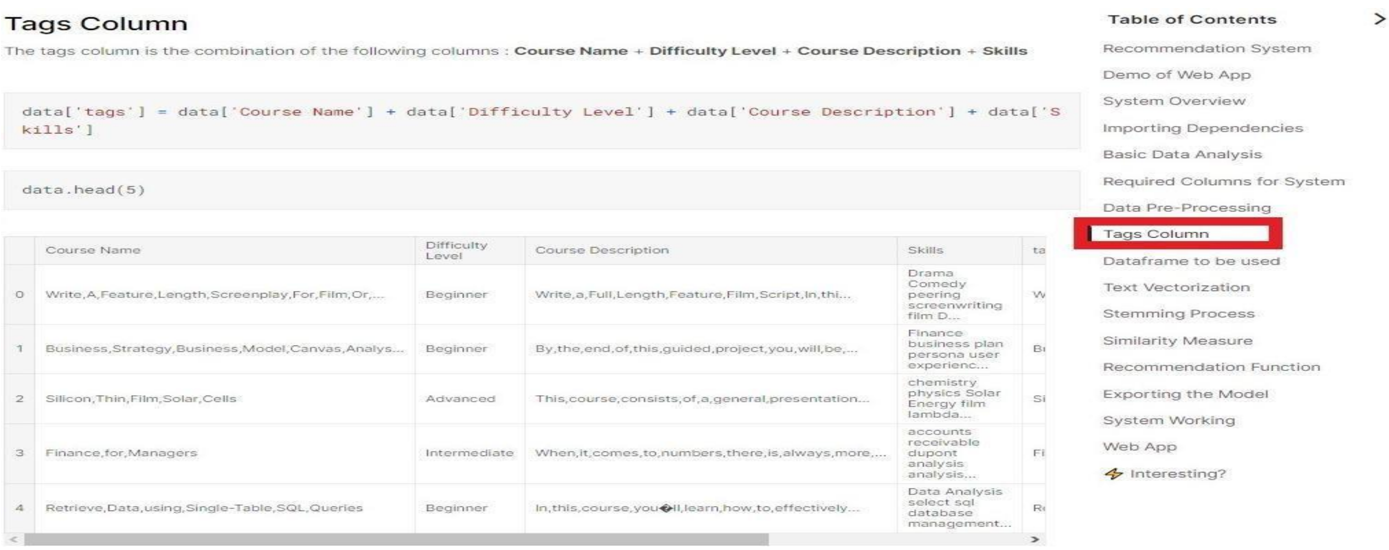
In [12]:

```
# Removing spaces between the words (Lambda funtions can be used as well)

data['Course Name'] = data['Course Name'].str.replace(' ',',')
data['Course Name'] = data['Course Name'].str.replace(',,',',')
data['Course Name'] = data['Course Name'].str.replace(':','')
data['Course Description'] = data['Course Description'].str.replace(' ',',')
data['Course Description'] = data['Course Description'].str.replace(',,',',')
data['Course Description'] = data['Course Description'].str.replace('_','')
data['Course Description'] = data['Course Description'].str.replace(':','')
data['Course Description'] = data['Course Description'].str.replace('(','')
data['Course Description'] = data['Course Description'].str.replace(')','')

#removing paranthesis from skills columns
data['Skills'] = data['Skills'].str.replace('(','')
data['Skills'] = data['Skills'].str.replace(')','')
```

```
/opt/conda/lib/python3.7/site-packages/ipykernel_launcher.py:10: FutureWarning: The default valu
e of regex will change from True to False in a future version. In addition, single character reg
ular expressions will *not* be treated as literal strings when regex=True.
  # Remove the CWD from sys.path while we load stuff.
/opt/conda/lib/python3.7/site-packages/ipykernel_launcher.py:11: FutureWarning: The default valu
e of regex will change from True to False in a future version. In addition, single character reg
ular expressions will *not* be treated as literal strings when regex=True.
  # This is added back by InteractiveShellApp.init_path()
/opt/conda/lib/python3.7/site-packages/ipykernel_launcher.py:14: FutureWarning: The default valu
e of regex will change from True to False in a future version. In addition, single character reg
ular expressions will *not* be treated as literal strings when regex=True.
```

Table of Contents

Fig 4.2.3: Data Preprocessing



Tags Column

The tags column is the combination of the following columns : **Course Name** + **Difficulty Level** + **Course Description** + **Skills**

```
data['tags'] = data['Course Name'] + data['Difficulty Level'] + data['Course Description'] + data['S
kills']
```

```
data.head(5)
```

	Course Name	Difficulty Level	Course Description	Skills	ta
0	Write,A,Feature,Length,Screenplay,For,Film,Or,...	Beginner	Write,a,Full,Length,Feature,Film,Script,In,thi...	Drama Comedy peering screenwriting film D...	W
1	Business,Strategy,Business,Model,Canvas,Analys...	Beginner	By,the,end,of,this,guided,project,you,will,be,...	Finance business plan persona user experienc...	B
2	Silicon,Thin,Film,Solar,Cells	Advanced	This,course,consists,of,a,general,presentation...	chemistry physics Solar Energy film lambda...	Si
3	Finance,for,Managers	Intermediate	When,it,comes,to,numbers,there,is,always,more,...	accounts receivable dupont analysis analysis...	Fi
4	Retrieve,Data,using,Single-Table,SQL,Queries	Beginner	In,this,course,you�ll,learn,how,to,effectively...	Data Analysis select sql database management...	R

Table of Contents

Fig 4.2.4 : Tags Column

Dataframe Used and Implemented :

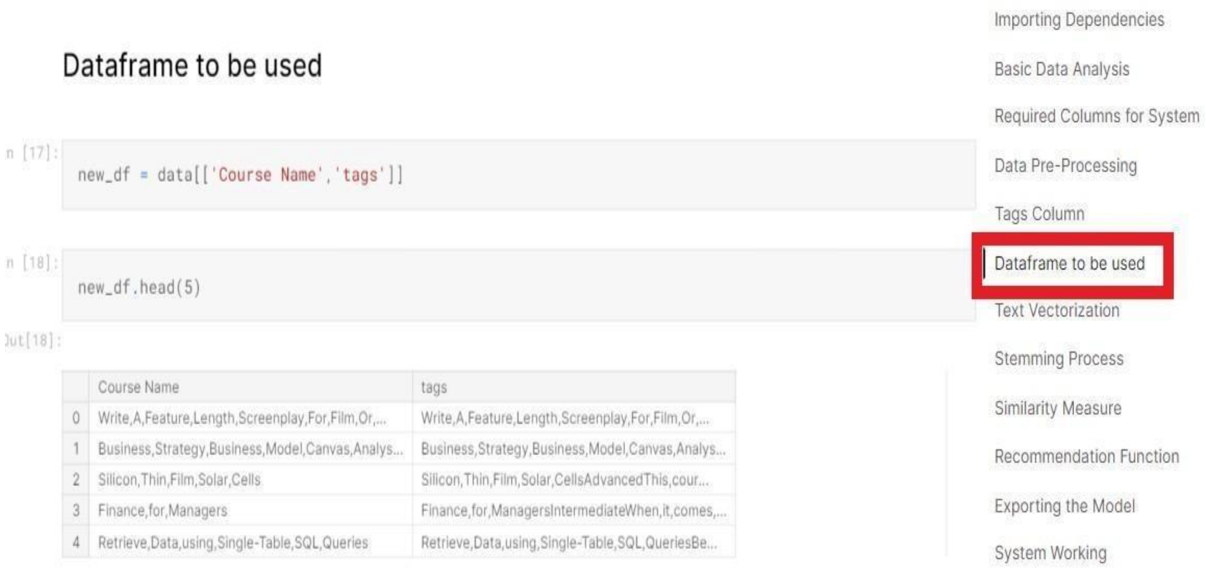


Fig 4.2.5 : Dataframe to be used

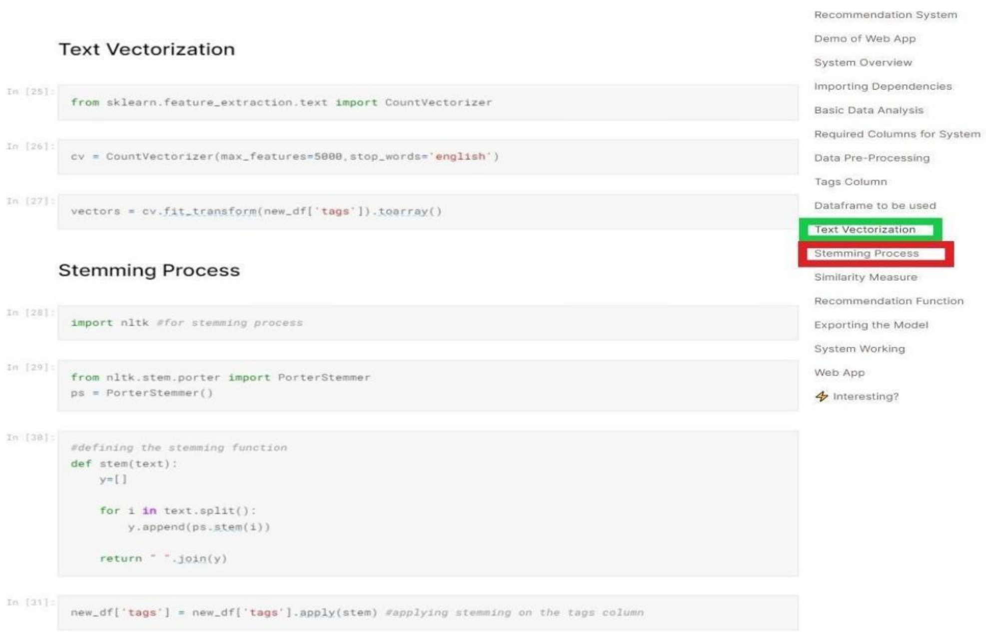


Fig 4.2.6 : Text Vectorization and Stemming Process

Text vectorization using Count Vectorizer from the scikit-learn library involves converting textual data, specifically the 'tags' column in the Data Frame 'new_df,' into numerical vectors. The process transforms each document (tag) into a vector representation, where each element corresponds to the frequency of a specific word in the document. The 'max_features=5000' parameter limits the vocabulary size to the top 5000 most frequent words. Additionally, 'stop_words='english'' removes common English stop words (e.g., 'the,' 'and') during the vectorization process. The resulting 'vectors' variable stores the numerical representations of the text data in a two-dimensional array, making it suitable for input into machine learning models that require numerical features. This approach facilitates the analysis of textual information and is commonly used in natural language processing tasks such as document classification and sentiment analysis.

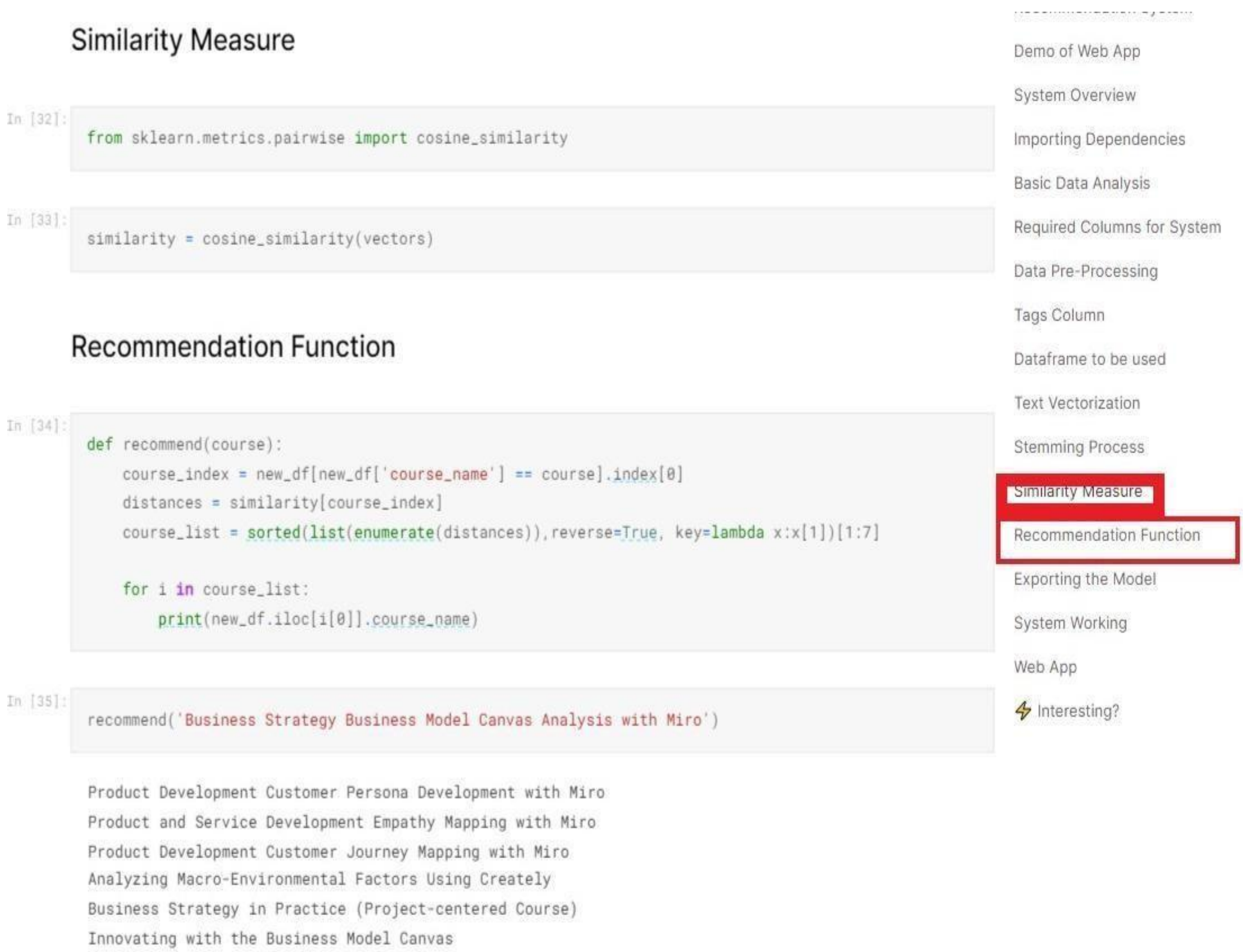


Fig 4.2.7: Similarity measure and Recommendation Function

The code utilizes cosine similarity to measure the similarity between courses based on their vectorized representations. The 'recommend' function takes a 'course' as input, identifies its index in the dataset, and calculates cosine similarities with other courses. It then sorts the courses by similarity in descending order, excluding the input course, and prints the names of the top 6 recommended courses. The recommendations aim to suggest courses with the highest similarity to the input course, providing users with personalized and relevant learning options.

Exporting the Model :

Exporting the Model

In [36]:
```
import pickle
```

In [37]:
```
# pickle.dump(similarity,open('similarity.pkl','wb'))
# pickle.dump(new_df.to_dict(),open('course_list.pkl','wb')) #contains the dataframe in dict
# pickle.dump(new_df,open('courses.pkl','wb'))
```

System Working

👉 View the Web app in action here

Web App

👉 Checkout the live application here Code to web app here

Fig 4.2.8 : Exporting the Model

The code employs the 'pickle' module to export the similarity matrix and dataset for later use. The 'similarity.pkl' file stores the cosine similarity matrix, while 'course_list.pkl' contains the dataset in dictionary format. 'courses.pkl' exports the entire DataFrame. Leveraging Streamlit, these files can be loaded and utilized in a web application. Streamlit enables seamless integration, allowing developers to create interactive and user-friendly interfaces that leverage the precomputed similarity data to provide dynamic and personalized course recommendations. By loading the exported files within a Streamlit app, users can effortlessly access and explore the course recommendations generated by the machine learning model.

CHAPTER 5 : CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

Concluding the project, it can be said that it is a robust and innovative Learning Management System (LMS) that transcends the conventional boundaries of online education. Through meticulous design and strategic integration of advanced technologies, our LMS stands as a testament to our commitment to redefining the learning experience for our users.

Personalisation Redefined:

The cornerstone of the project lies in the implementation of a hybrid recommendation system, marrying collaborative filtering with content-based filtering. This fusion results in a dynamic and intelligent learning experience, where each user is presented with precisely tailored course suggestions. This level of personalization has the potential to revolutionize how students navigate their educational journey, ensuring that the content they engage with aligns seamlessly with their individual goals and preferences.

User-Centric Approach:

At the heart of the project is a user-centric philosophy. From the initial user registration process to the exploration of a comprehensive course catalog, every aspect of the system has been crafted to prioritizethe needs and aspirations of our users. The seamless integration of payment processing with Stripe further enhances the user experience, offering a secure and efficient platform for financial transactions.

Intuitive Exploration and Dynamic Learning Paths:

The incorporation of a comprehensive course catalog not only empowers users to explore diverse subjects but also facilitates the creation of dynamic learning paths. Users can adapt their educational journey based on their evolving interests and skill levels, fostering a sense of autonomy and engagement.

Innovation in Financial Transactions:

The integration of Stripe for payment processing adds a layer of innovation and trust to our system. Users can confidently engage in financial transactions, secure in the knowledge that their sensitive information is handled with the utmost security and efficiency.

Future Prospects and Continuous Improvement:

As we conclude this phase of the project, our eyes are firmly set on the future. We recognize that education is an ever-evolving landscape, and our system is poised for continuous improvement. Future enhancements may include additional features, refined recommendation algorithms, and further optimizations to elevate the overall learning experience.

In essence, the project is more than just a Learning Management System; it's a catalyst for personalized and transformative learning experiences. By seamlessly integrating user-centric design, advanced recommendation systems, and secure payment processing, we aspire to contribute to a future where education is not only accessible but also tailored to the unique needs and aspirations of each learner.Further Scope

5.2 Future Scope for Next Semester:

In the pursuit of continual enhancement and innovation, the current project lays the foundation for future developments and refinements. The following areas highlight aspects that could not be fully realized in the current scope but present exciting opportunities for improvement and expansion in the next semester.

1. Skill Testing Exam:

Despite our best efforts, the full implementation of the skill assessment examination component facedchallenges that warrant dedicated attention in the next semester:

Comprehensive Skill Assessment Algorithms:

The current skill testing exam provides a foundational evaluation, but future iterations could benefit from the incorporation of more advanced assessment algorithms. This could involve the integration of adaptive testing methodologies, real-time performance analytics, and a broader range of question types to offer a more nuanced understanding of user skills.

Granular Proficiency Metrics:

The future scope involves delving deeper into granular proficiency metrics. Instead of a one-time assessment, an ongoing evaluation system could be implemented, providing continuous feedback to users. This could involve periodic skill check-ins, adaptive difficulty adjustments, and a more dynamic approach to tracking and enhancing user proficiency over time.

Integration with Learning Paths:

To maximize the impact of skill assessments, integrating the results seamlessly into the user's learning path is crucial. Future development could focus on refining the connection between assessment outcomes and personalized course recommendations, ensuring a more cohesive and adaptive learning journey.

2. Video Content Delivery:

While the integration of Mux has significantly enhanced the video content delivery aspect, future semesters present an opportunity for further refinement and expansion:

Advanced Streaming Features:

Future developments could explore additional streaming features, such as live Q&A sessions, interactive quizzes within videos, and social learning components. These features would contribute to a more dynamic and engaging video learning experience.

Optimizing Bandwidth Usage:

To ensure an uninterrupted learning experience across diverse network conditions, future efforts could focus on optimizing bandwidth usage. Implementing adaptive streaming techniques that dynamically adjust video quality based on the user's internet connection can further enhance the reliability of video content delivery.

3. Leveraging More Data:

Recognizing the pivotal role of data in refining the recommendation system and overall user experience,future semesters could explore the following:

Expanded User Profiles:

Collecting more comprehensive user data, including learning preferences, study habits, and feedback, can significantly enhance the accuracy of personalized recommendations. Future efforts could focus on implementing user surveys, behavioral analytics, and sentiment analysis to capture a more holistic view of user preferences.

Incorporating External Data Sources:

Integrating data from external sources, such as industry trends, job market demands, and emerging technologies, can augment the relevance of course recommendations. Collaborations with industry partners or the inclusion of open data sources could enrich the recommendation algorithms with real-worldinsights.

Enhanced User Interactivity:

Explore ways to enhance user interactivity with video content, such as bookmarking, note-taking features, and collaborative learning options. These additions would contribute to a richer and more immersive learning environment.

Continuous Machine Learning Model Refinement:

Implementing a strategy for continuous machine learning model refinement is essential. Future semesters could explore techniques such as reinforcement learning or ensemble methods to adapt the recommendation system dynamically to evolving user needs and changing educational landscapes.

In embracing these future scopes, the project not only acknowledges the areas that require further attention but also lays the groundwork for a more comprehensive and impactful educational platform in thesemesters to come.

REFERENCES

[1] https://www.youtube.com/watch?v=_hf_y-_sj5Y&t=14s

[2] https://www.youtube.com/watch?v=Big_aFLmekI

[3] Michael J. Pazzani, Daniel Billsus. Content-Based Recommendation Systems. Internet:

[4] H. Jafarkarimi, A. T. H. Sim and R. Saadatdoost, "A naive recommendation model for large databases", International Journal of Information and Education Technology, vol. 2, no. 3, pp. 216, 2012.

[5] Akhshabi S, Begen AC, Dovrolis C (2011) An experimental evaluation of rate- adaptation algorithms inadaptive streaming over HTTP.

[6] H. Zhang, T. Huang, Z. Lv, S. Liu and Z. Zhou, "MCRS: A course recommendation system for MOOCs", Multimedia Tools and Applications, vol. 77, pp. 7051-7069, 2018.

[7] Ashraf, E., Manickam, S., & Karuppayah, S. (2021). A comprehensive review of curse recommender systems in e-learning. Journal of Educators Online, 18, 23–35.

[8] Drachsler, H., Verbert, K., Santos, O. C., & Manouselis, N. (2015). Panorama of Recommender Systems to Support Learning. In F. Ricci, L. Rokach, & B. Shapira (Eds.), Recommender Systems Handbook (pp. 421–451).

[9] Klašnja-Milićević, A., Ivanović, M., & Nanopoulos, A. (2015). Recommender systems in e-learning environments: A survey of the state-of-the-art and possible extensions. Artificial Intelligence Review, 44(4), 571–604.

[10] Ferreira, V., Vasconcelos, G., & França, R. (2017). Mapeamento Sistemático sobre Sistemas de Recomendações Educacionais. Proceedings of the XXVIII Brazilian Symposium on Computers in Education, 253-262.

[11] Khanal, S. S., Prasad, P. W. C., Alsadoon, A., & Maag, A. (2019). A systematic review: Machine learning based recommendation systems for e-learning. Education and Information Technologies, 25(4), 2635–2664.

[12] Nascimento, P. D., Barreto, R., Primo, T., Gusmão, T., & Oliveira, E. (2017). Recomendação de Objetos de Aprendizagem baseada em Modelos de Estilos de Aprendizagem: Uma Revisão Sistemática da Literatura. Proceedings of XXVIII Brazilian Symposium on Computers in Education- SBIE, 2017, 213–222.

[13] Rivera, A. C., Tapia-Leon, M., & Lujan-Mora, S. (2018). Recommendation Systems in Education: A Systematic Mapping Study. Proceedings of the International Conference on Information Technology & Systems (ICITS 2018), 937–947.

[14] Tarus, J. K., Niu, Z., & Mustafa, G. (2018). Knowledge-based recommendation: A review of ontology-based recommender systems for e-learning. Artificial Intelligence Review, 50(1), 21–48.