A

Project Report

on

Adaptive Online Platform for Enhanced Teaching and Learning

submitted for partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

in

Computer Science

By

AMOD KATIYAR (2000290120022)

HIMANSHU KUMAR (2000290120076)

ANIKET BHARDWAJ (2000290120024)

Under the Supervision of

Dr. GAURAV DUBEY

Professor

Department of Computer Science

KIET Group of Institutions, Ghaziabad

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University, Lucknow

May 2024

DECLARATION

I/We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Amod Katiyar (2000290120022)

Himanshu Kumar (2000290120076)

Aniket Bhardwaj (2000290120024)

CERTIFICATE

This is to certify that Project Report entitled "**Adaptive Online Platform for Enhanced Teaching and Learning**" which is submitted by **Amod Katiyar, Aniket Bhardwaj, Himanshu Kumar** in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science of Dr. A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

.

Date:

Supervisor

Dr. GAURAV DUBEY

Professor

(Computer Science)

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe a special debt of gratitude to Professor **Dr. Gaurav Dubey**, Department of Computer Science, KIET, Ghaziabad, for his/her constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.

We also take the opportunity to acknowledge the contribution of **Dr. Ajay Kumar Srivastava**, Head of the Department of Computer Science, KIET, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

Finally, we acknowledge our friends for their contribution in the completion of the project.

Amod Katiyar (2000290120022)

Himanshu Kumar (2000290120076)

Aniket Bhardwaj (2000290120024)

TABLE OF CONTENTS

Page No.

ABSTRACT

This project introduces a user-friendly platform aimed at assisting students in navigating the diverse landscape of programming languages. By focusing on industry trends, the website provides clear and concise language recommendations, offering valuable insights into the rapidly evolving tech environment. The primary objective is to streamline students' learning journeys by creating a centralized hub that not only identifies current language demands but also serves as a strategic guide, helping them connect their passion with proficiency.

In addition to its recommendation feature, the platform goes beyond the basics, encouraging students to explore learning paths aligned with emerging trends. By leveraging industry data, the platform empowers students to make informed decisions, fostering a deeper connection between their interests and the ever-changing world of technology. The platform aims to be a comprehensive resource, offering guidance on honing fundamental concepts and adapting to new industry demands, ensuring that students stay ahead in the competitive landscape of programming.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

1. UI: User Interface
2. DFD: Data Flow Diagram
3. ER: Entity-Relationship
4. RDBMS: Relational Database Management System
5. SRS: Software Requirement Specification
6. SDLC: Software Development Life Cycle
7. HTML: Hypertext Markup Language
8. CSS: Cascading Style Sheets
9. JS: JavaScript
10. AI: Artificial Intelligence
11. ML: Machine Learning
12. IEEE: Institute of Electrical and Electronics Engineers
13. API: Application Programming Interface

CHAPTER 1
INTRODUCTION

1.1 Introduction to Project

In today's fast-paced tech world, students face the challenge of choosing relevant programming languages to learn. This project introduces a user-friendly platform designed to simplify this decision-making process. By focusing on real-time industry trends, the platform provides clear language recommendations, aiming to bridge the gap between students' passion and the demands of the ever-evolving technology sector. This report explores the development and functionality of the platform, highlighting its significance in guiding students towards informed choices and proficiency in programming.

1.2 Project Category: The project falls under the category of an internet-based application development initiative. It revolves around the creation of a user-friendly platform accessible through the web to assist students in navigating programming language choices.

1.3 Objectives: The primary objectives of the project are:

1. Simplifying the process of selecting programming languages for students.
2. Providing real-time insights into industry trends.
3. Bridging the gap between students' passion and the demands of the technology sector.
4. Offering a centralized hub for language recommendations and strategic guidance.

1.4 Problem Formulation: The project addresses the challenges students face in selecting programming languages amid a myriad of choices. The goal is to streamline decision-making, considering both individual interests and the dynamic demands of the industry.

1.5 Proposed System: The proposed system is an internet-based platform that offers concise language recommendations and insights into current industry trends. It serves as a guide for students to make informed decisions about which programming languages to learn based on their interests and the evolving needs of the tech sector.

1.6 Structure of The Report:

Chapter 1

The report outlines the development of a user-friendly platform designed to assist students in selecting relevant programming languages amidst the fast-paced technology landscape. The introduction emphasizes the challenges students face in making informed decisions and the necessity for a tool that bridges the gap between their passion and the evolving demands of the tech sector. The project falls under the category of internet-based application development, focusing on real-time industry trends and language recommendations. Objectives include simplifying the decision-making process, providing industry insights, and offering a centralized hub for strategic guidance. The report's structure entails a detailed exploration of the platform's development, including methodologies, results, and conclusions, with a focus on guiding students toward proficient programming language choices.

Chapter 2

This chapter provides a comprehensive literature review on programming language trends and educational tools designed to aid in language selection. Scholarly papers and articles are summarized, highlighting key insights into the popularity and utility of various programming languages, the evolution of industry demands, and existing educational platforms. Research gaps are identified, including the lack of real-time trend analysis and user-friendly interfaces. The chapter formulates the problem statement, addressing the urgent need for a platform that consolidates industry trends and provides clear language recommendations to guide students in their learning paths.

Chapter 3

This chapter outlines the proposed system for assisting students in choosing programming languages. It details the system architecture, emphasizing features such as industry trend analysis, language recommendation algorithms, user-friendly interface design, and real-time data integration. The unique approach ensures that the platform is accessible, intuitive, and tailored to meet the diverse needs of students, distinguishing it from existing solutions.

Chapter 4

This chapter delves into the requirement analysis and system specifications for implementing the language recommendation platform. A feasibility study is conducted, covering technical, economic, and operational aspects. The software requirement specification outlines the system's introduction, overall description, assumptions, and dependencies. It details the proposed methodology, operating environment, constraints, external interface requirements, and non-functional requirements. Additionally, it discusses software quality attributes, business rules, and technical requirements, concluding with the choice of the Agile SDLC model for development. Diagrams, including DFDs and an ER diagram, further elucidate the system's design and functionality.

Chapter 5

This chapter introduces the implementation phase, outlining the languages, tools, and technologies employed in developing the platform. It highlights the utilization of HTML/CSS, JavaScript, Python, React, Mongo DB, and data analysis libraries. This comprehensive approach aims to ensure robustness and functionality in the platform, enhancing its utility for students.

Chapter 6

This chapter focuses on testing and maintenance, detailing various testing techniques and test cases employed. Test cases cover user registration, login, language recommendation, and trend analysis scenarios, ensuring the system's functionality, reliability, and user-friendliness. This meticulous testing process aims to deliver a high-quality and dependable platform.

Chapter 7

This chapter showcases the results and discussions of the implemented platform, presenting user interface representations, module descriptions, snapshots, back-end representations, and database tables. It provides insights into the various functionalities and components of the system, illustrating how students interact with the platform, receive language recommendations, and access real-time industry trends. The snapshots offer a visual understanding of the system's interface and backend processes, enhancing comprehension.

Chapter 8

This chapter concludes the report by summarizing the research's significance in aiding students to choose relevant programming languages through a user-friendly platform. It emphasizes the platform's role in bridging the gap between students' interests and industry demands. The chapter outlines future scope, suggesting avenues for improvement like enhanced data analytics, machine learning integration, and expanding the platform's features to include personalized learning paths and advanced career guidance, to further assist students in their educational and professional journeys.

1.7 Unique Features of the System:

The system's unique features include:

1. Industry trend analysis.
2. Clear and concise language recommendations.
3. A user-friendly interface for easy navigation.
4. Encouraging exploration of dynamic learning paths.
5. Bridging the gap between passion and proficiency in programming.

CHAPTER 2

LITERATURE REVIEW

Table: 2.1 Literature Review

Sr. no	PAPER TITLE	AUTHORS	PUBLISH YEAR	SUMMARY
1	Review of WordPress website vs coding website	Swapnil S More	2018	This research paper mainly aims at shedding light on the two ways you can create a website, that is by using Content Management System (e.g. -WordPress Website) and Coding Website (HTML).
2	A review and analysis of technologies for developing web applications	SOLOMON ANTONY	2012	Object oriented approach build web applications very efficient when one can accomplish more in less time. Because it uses modern processes, by this both developers and clients can benefit from this. To develop these types of applications there are so many scripting languages and new technologies are there we don't have to stick to one. It gives good knowledge to the developers as well as clients in choosing a web application platform
3	Web Application Development -a study on UML Web Application Extension	Andreas Oskarsson, Martin Kling, Tobias Norberg	2002	The use of UML for Web application development will result in good design, regarding maintainability. Extensibility was supported through low coupling, high cohesion, and the possibility to create generalization/specialization hierarchies. Reusability was

				supported by the ability to apply white-box reuse. This thesis was unable to resolve whether UML WAE supports black-box reuse. The produced documentation was understandable, easy to read and had highly traceable diagrams. With our three criteria extensibility, reusability and documentation examined and found to be in support, the conclusion is that the use of UML for Web application development resulted in good design regarding maintainability.
4	Website Development Technologies: A Review	Pratiksha D Dutonde , Shivani S Mamidwar , Monali Sunil Korvate	2022	The Worldwide internet represents the highest technology to the perfect of a very distributed network atmosphere for polymorphic communication. As such, it should be although of as a paradigm shift aloof from earlier network protocols. Web Applications design issues the look and implementation of pc code that runs on internet servers, rather than running only on desktop computers, laptops, or mobile devices
5	Research on HTML5 in Web Development	Ch Rajesh,K S V Krishna Srikanth	2014	HTML5 introduces new elements and featuresthat allow developers to improve interoperability, handling elements in a precise way saving time and costs. HTML5 is an awesome technology and hasthe possibility to make the web even more predominant and extensive as it is today from desktop computers to mobile devices and in the future maybe even domestics appliances. The potential of

				HTML5 will soften the line between desktop and online applications. The problem HTML5 may suffer in the coming days is that an opportunity will be available for malware writers which may make today's common hacks
6	Big Data-Based Improved Data Acquisition and Storage System for Designing Industrial Data Platform	DAOQU GENG , CHENGYUN ZHANG , CHENGJING XIA , XUE XIA , QILIN LIU , AND XINSHUAI FU	2019	This paper investigates optimizing industrial data platform design through efficient compression and serialization methods, achieving significant reductions in compression and serialization times. LZ4 outperforms LZO for file storage, and ProTour serialization demonstrates superior efficiency, enabling diverse data analysis functionalities such as machine learning and optimization.
7	Exploratory Analysis for Big Social Data Using Deep Network	CHAO WU, GUOLONG WANG , JIANGCHENG ZHU , PIYAWAT LERTVITTAYAKUMJORN , SIMON HU , (Member, IEEE), CHILIE TAN , HONG MI, YADAN XU AND JUN XIAO	2019	This paper introduces a novel exploratory analysis paradigm for big social data, leveraging deep networks to uncover complex relationships without manual feature selection. Through three experiments, including prediction and network capacity modification analysis, the proposed methodology demonstrates applicability across various scenarios, enabling data-driven social science research aided by advancements in deep learning algorithms.
8	Mining Conditional Functional Dependency Rules on Big Data	Mingda Li, Hongzhi Wang, and Jianzhong L	2020	This paper presents a novel approach for mining Conditional Functional Dependency (CFD) rules on big data, overcoming challenges of dataset volume

				and quality. By employing sampling algorithms and fault-tolerant rule discovery techniques, the method efficiently discovers effective CFD rules even on large and low-quality datasets. Experimental results confirm its effectiveness in discovering rules on billion-tuple data within a reasonable time frame, outperforming existing algorithms and offering potential extensions to parallel platforms.
9	Big Data Analytics and Mining for Effective Visualization and Trends Forecasting of Crime Data	MINGCHEN FENG, JIANGBIN ZHENG, JINCHANG REN , (Senior Member, IEEE), AMIR HUSSAIN , (Senior Member, IEEE), XIUXIU LI , YUE XI , AND QIAOYUAN LIU	2019	This paper explores the application of Big Data Analytics (BDA) in visualizing and forecasting crime trends using datasets from San Francisco, Chicago, and Philadelphia. Through exploratory data analysis, statistical analysis, and visualization, significant patterns are discovered, leveraging 13 featured attributes of crime incidents. Future plans include enhancing the analytics platform to incorporate multivariate visualization, graph mining techniques, and fine-grained spatial analysis for uncovering more intricate patterns and trends in crime data.
10	Social Set Analysis: A Set Theoretical Approach to Big Data Analytics	RAVI VATRAPU1,2, RAGHAVA RAO MUKKAMALA1, ABID HUSSAIN1 , AND BENJAMIN FLESCH1	2016	This paper explores the application of Big Data Analytics (BDA) in visualizing and forecasting crime trends using datasets from San Francisco, Chicago, and Philadelphia. Through exploratory data analysis, statistical analysis, and visualization, significant patterns are discovered,

				leveraging 13 featured attributes of crime incidents. Future plans include enhancing the analytics platform to incorporate multivariate visualization, graph mining techniques, and fine-grained spatial analysis for uncovering more intricate patterns and trends in crime data.
11	An Overview of Data Analysis and Interpretations in Research.	Dibekulu, Dawit.	2020	This paper explores the vital role of data analysis in research, highlighting its necessity for generating new knowledge and solving problems.
12	HTML5 -Unleashing the Power of the Web	Lubbers, Peter & Albers, Brian & Salim, Frank & Kumar, Mahesh.	2023	This paper explores a new approach for HTML and CSS, making them accessible to everyone with engaging visuals and a flexible structure. It offers an enjoyable, user-friendly experience that appeals to hobbyists, students, and professionals alike.
13	Use of React.js in Modern Web Development and its Comparative Analysis	Analysis Kunal Goyal, Dr. Vishal Shrivastava , Dr. Akhil Pandey	2024	This paper explores the significance of React.js in web development, highlighting its versatility and flexibility. By comparing React.js to other front-end methodologies, it demonstrates why industry leaders like Facebook, Google, and Airbnb favor this approach for crafting dynamic web applications.
14	React-JS: A Cutting-Edge Framework for Web Designing	Navratan Mal, Dr. Vishal Shrivastava, Dr. Akhil Pandey.	2024	This paper explores React-JS, a top JavaScript toolkit for creating user interfaces, known for its declarative approach, flexibility, and efficiency. It examines React-JS's features, including the virtual DOM, JSX syntax, and one-way data flow, while comparing it

				to other frameworks and discussing its use in both web and mobile application development.
15	A Comparative Analysis of Database Management Systems for Time Series Data	TOVE VERNER-CARLSSON VALERIO LOMANTO	2023	This paper explores the performance of PostgreSQL (with TimescaleDB) and MongoDB for time series data by comparing their query execution times. Using weather data from the Swedish Meteorological and Hydrological Institute, we conduct benchmark tests and analyze query-by-query results.
16	An Analysis of Time Series Analysis and Forecasting Techniques.	Malik, Pankaj & Dangi, Aditya & Singh, Aditya & Asst, Thakur & Pratap, Aditya & Parihar, Singh & Sharma, Utkarsh & Mishra, Lakshya.	2023	This paper explores time series forecasting, a technique that predicts future events by analyzing past trends, assuming that future trends will mirror historical patterns
17	IMPLEMENTATION AND COMPARISON OF MERN STACK TECHNOLOGY WITH HTML/CSS, SQL, PHP & MEAN IN WEB DEVELOPMENT	Er. Vikas Goyal, Ashish Kumar Mishra, Daljeet Singh,	2023	This paper provides a concise overview of web development with the MERN (MongoDB, Express.js, React, Node.js) stack, comparing it with other popular technologies. The MERN stack, a full-stack JavaScript framework, is widely used for building modern web applications
18	Predicting Stock Prices Using Hybrid LSTM andARIMA Model	Chi Ma, Jie Wu* , Hui Hu, YueNai Chen, JingYan Li,	2023	This paper provides an exploration into the intersection of China's evolving financial market and the growing demand for precise financial information services.

This chapter provides a comprehensive review of the existing literature relevant to the proposed platform designed to assist students in choosing and learning programming languages. The review focuses on two main areas: web development and machine learning, which are essential components of the proposed platform.

Web Development

Numerous studies have delved into various web development technologies, offering insights into the strengths and weaknesses of different frameworks and languages. For instance, Antony (2012) highlighted the evolution of HTML5, which revolutionized web development by enabling rich and interactive web pages. Similarly, Oskarsson et al. (2002) discussed the importance of UML (Unified Modeling Language) in web application development, emphasizing its role in ensuring the functionality, scalability, and user experience of web platforms. Additionally, Dutonde et al. (2022) explored the usage of WordPress as a Content Management System (CMS) for building websites, noting its ease of use and plugin support. These studies provide valuable insights into the technological landscape that informs the development of the proposed platform.

Machine Learning

In the realm of machine learning, several studies have focused on recommendation systems, particularly their application in suggesting relevant content or products to users. Cheng et al. (2016) and Gandhi (2023) demonstrated the effectiveness of recommendation systems in personalizing user experiences based on past behavior or preferences. Singhal et al. (2021) highlighted the importance of incorporating domain-specific knowledge into recommendation systems to improve their accuracy and relevance to users' needs. Moreover, Li et al. (2021) and Abbas (2018) showcased the potential of machine learning in forecasting future trends, which can be applied to predict in-demand programming languages in the job market.

Proposed Improvements

Based on the literature review, the proposed platform aims to build upon existing web development technologies and recommendation systems. It seeks to integrate machine learning techniques to personalize the user experience and provide future-oriented insights into language popularity trends. By leveraging the strengths of collaborative filtering and content-based approaches, the platform aims to offer data-driven recommendations tailored to individual student needs and career goals. Additionally, it plans to incorporate time series analysis and forecasting models to predict future trends in language popularity, empowering students with the foresight to navigate the dynamic technological landscape.

Conclusion

The literature review underscores the importance of leveraging technology, particularly web development and machine learning, to address the challenges faced by students in choosing and learning programming languages. By building upon existing research and methodologies, the proposed platform aims to revolutionize programming language acquisition by providing personalized recommendations and future-oriented insights.

2.2 Research Gaps

While existing literature provides valuable insights into web development technologies, recommendation systems, and machine learning techniques, there remains a noticeable gap in the integration of these areas to address the specific challenge of programming language acquisition for students.

1. Integration of Web Development and Machine Learning: While some studies discuss web development technologies and others explore machine learning techniques, there is a lack of research that integrates these areas to create a comprehensive platform for programming language learning. Existing platforms may focus on either web development or machine learning but do not combine both aspects effectively to address the unique needs of programming language learners.

2. Personalized Learning Roadmaps: While recommendation systems are widely used to personalize user experiences, there is limited research on their application to create personalized learning roadmaps for programming language learners. Existing platforms may offer recommendations based on user preferences, but they often lack the depth and specificity required to guide students through a structured learning journey tailored to their individual goals and interests.

3. Forecasting Future Trends in Language Popularity: While some studies explore time series analysis and forecasting models, there is limited research on their application to predict future trends in programming language popularity specifically. Existing forecasting models may focus on broader trends in technology or consumer behavior but do not provide insights tailored to the programming language landscape, leaving students without accurate guidance on which languages to prioritize for learning.

4. Evaluation of Proposed Solutions: While the proposed platform aims to address the identified research gap, there is a need for empirical evaluation to assess its effectiveness in guiding programming language learners. Existing studies may propose theoretical frameworks or conceptual models, but empirical validation through user testing and feedback is essential to validate the platform's efficacy in real-world learning scenarios.

In summary, the research gap lies in the integration of web development and machine learning to create personalized learning roadmaps for programming language learners, incorporating forecasting models to predict future trends in language popularity, and empirical evaluation of proposed solutions to validate their effectiveness in practice. Closing this gap requires interdisciplinary research that leverages insights from web development, machine learning, and education to develop innovative platforms that meet the evolving needs of programming language learners.

2.3 Problem Formulation:

The problem formulation revolves around the development of a comprehensive platform that addresses the challenges faced by programming language learners, particularly in the context of web development technologies. The main objectives include:

1. Integrating Web Development and Machine Learning: The platform should seamlessly integrate web development technologies with machine learning techniques to provide a holistic learning experience for students. This involves leveraging web development frameworks and tools alongside machine learning algorithms to analyze user interactions and preferences, thereby personalizing the learning journey.

2. Creating Personalized Learning Roadmaps: The platform should generate personalized learning roadmaps for individual learners based on their goals, interests, and proficiency levels. This involves utilizing recommendation systems to suggest relevant learning resources, projects, and activities tailored to each student's unique needs and learning style.

3. Forecasting Future Trends in Language Popularity: The platform should incorporate time series analysis and forecasting models to predict future trends in programming language popularity. This involves analyzing historical data on language usage, industry trends, and developer interactions to identify emerging languages and technologies that are likely to gain prominence in the future.

4. Empirical Evaluation of Proposed Solutions: The platform should undergo empirical evaluation to assess its effectiveness in guiding programming language learners. This involves conducting user testing and gathering feedback from students to evaluate the platform's usability, accuracy of recommendations, and overall impact on learning outcomes.

By addressing these objectives, the proposed platform aims to provide students with a dynamic and data-driven approach to programming language acquisition, empowering them to make informed decisions and navigate the ever-evolving landscape of web development technologies with confidence.

CHAPTER 3

PROPOSED SYSTEM

3.1 Proposed System

We will start by identifying the most popular courses and programming languages, such as Java, C, and Python. Using Natural Language Processing (NLP) and APIs, we will extract high-quality links and resources from the web to provide students with the best learning materials available. This system is designed to assist students in choosing their learning paths and supplying them with the resources they need to succeed.

Our system will deploy a comprehensive recommender model that combines various recommendation techniques:

1. **Content-Based Methods**: These will analyze the content of available courses and match them with the student's preferences and past interactions.

2. **Collaborative Filtering Methods**:

- **Memory-Based**: This will rely on similarities between users and items, using historical interaction data to make recommendations.
- **Model-Based**: Utilizing advanced machine learning algorithms, we will analyze user interactions and predict future preferences.

3. **Hybrid Methods**: By integrating content-based and collaborative filtering approaches, we aim to enhance the accuracy and relevance of our recommendations.

The system will also display real-time graphs and predictions, illustrating the popularity and usage trends of various programming languages and courses. This visual representation will help students make informed decisions based on the latest data.

Additionally, we will use NLP to analyze and extract keywords such as "Python," "data science," "best," and "top" to identify trending and high-quality courses. APIs will facilitate the extraction of this data, ensuring that our recommendations are always up to date with the latest trends and resources.

Overall, our proposed system aims to provide a personalized and data-driven learning experience, guiding students through their educational journey with optimal recommendations and real-time insights.

3.1.1 Flow Chart

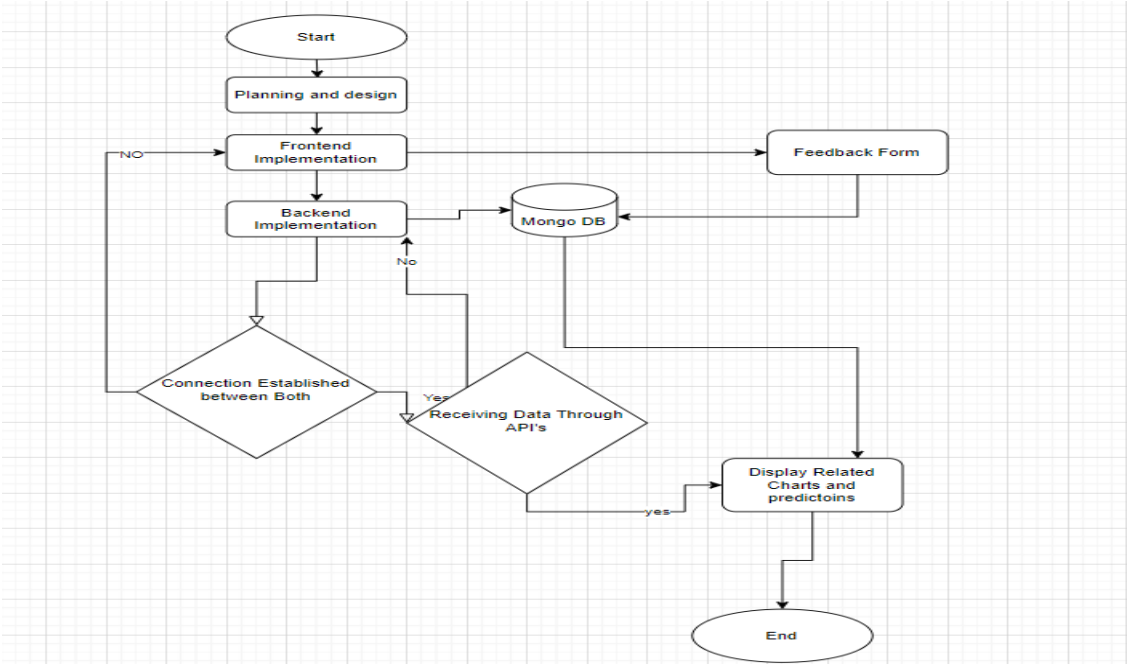


Fig:3.1 Flow Chart

The flowchart represents the process of developing and deploying a web application that displays data-related charts and predictions:

1. Start: The process begins.
2. Planning and Design: Initial phase involving planning and design of the application.
3. Frontend Implementation: Development of the user interface and user experience.
    - If there's an issue, it loops back to planning and design.
4. Backend Implementation: Development of the server-side logic and database integration.
    - Connects to MongoDB for data storage.
5. Connection Established between Both: Ensuring the frontend and backend communicate properly.
    - If not connected, it loops back to backend implementation.
6. Receiving Data Through APIs: Fetching data from external APIs.
    - If not received, it loops back to backend implementation.
7. Display Related Charts and Predictions: Rendering data visualizations and predictions on the frontend.
8. Feedback Form: Collecting user feedback and storing it in MongoDB.
9. End: The process completes.

3.1.2 ML Algorithm

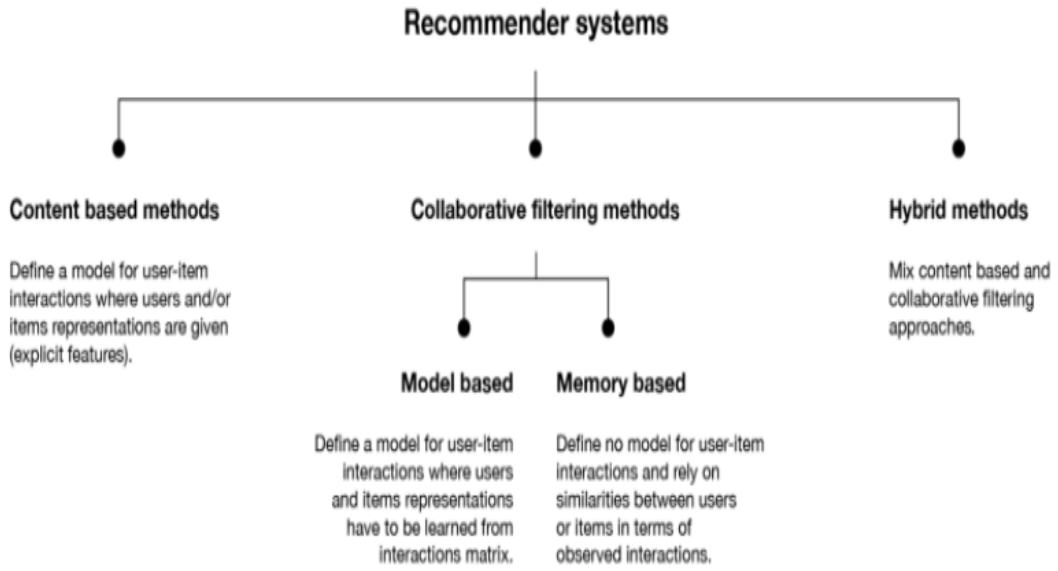


Fig:3.2 ML Algorithm

Explanation:

- Content-Based Methods: These recommend items by comparing the content of items and user preferences. It uses explicit features like keywords or attributes.

- Collaborative Filtering Methods:
    - Model-Based: These methods use machine learning models to predict user preferences based on past interactions.
    - Memory-Based: These rely on historical data and use similarity measures between users or items to make recommendations.

- Hybrid Methods: Combine both content-based and collaborative filtering methods to improve recommendation accuracy.

3.1.3 Detailed Types

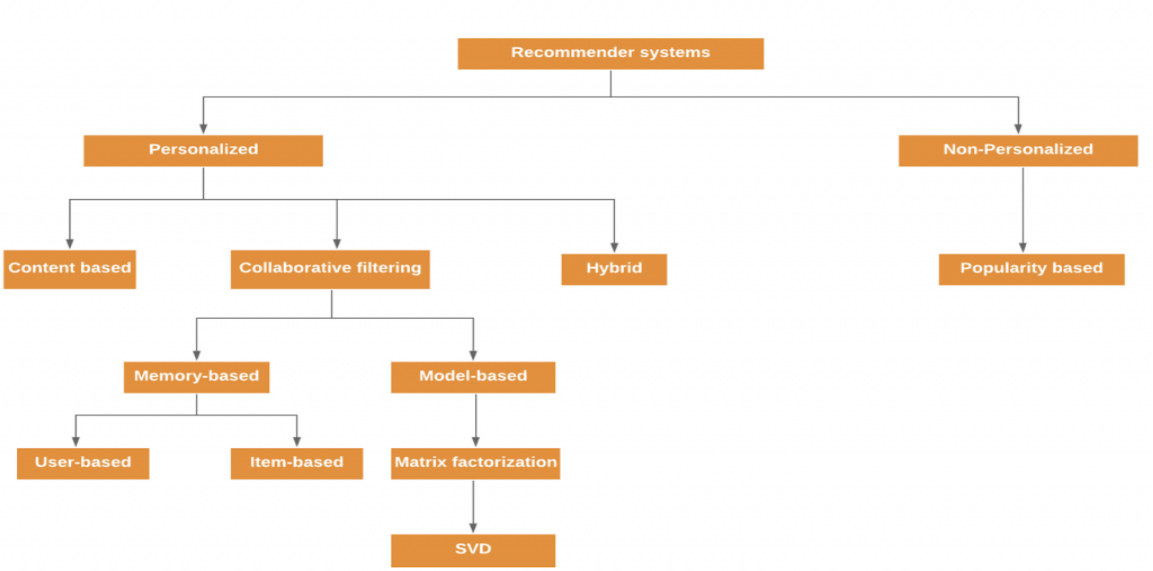


Fig:3.3 Detailed Types

- Recommender Systems:
  - Personalized:
    - Content-Based: Recommendations based on the similarity between items and user profiles.
    - Collaborative Filtering:
      - Memory-Based:
        - User-Based: Similar users' preferences are used for recommendations.
        - Item-Based: Similar items are recommended to users.
      - Model-Based:
        - Matrix Factorization: Techniques like Singular Value Decomposition (SVD) to identify latent factors from the interaction matrix.
    - Hybrid: Combining both content-based and collaborative filtering approaches for better recommendations.
  - Non-Personalized:
    - Popularity-Based: Recommendations based on the popularity of items, not personalized to individual users.

3.1.4 Decision Tree Approach

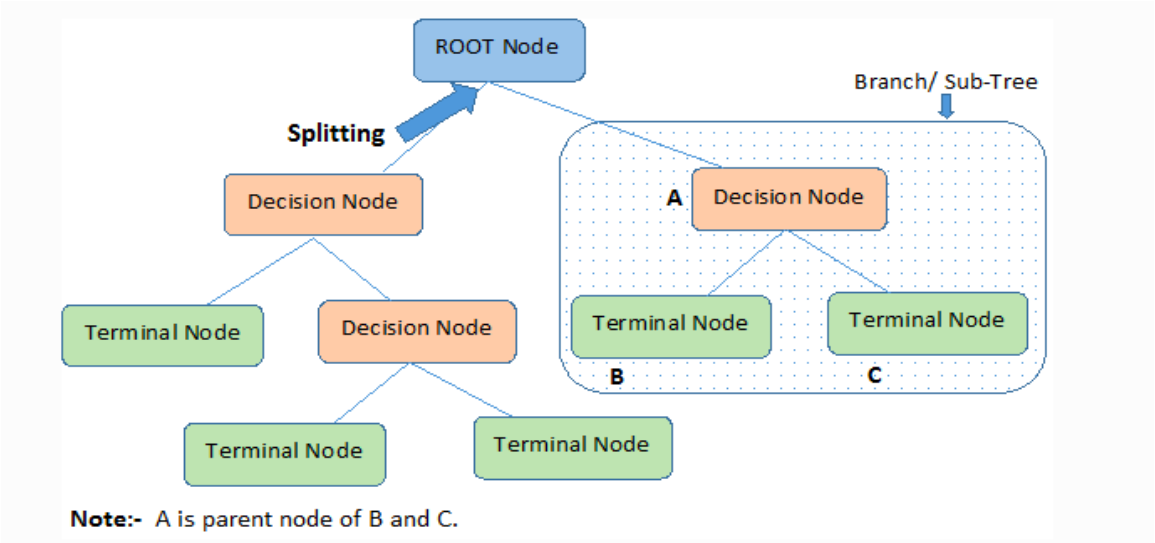


Fig: 3.4 Decision Tree

A decision tree is a flowchart-like structure used in decision-making and machine learning. Here's an explanation of the key components in the image:

- Root Node: This is the topmost node of the tree. It represents the entire dataset, which is split into subsets as you move down the tree.
- Splitting: This is the process of dividing a node into two or more sub-nodes based on certain criteria. Splitting continues until the data is perfectly classified or a stopping criterion is met.
- Decision Node: These are nodes where the data is split based on some condition. They represent intermediate decisions in the process.
- Terminal Node (Leaf Node): These are the end nodes of a tree that do not split further. They represent the final decision or outcome.
- Branch/Sub-Tree: A subsection of the entire tree, which can be considered as a tree in itself. It contains a decision node and all of its successor nodes.
- Parent Node: A node that has one or more child nodes. In the image, node A is the parent node for B and C.

3.1.5 NLP Approach

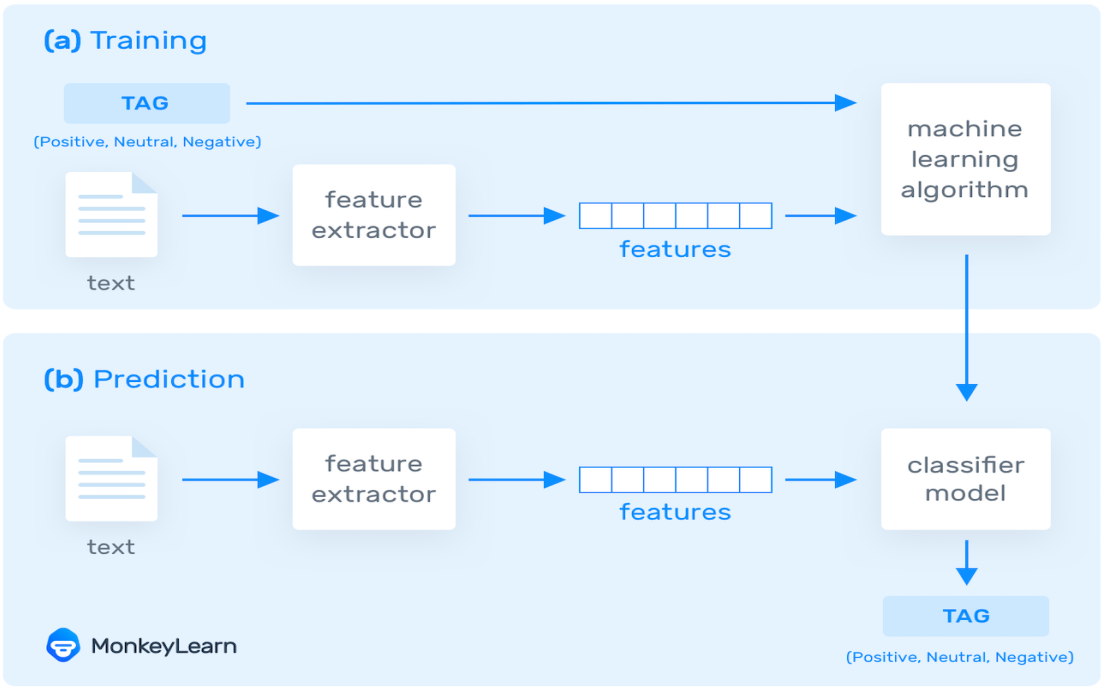


Fig: 3.5 NLP Approach

This image shows the process of text classification using machine learning, divided into two stages: training and prediction.

(a) Training Phase:

- Text: The input data, typically in the form of text documents.
- Feature Extractor: This component converts the text into numerical features that can be processed by the machine learning algorithm. Common techniques include Bag of Words, TF-IDF, or word embeddings.
- Features: The output of the feature extractor, which is a numerical representation of the text data.
- Machine Learning Algorithm: The algorithm used to learn patterns from the features and the associated tags. Examples include logistic regression, support vector machines, or neural networks.
- Tag: The labels or categories assigned to the text during training, such as Positive, Neutral, or Negative.

(b) Prediction Phase:

- Text: New text data that needs to be classified.
- Feature Extractor: The same feature extraction process is applied to the new text data to convert it into numerical features.

- Features: The numerical representation of the new text data.
- Classifier Model: The trained machine learning model that uses the features to predict the tag.
- Tag: The predicted label for the new text data, such as Positive, Neutral, or Negative.

Overall, the training phase involves learning a model from labelled data, while the prediction phase uses this model to classify new, unseen data.

3.2 Unique Features of the System

The system's unique features include:

1. **Industry Trend Analysis**

   The platform uses real-time data to analyse current industry trends in programming languages, providing students with up-to-date information on which languages are most in demand. This ensures that students can make informed decisions about which languages to learn based on the latest market needs.

2. **Clear and Concise Language Recommendations**

   Based on individual profiles that include academic background, interests, and career goals, the platform offers tailored programming language recommendations. These suggestions are easy to understand, helping students quickly identify which languages are most suitable for their objectives.

3. **A User-Friendly Interface for Easy Navigation**

   The platform features an intuitive interface designed for ease of use. Its layout ensures that students can effortlessly navigate through various features, access industry trend data, receive recommendations, and find learning resources without any technical hassle.

4. **Bridging the Gap Between Passion and Proficiency in Programming**

   By aligning students' interests with industry demands, the platform helps them develop proficiency in languages they are passionate about while also meeting market needs. This approach ensures that students can pursue their interests while enhancing their employability and relevance in the tech industry.

CHAPTER 4

REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATION

4.1 Feasibility Study

- **Technical Feasibility**: The project's technical feasibility is high, given the availability of web development frameworks that can support real-time data processing and analytics. The development team needs to have skills in web development, database management, and data analysis.

- **Economical Feasibility**: Economically, the project requires initial investment in web hosting, domain registration, and potentially premium APIs for accessing real-time industry trends. However, these costs are manageable and can be offset by potential monetization strategies, such as subscriptions or advertisements.

- **Operational Feasibility**: Operationally, the project is feasible as it aligns with the increasing demand for tech education resources. The platform's ease of use and the value it provides make it likely to be welcomed by its target users – students and learners interested in programming.

4.2 Software Requirement Specification

4.2.1 Data Requirement

- Real-time industry trends data.
- Comprehensive database of programming languages and their resources.
- User data for personalized recommendations.

4.2.2 Functional Requirement

- User account creation and management.
- Dynamic display of programming languages and trends.
- Search and filter options for languages and resources.
- Feedback and rating system for resources.

4.2.3 Performance Requirement

- The platform should be capable of handling high traffic volumes without significant lag.
- Real-time updates and notifications without system overload.
- Quick response time for user queries and actions.

4.2.4 Maintainability Requirement

- The system should be designed for easy updates and scalability.
- Regular updates for the tools and technologies used.
- Efficient error logging and debugging capabilities.

4.2.5 Security Requirement

- Implementation of secure authentication mechanisms.
- Encryption of sensitive user data.
- Regular security audits and compliance with data protection regulations.

4.3 SDLC Model to be Used.

The Agile SDLC model will be utilized for this project. Agile is suitable due to its flexibility, emphasis on customer feedback, and iterative approach, allowing for continuous improvement and adaptation to changing requirements. This model supports the dynamic nature of the project's goals, especially considering the need for regular updates based on real-time industry trends and user feedback.

Fig: 4.1 Agile Method (SDLC)

4.4 Detail Design

In the detail design phase, we focus on elaborating the architecture, components, and interactions of the system. This includes designing the database schema, defining APIs and endpoints, outlining user interfaces, and specifying the integration of external services or modules.

Components of Detail Design:

1. **Database Schema Design:** Define the structure of the database including tables, relationships, constraints, and indexes. Use MongoDB for scalability and flexibility in handling structured and unstructured data related to users, courses, progress tracking, and system configurations.
2. **API Design:** Design RESTful APIs for communication between frontend and backend components. Define endpoints for user authentication, course management, progress tracking, recommendation system interactions, and community features like forums and chats.
3. **User Interface Design:** Create wireframes and mock-ups for the user interface using tools like Figma or Adobe XD. Design intuitive and responsive interfaces for user registration, login, dashboard, course catalogue, progress dashboard, community forums, and admin panels.
4. **Integration Design:** Plan integration with external services such as NLP libraries for text analysis, machine learning models for recommendation systems, data visualization tools for analytics, and email services for notifications and alerts.

4.5 System Design

DFD Level 0: System Overview

- **Processes:** Represented as circles, processes include user authentication, course management, recommendation system, community interactions, and admin functionalities.
- **Data Stores:** Represented as rectangles, data stores include the MongoDB database for storing user data, course information, progress details, and system configurations.
- **External Entities:** Represented as squares, external entities include users, admins, external APIs for NLP and machine learning, and external databases if required.

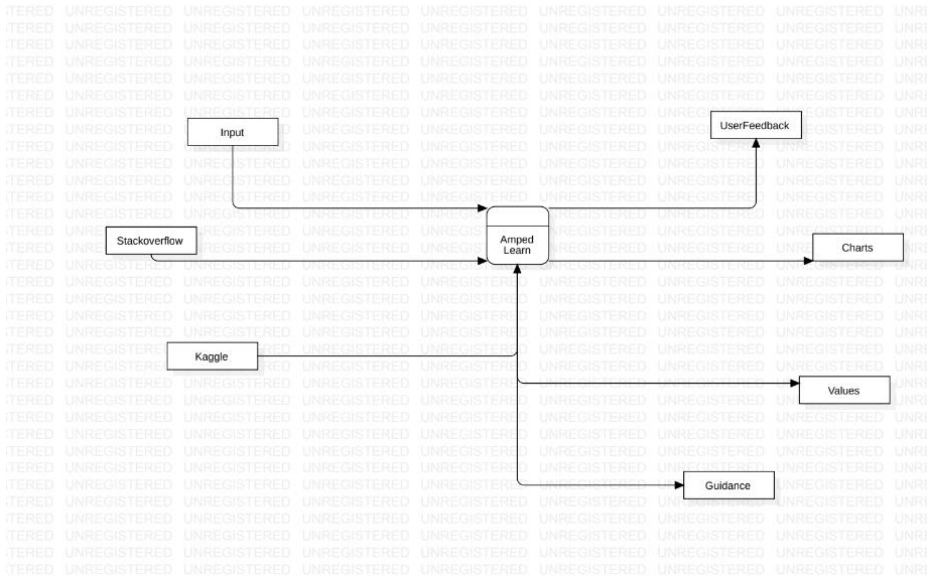


Fig: 4.2 Data Flow Diagram (level 0)

DFD Level 1: Detailed Process Decomposition

- **User Authentication Process:** Includes sub-processes for user registration, login, password management, and email verification.
- **Course Management Process:** Includes sub-processes for course creation, editing, deletion, and content updates.
- **Recommendation System Process:** Involves sub-processes for data analysis, recommendation generation, and personalized course suggestions.
- **Community Interaction Process:** Includes sub-processes for forum postings, chat interactions, user collaborations, and content sharing.
- **Admin Management Process:** Includes sub-processes for user management, content moderation, analytics monitoring, and system configurations.

These DFDs provide a structured view of the system's processes, data flows, and interactions at different levels of abstraction, aiding in understanding system functionalities and dependencies for further implementation and testing phases.

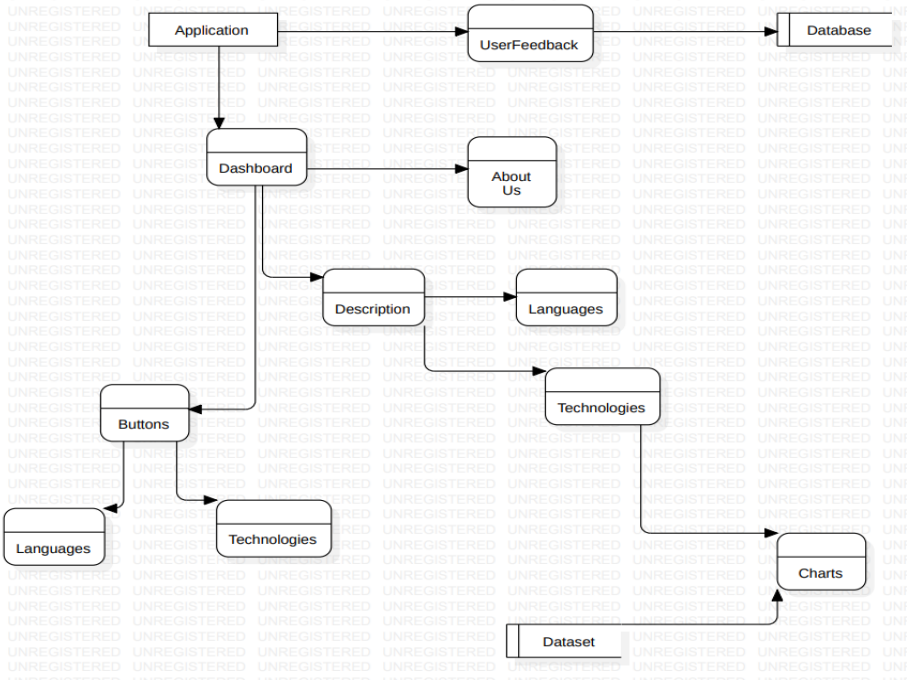


Fig: 4.3 Data Flow Diagram (level 1)

4.6 Database Design

For the MongoDB database design, we create two collections: "feedback" and "categories".

1. Feedback Collection: This collection stores individual user feedback, containing fields such as user identifiers, selected programming languages, optional comments, and timestamps of submissions. This design facilitates the understanding of user preferences and experiences, allowing for efficient data analysis.
2. Categories Collection: The "categories" collection maintains a count of each programming language selected by users. This count is incremented based on the feedback data received. Such a design enables the generation of charts and visualizations to track language popularity trends over time, offering valuable insights into current industry demands and user preferences.

By organizing data into these two collections, the MongoDB database effectively manages user feedback and enables the analysis of programming language trends.

CHAPTER 5
IMPLEMENTATION

5.1 Introduction to Tools and Technologies Used

In the development of our platform, we have carefully selected a range of languages, tools, and technologies to ensure robustness, scalability, and user-friendliness. Below is an overview of the key components used in the implementation:

Programming Languages:

- JavaScript: JavaScript plays a central role in both frontend and backend development. It is a versatile language known for its ability to create interactive and dynamic web pages. For backend development, we utilize Node.js, which allows us to run JavaScript server-side, enabling efficient handling of server requests and real-time data processing.

Frameworks:

- React: React.js is a powerful JavaScript library for building user interfaces. We leverage React for our frontend development to create modular and reusable components, resulting in a responsive and intuitive user experience.
- Express.js: Express is a minimalist web framework for Node.js, providing essential features for building robust web applications and APIs. We use Express to handle routing, middleware, and database interactions on the server side.

Web Technologies:

- HTML5 (Hypertext Markup Language): HTML5 forms the backbone of our web pages, defining the structure and content elements that users interact with.
- CSS3 (Cascading Style Sheets): CSS3 is used for styling and layout purposes, ensuring visual consistency and aesthetics across different devices and screen sizes.
- MongoDB: MongoDB, a NoSQL database, serves as our data store for storing user information, course data, and progress tracking. Its flexible schema and scalability make it ideal for our needs.
- NLP (Natural Language Processing): NLP techniques are employed to analyse and extract insights from textual data. This aids in building a recommendation system that suggests relevant courses and learning paths based on user preferences and industry trends.

- XGBoost and Decision Trees: Machine learning algorithms such as XGBoost and Decision Trees are utilized for data analysis, trend forecasting, and generating personalized recommendations for users.

By leveraging this stack of languages, frameworks, and technologies, we aim to create a feature-rich and user-centric platform that empowers learners to explore, compare, and excel in various programming languages and domains while providing valuable insights and guidance throughout their learning journey.

Chapter 6

TESTING, AND MAINTENANCE

6.1 Testing Techniques and Test Cases Used

6.1.1 Testing Techniques

- **Functional Testing:**

  Functional testing ensures that each function of the system operates as intended. In our testing approach, we rigorously examined key functions and modules to guarantee seamless performance.

- **Usability Testing:**

  Usability testing was employed to assess the user-friendliness of the system. Many criteria were considered during this testing phase to enhance the overall user experience.

- **Compatibility Testing:**

  To ensure widespread accessibility, we conducted compatibility testing across various browsers, operating systems, and devices. The application was tested on Google Chrome, Microsoft Edge, Windows 10, Desktop, Laptops ensuring consistent functionality.

- **Performance Testing:**

  While primarily focusing on manual testing, we incorporated performance considerations.

6.1.2 Test Cases

- **Test Case Design:**

  Our test case design followed a meticulous methodology to cover all aspects of system functionality. We used Microsoft Excel to document detailed test cases for comprehensive coverage.

- **Test Case Execution**

  Execution involved setting up a controlled environment with specific configurations. Each test case was executed systematically, and the outcomes were recorded for analysis.

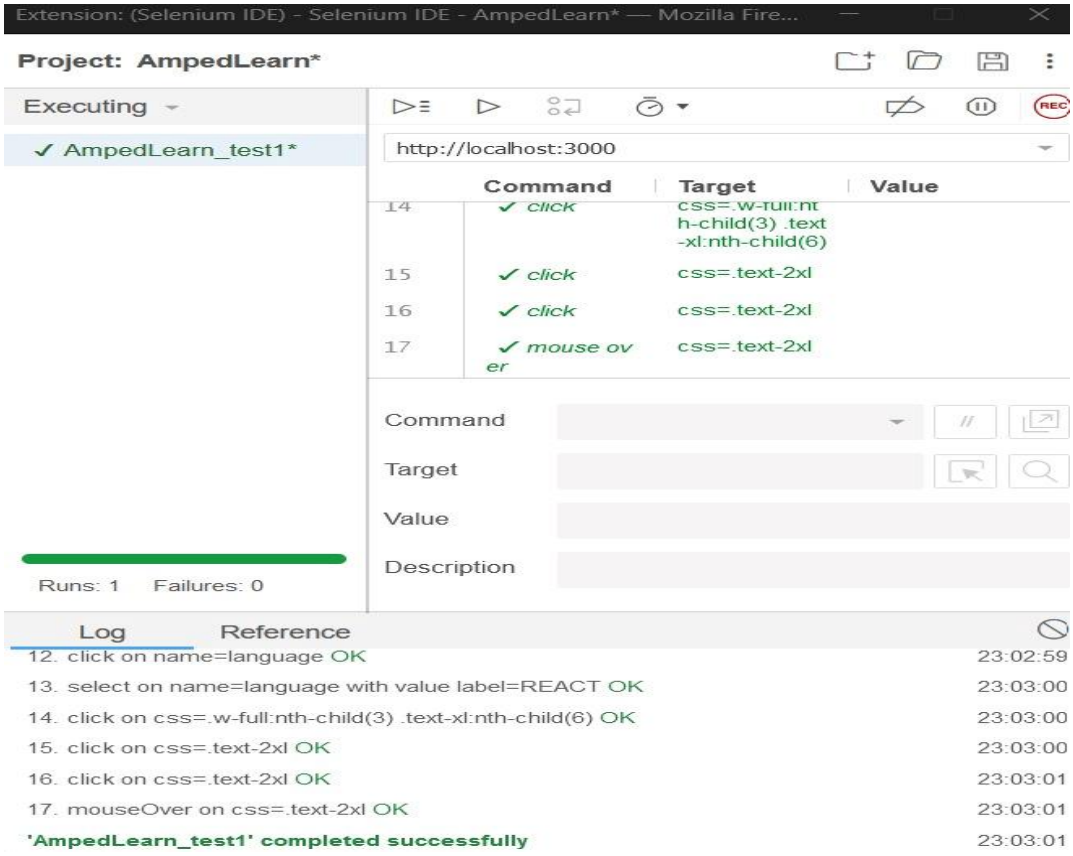


Fig: 6.1 Selenium Test Case

• **Results and Observations:**

Upon test case execution, we summarized results, highlighting any discrepancies, issues, or noteworthy observations. This comprehensive analysis forms the basis for refining and improving the system.

• **Regression Testing:**

To ensure that new updates or fixes did not adversely impact existing functionalities, we conducted regression testing. This step was crucial to maintain the integrity of the system across iterations.

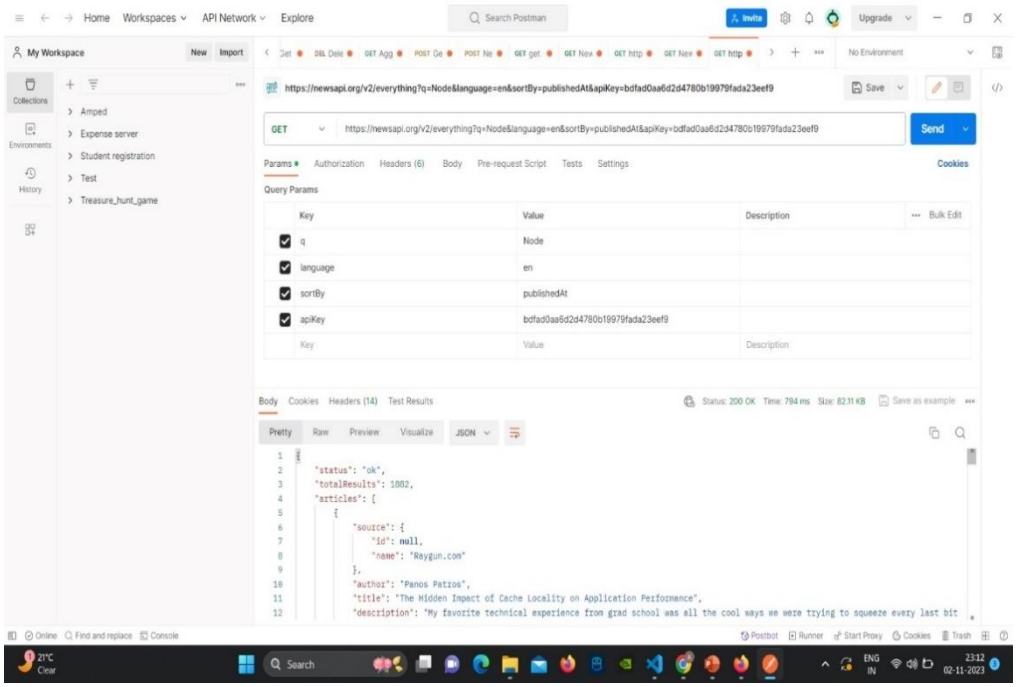


Fig: 6.2 API Testing Using Postman

- Endpoint: https://newsapi.org/v2/everything
- Parameters:
  - q: Node
  - language: en
  - sortBy: publishedAt
  - apiKey: [Your API Key]
- Response: Returns articles related to Node.js.

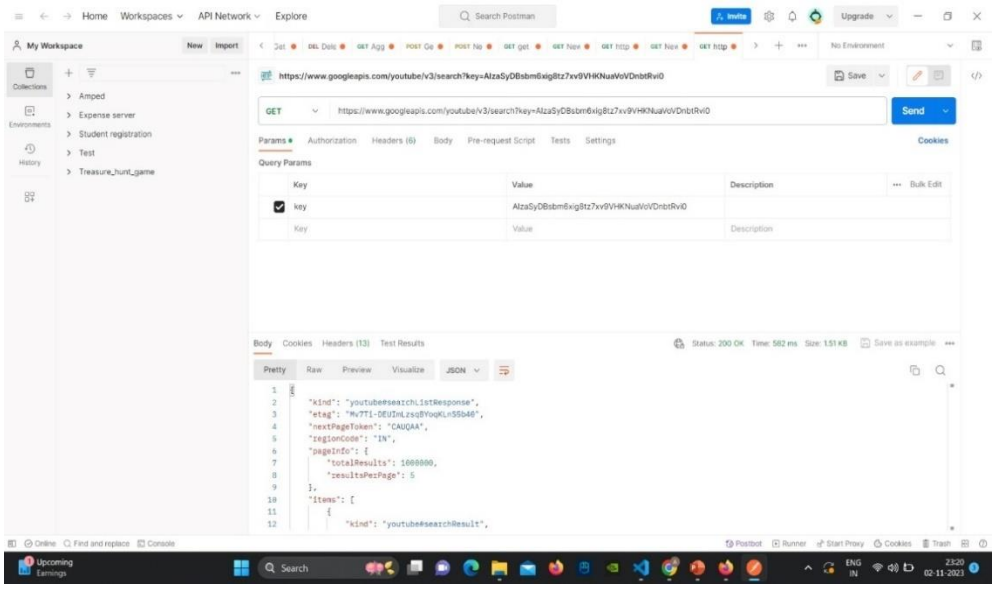


Fig: 6.3 API Testing Using Postman

- Endpoint: https://www.googleapis.com/youtube/v3/search
- Parameters:
    - key: [Your API Key]
- Response: Returns search results for YouTube videos.

6.2 Maintenance

The maintenance phase involves continuous monitoring, updates, and improvements to ensure the system's longevity and relevance. Regular checks for data quality, model performance, and software updates are implemented. User feedback and changing environmental conditions are considered for ongoing enhancements to the prediction system. Maintenance activities aim to address emerging challenges, implement security patches, and incorporate advancements in technology for sustained system effectiveness.

CHAPTER 7

RESULTS AND DISCUSSIONS

7.1 User Interface Representation

The user interface (UI) of the pollution forecasting system is thoughtfully designed for user-friendly interaction. It encompasses intuitive navigation, clear visualizations, and accessibility features to ensure a seamless experience for users. The design prioritizes displaying key information, trends, and alerts to facilitate effective decision-making.

Fig: 7.1 User Interface

7.1.1 Brief Description of Various Modules of the System

1. **Language Comparison and Analysis Module:**

    - This module allows users to compare different programming languages based on various parameters such as popularity, job opportunities, salary trends, and industry demand.
    - It integrates data analysis and visualization techniques to present insights in a user-friendly manner.

2. **Learning Resources Recommendation System:**

- Using natural language processing (NLP) techniques and machine learning algorithms, this module recommends learning resources (courses, tutorials, articles) based on user preferences, historical data, and industry trends.
- It provides personalized recommendations to help users choose the most suitable learning path for their goals.

3. **Community Interaction and Feedback Module:**

- Facilitating community engagement, this module includes features such as discussion forums, chat rooms, and peer-to-peer support channels.
- Users can interact with peers, share experiences, ask questions, and provide assistance, fostering a collaborative learning environment.

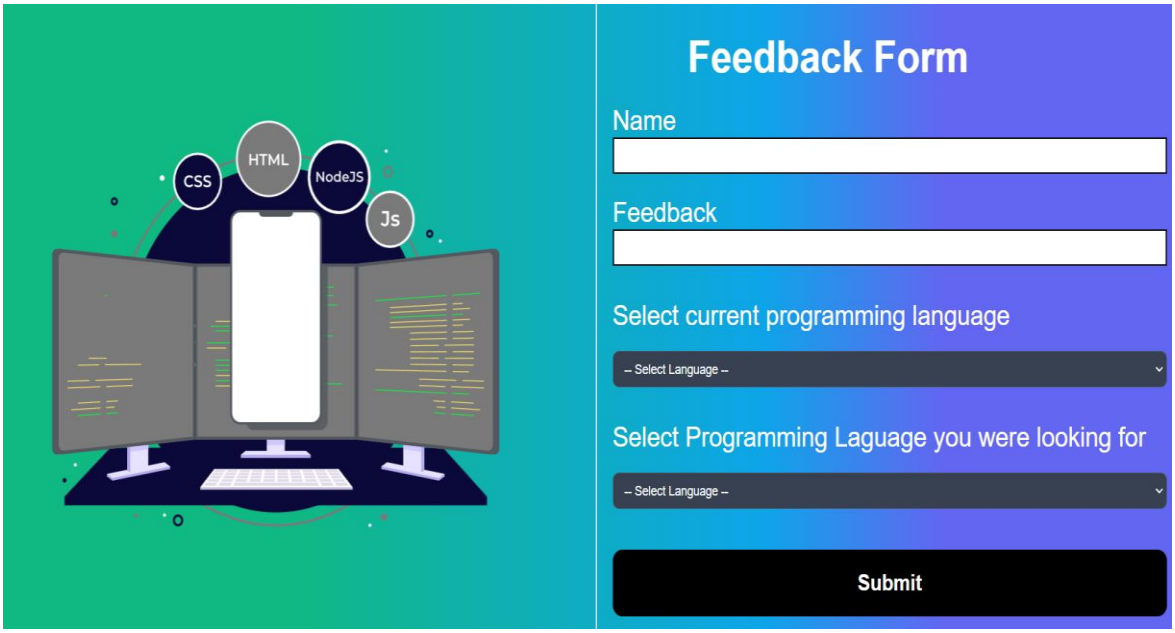


Fig: 7.2 Feedback Form

4. **Admin Dashboard and Content Management Module:**

- This module provides administrators with a centralized dashboard to manage users, content, and system configurations.
- Admins can add/edit/delete courses, monitor user activity, analyse system metrics, and ensure overall platform functionality and security.

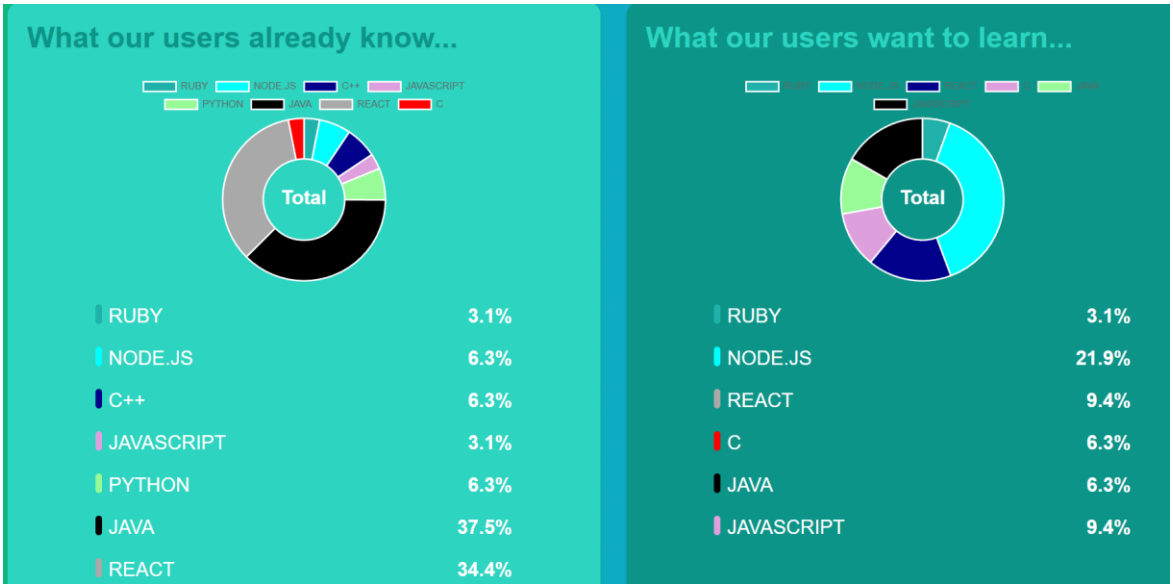


Fig: 7.3 Feedback Chart

5. **Real-time Data Analysis and Visualization Module:**

- Integrating data analytics tools and visualization libraries, this module generates real-time insights into language usage trends, job market demands, and user engagement metrics.

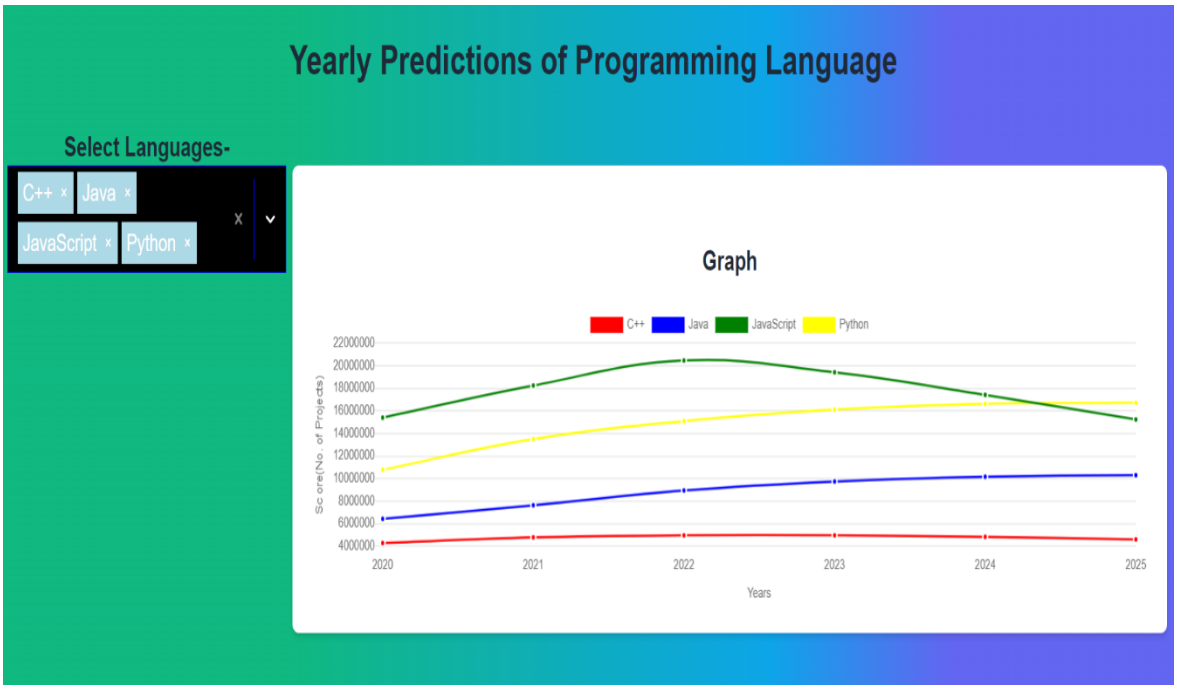


Fig: 7.4 Prediction Graph

- Interactive charts, graphs, and dashboards enhance data interpretation and decision-making for both users and administrators.

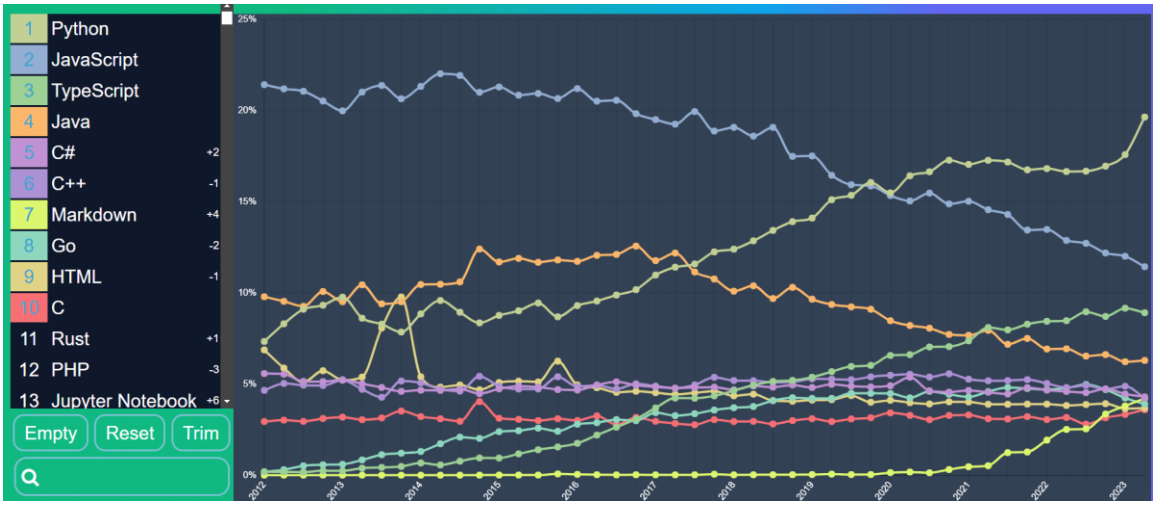


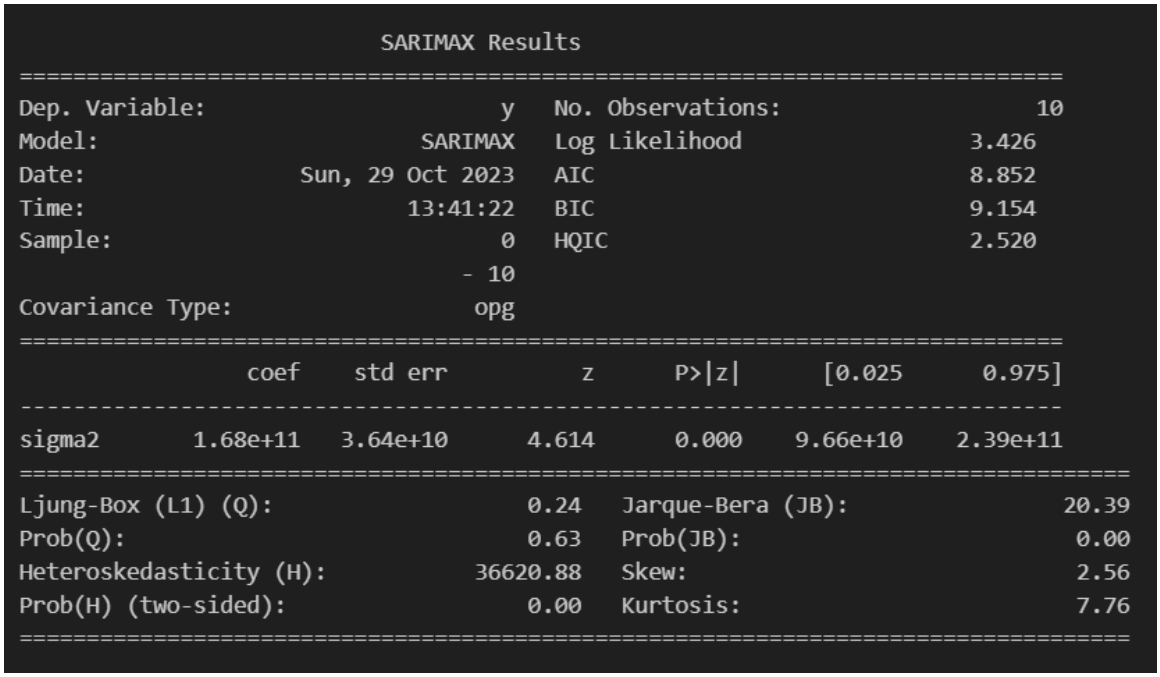
Fig: 7.5 Prediction Graph

These interconnected modules form the backbone of the system, providing a comprehensive learning ecosystem that empowers users with knowledge, skills, and networking opportunities in the tech industry.

7.2 Key Findings

The code aims to improve the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) values by refining the model parameters.

Enhanced visualizations are generated to represent the forecasted data effectively.



```
                           SARIMAX Results
==============================================================================
Dep. Variable:                      y   No. Observations:                   10
Model:                        SARIMAX   Log Likelihood                3.426
Date:                Sun, 29 Oct 2023   AIC                           8.852
Time:                        13:41:22   BIC                           9.154
Sample:                             0   HQIC                          2.520
                                  - 10
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
sigma2      1.68e+11   3.64e+10      4.614      0.000    9.66e+10    2.39e+11
===================================================================================
Ljung-Box (L1) (Q):                   0.24   Jarque-Bera (JB):                20.39
Prob(Q):                              0.63   Prob(JB):                         0.00
Heteroskedasticity (H):           36620.88   Skew:                             2.56
Prob(H) (two-sided):                  0.00   Kurtosis:                         7.76
===================================================================================
```

Fig: 7.6 ML Result

Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots are instrumental tools in the exploration and modelling of time series data.

These plots act as visual representations of the correlation between a time series and its lagged versions, offering valuable insights into the underlying structure of the data.

One of the key aspects to analyse in ACF and PACF plots is the presence of significant peaks that fall outside the designated confidence interval. These peaks signify the existence of statistically relevant relationships between the current value of the time series and its past values at specific lags.

7.3 Database Design

MongoDB database comprises two collections to effectively manage user feedback and analyze programming language trends.

The **feedback** collection stores individual user feedback, including user identifiers, selected programming languages, optional comments, and timestamps of submissions. This data helps in understanding user preferences and experiences.

The **categories** collection helps maintain a count of each programming language selected by users, incremented based on the feedback data. This enables the generation of charts and visualizations to track language popularity trends over time, providing valuable insights into current industry demands and user preferences.

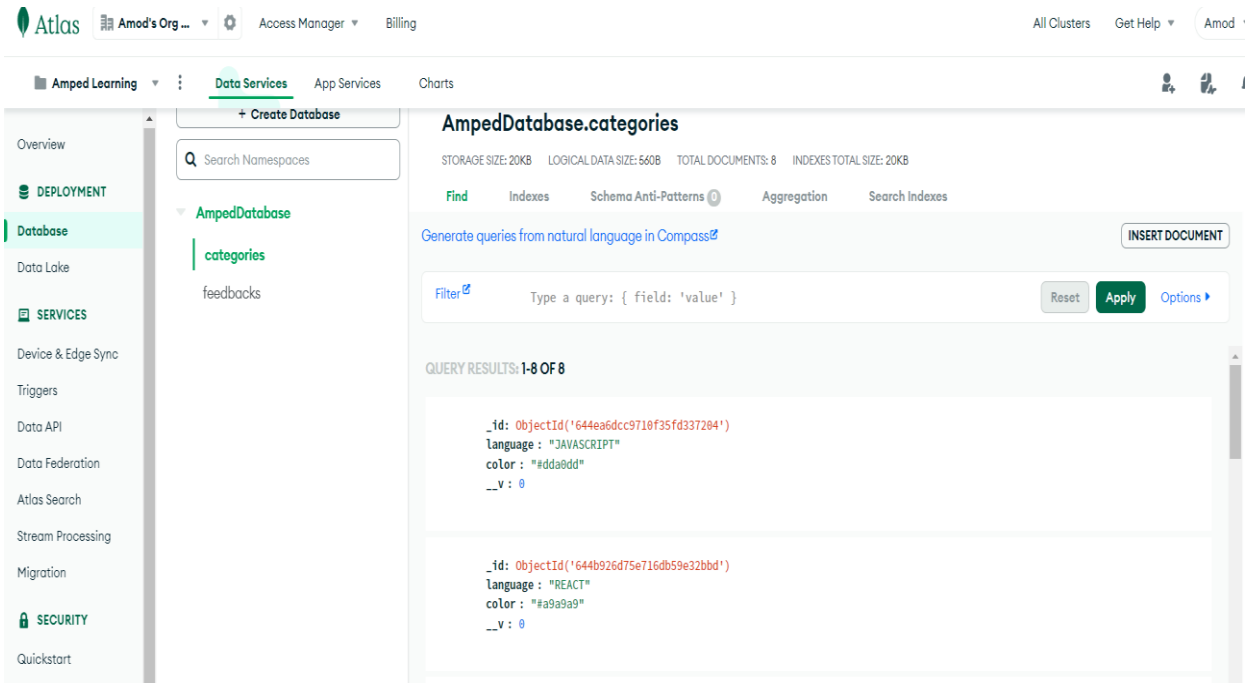


Fig: 7.7 Database Snapshot

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

8.1 CONCLUSION

The final platform would be able to overcome all the problems that students are currently experiencing in their learning. The students will have a clear idea of which domain to choose and a perfect road map to help them achieve their goals and land a job in an MNC.

The recommendation system will help students understand which domains are currently thriving and what job profiles are needed in the industry.

This allows the student to easily plan their journey ahead and learn something that they are truly interested in rather than something that they will not feel like continuing in the future.

8.2 Future Scope

Mobile App Development:

Develop a mobile app version of the platform to cater to users' on-the-go learning needs. Ensure seamless synchronization of progress, notifications, and access to learning materials across multiple devices.

Enhanced Learning Paths:

Expand the platform to offer curated learning paths tailored to specific career goals and industry sectors. Integrate advanced analytics to track user preferences and suggest customized learning journeys.

Advanced Certification Programs:

Develop advanced certification programs in collaboration with industry experts and organizations. Offer specialized certifications in emerging technologies and niche domains to enhance user credibility and career prospects.

REFERENCES

1 A review and analysis of technologies for developing web applications, BY-SOLOMON ANTONY

2 Web Application Development -a study on UML Web Application Extension, Andreas Oskarsson, Martin Kling, Tobias Norberg

3 Research on HTML5 in Web Development, Ch Rajesh, K S V Krishna Srikanth

4 Review of WordPress website vs coding website By-Swapnil S More

5 Website Development Technologies: A Review By - Pratiksha D Dutonde , Shivani S Mamidwar , Monali Sunil Korvate

6 Wide & Deep Learning for Recommender Systems

Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, Hemal Shah

7 Research Paper Recommendation System Using Natural Language Processing Submitted in partial fulfilment of the requirements for the degree of Bachelor of Technology in Computer Science & Engineering by Siddharth Sanjay Gandhi 19BCE0005 Under the guidance of Dr. Akila Victor School of Computer Science & Engineering VIT Vellore Dr. Felix Dietrich, Department of Informatics Technical University of Munich

8 RESEARCH ON COVID-19 EPIDEMIC BASED ON ARIMA MODEL Li Zhihao et al 2021 J. Phys.: Conf. Ser. 2012 012063

9 Forecasting with Machine Learning

Dr. Osman Mohamed Abbas

10 Research Paper On Recommendation System Dr. Alka Singhal Assit Professor,CSE department, IPEC Ghaziabad, India Shivangi Rastogi Student, CSE department, IPEC Ghaziabad, India Nikhil Panchal Student, CSE department, IPEC Ghaziabad, India Shivani Chauhan Student, CSE department IPEC Ghaziabad, India Shradha Varshney Student, CSE department, IPEC Ghaziabad, India

11 Dibekulu, Dawit. (2020). An Overview of Data Analysis and Interpretations in Research. 1-27. 10.14662/IJARER2020.015.

12 Lubbers, Peter & Albers, Brian & Salim, Frank & Kumar, Mahesh. (2023). HTML5 -Unleashing the Power of the Web Reference By:-"HTML and CSS: Design and Build Websites" by Jon Duckett and "Pro HTML5 Programming" by. 10.13140/RG.2.2.11587.37923.

13 Use of React.js in Modern Web Development and its Comparative Analysis Kunal Goyal, Dr. Vishal Shrivastava , Dr. Akhil Pandey, 2024

14 React-JS: A Cutting-Edge Framework for Web Designing Navratan Mal, Dr. Vishal Shrivastava, Dr. Akhil Pandey. 2024

15 A Comparative Analysis of Database Management Systems for Time Series Data TOVE VERNER-CARLSSON VALERIO LOMANTO, 2023

16 Malik, Pankaj & Dangi, Aditya & Singh, Aditya & Asst, Thakur & Pratap, Aditya & Parihar, Singh & Sharma, Utkarsh & Mishra, Lakshya. (2023). An Analysis of Time Series Analysis and Forecasting Techniques. IJARCCE. Vol-9. 2023.

17 IMPLEMENTATION AND COMPARISON OF MERN STACK TECHNOLOGY WITH HTML/CSS, SQL, PHP & MEAN IN WEB DEVELOPMENT Er. Vikas Goyal, Ashish Kumar Mishra, Daljeet Singh, 2023

18 Predicting Stock Prices Using Hybrid LSTM and ARIMA Model Chi Ma, Jie Wu* , Hui Hu, YueNai Chen, JingYan Li, 2023

19 Github- https://github.com/KIET-Github/CS-2024-A

Proof of patent publication (Screenshot of Publication)

Office of the Controller General of Patents, Designs & Trade Marks
Department for Promotion of Industry and Internal Trade
Ministry of Commerce & Industry,
Government of India

INTELLECTUAL PROPERTY INDIA
PATENTS | DESIGNS | TRADE MARKS | GEOGRAPHICAL INDICATIONS

Application Details	
APPLICATION NUMBER	202411002782
APPLICATION TYPE	ORDINARY APPLICATION
DATE OF FILING	14/01/2024
APPLICANT NAME	1 . Harsh khatter 2 . Himanshu Kumar 3 . Amod Katiyar 4 . Aniket Bhardwaj 5 . Gaurav Dubey
TITLE OF INVENTION	ADAPTIVE ONLINE PLATFORM FOR ENHANCED TEACHING AND LEARNING
FIELD OF INVENTION	PHYSICS
E-MAIL (As Per Record)	harsh.khatter@kiet.edu
ADDITIONAL-EMAIL (As Per Record)	
E-MAIL (UPDATED Online)	
PRIORITY DATE	
REQUEST FOR EXAMINATION DATE	--
PUBLICATION DATE (U/S 11A)	02/02/2024

Application Status	
APPLICATION STATUS	Awaiting Request for Examination

View Documents

Filed → Published → RQ Filed → Under Examination → Disposed

In case of any discrepancy in status, kindly contact ipo-helpdesk@nic.in

Patent Proof

Proof of Research Paper Submission (Screenshot of Acceptance)

Most Urgent || IJSRD || Article Acceptance Inbox

IJSRD Journal 28/12/2023

to bcc: me

Dear Author(s),

Your paper is accepted in our journal.
It is our humble duty to inform /remind you regarding paper status. We have selected your paper. To complete it please follow the remaining process of Paper publication.

Adaptive Online Platform for Enhanced Teaching and Learning

Dr. Gaurav Dubey
Computer Science
KIET Group of Institution
Ghaziabad, Uttar Pradesh, India
gaurav.dubey@kiet.edu

Amod Katiyar
Computer Science
KIET Group of Institution
Ghaziabad, Uttar Pradesh, India
amod.2024cs1058@kiet.edu

Himanshu Kumar
Computer Science
KIET Group of Institution
Ghaziabad, Uttar Pradesh, India
himanshukumarrai786@gmail.com

Aniket Bhardwaj
Computer Science
KIET Group of Institution
Ghaziabad, Uttar Pradesh, India
aniket.2024cs1199@kiet.edu

***Abstract*—For students today, the ever-expanding array of educational options poses a special difficulty. Even with the abundance of resources and routes available, navigating this intricate environment can be daunting. The important questions of "what to learn" and "how to learn it effectively" are ones that students struggle with. An innovative approach to closing this knowledge gap is presented in this research. It is a brand-new platform designed to help students get the most of their educational experiences, especially when it comes to learning programming languages. When faced with the age-old dilemma of "what to learn," students frequently find it difficult to set priorities among the many programming languages available. By using data-driven insights, our platform seeks to resolve this conundrum by highlighting the most in-demand programming languages in the market.This article clarifies the platform's architecture, methodology, and functionality, highlighting its potential to improve learning outcomes and give students essential skills for success in the workplace.**

***Index Terms*—Educational opportunities, Programming language , Platform, Data-driven insights, Industry demands, Methodology, Architecture, Functionality, educational experiences, Professional success.**

I. Introduction

The landscape of education has been permanently altered by the digital revolution. Technological developments have democratized access to knowledge like never before, spurred by the internet and mobile technology. These days, a huge library of educational resources is available online for use on laptops, tablets, and smartphones. Geographical restrictions have been eliminated by this increased accessibility, resulting in a fully international learning environment where anyone with an internet connection can enroll as a student and research or learn anything.Research can be considered as an area of investigation to solve a problem within a short period of time or in the coming long future.[11] But there's a hidden problem with all of these instructional resources: the paradox of choice. When presented with an infinite number of educational options, students may experience feelings of overload. The important issues of "what to learn" and "how to learn it effectively" are prevalent and frequently throw a pall over a student's academic career.

This paper presents an innovative approach to this pressing problem. We see a platform created using MERN stack especially to give students more authority in the era of digital education.The MERN stack is a powerful and popular technology stack for building web applications. The word MERN form is M for MongoDB E for Express R for ReactJS and N for NodeJS.[16] Our platform addresses the challenge of "what to learn" by highlighting the most in-demand programming languages in the market through the use of data-driven insights. Students get understanding of the abilities that transfer into practical employment chances as a result. Beyond just identification, the platform acts as a whole learning path for these abilities.

II. Literature Survey

This review of the literature looks at pertinent studies on machine learning and web development, with an emphasis on topics that apply to our suggested platform, which helps students select and learn programming languages.

A. *Web Development*

Web technology refers to the tools, languages, and frameworks used to create and manage websites and web applications.[16] Numerous studies (e.g., [1] Antony, 2012; [2] Oskarsson et al., 2002; [3] Rajesh Krishna Srikanth, 2014; [5] Dutonde et al., 2022) examine different web development tools. These studies offer insightful information about the benefits and drawbacks of various frameworks and languages (such as HTML5, UML, and WordPress) that can be taken into account when developing our platform. Knowing these

technologies enables us to choose the best course of action for ensuring the functioning, scalability, and user experience of the platform.

B. Machine Learning

Data analysis is changing the collected row data into meaningful facts and ideas to be understood either qualitatively or quantitatively. [11] Research on machine learning-based recommendation systems ([6] Cheng et al., 2016; [7] Gandhi, 2023; [9] Singhal et al., 2021) is especially pertinent to our endeavor. These studies show how machine learning may effectively recommend products or information to people based on their past actions or interests. Using these methods, we may advise students on programming languages according to their interests, objectives for their careers, and current skill levels. By adding domain-specific knowledge about programming languages, job market trends, and individual learning styles, our approach can outperform current recommendation systems.

Our project can benefit indirectly from the forecasting with machine learning papers ([8] Li et al., 2021; [10] Abbas, 2018). These studies demonstrate the ability of machine learning to forecast future trends, even when they do not offer linguistic recommendations directly. We may be able to incorporate these methods to predict which programming languages will be in demand in the labor market, assisting students in making well-informed choices regarding their educational pathways. The analysis and interpretation of data represent the application of deductive and inductive logic to the research process.[11] In this case, we might enhance current approaches by concentrating on labor market information unique to the programming language space and integrating real-time updates to guarantee the most precise projections.

The literature research indicates that current web development tools offer a strong basis on which to create our platform. But by using machine learning techniques to customize the user experience, we hope to go beyond just providing facts.

We plan to use recommendation algorithms (see [6] Cheng et al., 2016 for inspiration) on our platform to select programming languages based on the needs and career objectives of each learner. This tailored method differs from the general suggestions that are frequently present on current platforms. Furthermore, we intend to investigate how machine learning might be used to predict future trends in the labor market (as in [8] Li et al., 2021), providing students with insightful knowledge about how the IT industry is changing.

Although extant research offers significant perspectives on web development technologies, recommendation systems, and machine learning methodologies, a discernible deficiency persists in the amalgamation of these domains to tackle the particular predicament of programming language acquisition among students.

III. Methodology

HTML5, the fifth iteration, is a significant milestone in the evolution of web technologies [12]. HTML5 is a new standard for HTML which allows us to build rich and interactive web pages which bring HTML into the world of application development started in the year 2004. HTML moves from simply describing the basics of a text based web for presenting audio, video and animations to enabling offline functionality, geo location and local storage in client side databases [3].

We'll be developing our web user interface with React. React.js offers a number of benefits for front-end developers. One of the main benefits is its declarative nature-, which allows developers to easily describe how their UI should look based on changes in data [13].

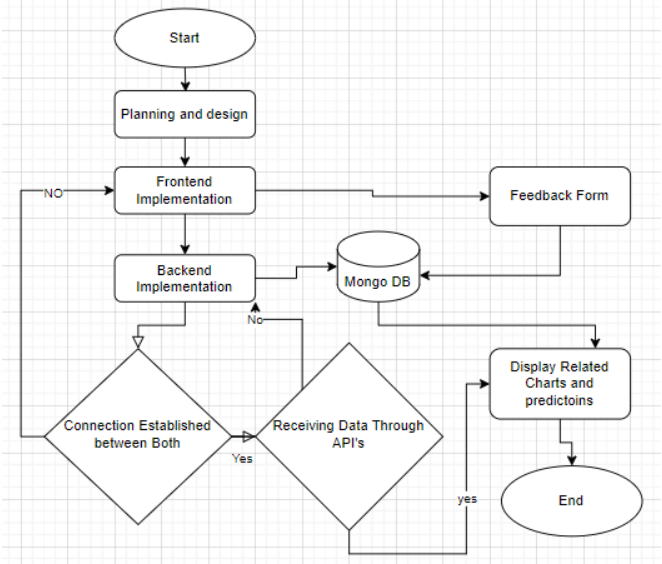


Fig. 1. Flowchart

We'll start with the most popular languages. After that, we'll use NLP and API to get the top programming languages off the internet .All of this will be done to help the student make his own decisions and provide him the tools he needs to be successful. A recommender system can be viewed as a search ranking system, where the input query is a set of user and contextual information, and the output is a ranked list of items. Given a query, the recommendation task is to find the relevant items in a database and then rank the items based on certain objectives, such as clicks or purchases [6].
A recommender system essentially works as a customized search engine for recommendations. It creates a prioritized list of pertinent items after taking into account the user data and context as the "query." Prioritization is based on goals such as clicks and purchases from users. Basically, there are three types:

A. *Collaborative filtering*

They power recommendation systems by leveraging user similarities. These systems analyze user ratings or preferences to identify users with similar tastes. This underlying assumption is that users who agreed in the past are likely to have similar future preferences. A key strength is the ability to recommend items without relying on detailed product descriptions, making it suitable for various domains. Essentially, collaborative filtering uses past user behavior to predict what a user might like based on their similarity to others [9].

B. *Content-based filtering*

Content-based filtering methods analyse the textual content of research papers,such as titles,abstracts,or full texts,to identify similarities and generate recommendations.These methods rely on extracting features from the documents and calculating the similarity between them [7]. The system then compares item profiles to user preferences, recommending items with the most similar profiles. However, this method is limited to items with well-defined features and struggles with conceptually similar items lacking explicit feature overlap.

C. *Hybrid approaches*

They blend collaborative and content-based screening methods. By doing this, they are able to take advantage of the advantages of both: collaborative filtering for tailored recommendations based on user similarities and content-based filtering for new things or users. Each method's contributions can be weighted according to the circumstances; for new products with few user ratings, content-based filtering might be given more weight. Hybrid filtering may be able to produce more precise and tailored recommendations by fusing various recommendation systems, but the process of creating and putting into practice these recommendations may become more difficult because of how the various strategies must be combined.

IV. Systems

Web applications tend to be multi-tiered by nature, with the most common structure being the three-tiered architecture. In its most common form, the three tiers are (i) Presentation layer, (ii) Application layer and (iii) Storage layers. [1]

Web applications can be implemented either through coding website or through Content Management System (CMS) tool WordPress. WordPress is a free CMS tool which helps us to build a website very easily also it supports lack of plug-ins whereas Coding Website is hard to code but reliable. [4]

We will deploy a model of recommended systems that will analyze historical data and trends to determine the popularity and usage of languages and then recommend it to students. Data analysis proves to be crucial in this process, provides a meaningful base to critical decisions, and helps to create a complete dissertation proposal [11].

NLP (Natural Language Processing) techniques power recommendation systems for research papers. By analyzing the text of titles, abstracts, or even full papers, these systems can identify relevant content and suggest similar papers to users. This analysis involves extracting key features, like keywords and concepts, from the documents. These features are then used to calculate the similarity between different papers, allowing the system to recommend related research based on the user's interests.

.Content-based recommenders use a variety of machine-learning algorithms, including Naive Bayes, support vector machines, decision trees, and kNN. As bag-of-words and vector represen- tations can have hundreds or thousands of dimensions, techniques as Latent Dirichlet Allocation (LDA) are often adopted. The content may also require natural language processing (NLP) tech- niques to make use of semantic and syntactic char- acteristics.

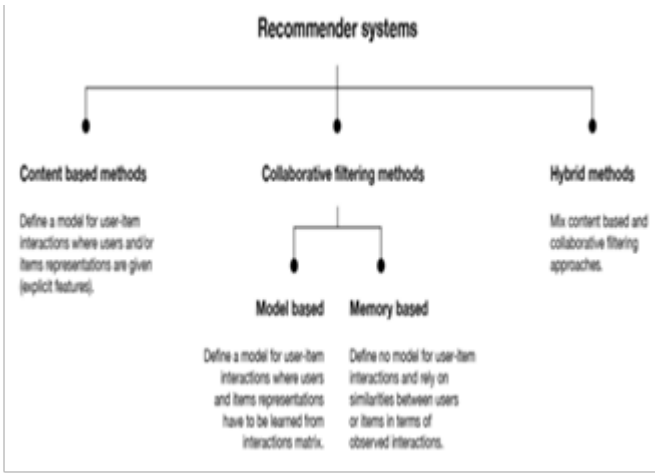


Fig. 2. Types of Recommender system Algorithms

In the last two decades, machine learning models have drawn attention and have established themselves as serious contenders to classical statistical models in the forecasting community. These models, also called black-box or data driven models, are examples of nonparametric nonlinear models which use only historical data to learn the stochastic dependency between the past and the future. For instance, Werbos found that Artificial Neural Networks (ANNs) outperform the classical statistical methods such as linear regression and Box-Jenkins approaches. [10]

Additionally, some machine learning models boast the ability to automatically extract relevant features from the data, reducing the need for manual feature engineering – a time- consuming and expertise-dependent process. [10]

V. Implementation (Website)

At the presentation layer, the web pages are rendered to the browser. Traditionally, the webpages contain only HTML

code. Now a days, the presentation layer of the web applications provide nearly the same user experience as desk top applications. [1]

The basic advantage for the developers and browsers is that they would be able to do more without the need of mastering multiple proprietary technologies that can develop rich web pages, enhanced forms and web based applications. [3]

Wordpress can be used for development in certain settings, however it does have some limitations. Some of the common issues are listed below: (i) The issue with the White Screen of Death, (ii) WordPress Memory Cap

Normally no issues are occurred when we run coding website, Whereas WordPress website sometime occurs issues, to avoid it all plug-ins should be updated timely also the theme should be up to date. [4]

The Development of a website is dependent upon major technologies : (i) Frontend Technologies, (ii) Backend Technologies, (iii) Databases.

A. *Frontend Technologies*

JavaScript is a Scripting language. It's principally abbreviated as JS. It is aforementioned that JavaScript is that the updated version of the ECMA script. JavaScript could be a light-weight, cross-platform, and taken scripting language. [5]

ReactJS's virtual DOM a memory-based clone of the genuine DOM — improves efficiency by updating the original DOM only when necessary. By allowing HTML-like syntax to be used within JavaScript code, the JSX JavaScript extension makes the process of creating ReactJS code simpler [14].

B. *Backend Technologies*

Back-end development focuses on the server-side aspects of a net site—an internet site—a web site or web application. this kind of development cares with web site design, scripting, and communication with databases. Back-end code permits the communication between browsers and data from databases. Back-end development works in conjunction with front-end development to supply users with a useful and interactive expertise.[5]

C. *Databases*

Database is that the assortment of inter-related knowledge that helps in economical retrieval, insertion and deletion of information from information and organizes the info within the style of tables, views, schemas, reports etc. SQL databases square measure structured, and NoSQL databases don't seem to be structured.[5]

We conclude that certain queries, and their corresponding real-world use cases, may be better suited for one of the two DBMSs due to the alignment between query structure and the strengths of that system [15]. Scaling SQL databases can be challenging, often involving vertical scaling (adding more resources to a single server) or complex sharding setups. [17].

VI. Implementation (Recommendation system)

Recommendation systems employ an automatic information filtering mechanism to suggest an item based on the user's preferences. By filtering the data source and providing consumers with pertinent information, it helps users. Collaborative filtering and content-based techniques are the two primary methods for developing recommendation systems.Information has increased as a result of the growth of the internet, particularly mobile Internet.

More than 80recent years. With the increase of information, the access of people to useful information is more difficult. Hence, the role of recommendation systems have become inevitable. [9]

A. *Collaborative filtering*

a) Data Acquisition through extracting GitHub Event Data:: The initial stage of the process involves data acquisition. Our code delves into a treasure trove of JSON files, each one meticulously recording GitHub events. These events encompass a wide range of interactions, including issues raised, pull requests submitted, and stars bestowed upon repositories. By meticulously parsing through these JSON files, we extract the valuable data they contain.

Basically collaborative filtering is based on collecting and analyzing a large amount of information on users' behaviors, activities or preferences and predicting what users will like based on their similarity to other users. [9]

B. *Data Consolidation by building a Unified Dataset:*

After extracting the data from each of the separate JSON files, we start the process of consolidating the data. This is an important stage where the data extracted from each individual file is combined into a single, cohesive dataset. The basis for further data processing and analysis is provided by this combined dataset. Through data aggregation, we generate an all-encompassing image of user activity and code contributions across several repositories. Exploring trends and patterns within the data in a more comprehensive way is made possible by this cohesive structure.

C. *Data Structuring by tailoring for Time Series Analysis*

Following a successful data consolidation, we move on to data structuring. In this step, the raw data must be converted into a format made especially for time series analysis. In order to do this, we carefully classify the data according to three important dimensions: (i) Year, (ii) Name of Language, (iii) Numbers.

By means of this methodical data structuring procedure, we convert the unprocessed data into a structured format that is intended to reveal the mysteries buried within the time series. Strong time series analysis methods can be applied thanks to this organized data.

D. *Time Series Analysis:*

Auto-Regressive Integrated Moving Average (ARIMA) models are utilized for time series analysis. The basic form of the ARIMA model is ARIMA (p, d, q), "p" is autoregressive coefficient, "d" is the order of difference made when the time series becomes stationary, "q" is number of moving average terms. First perform stationarity test on the time series, if the sequence is not stationary, use methods such as difference and logarithm to make the sequence stationary. Then use the autocorrelation function (ACF) graph and partial autocorrelation function (PACF) graph recognize and rank the model [8]. The ARIMA models are applied to predict the future trends in language popularity.

E. *Model Parameter Tuning:*

The ARIMA model parameters—order and seasonal order, among others—are adjusted for every language in order to maximize forecast accuracy. The evaluation metric used by the algorithm to identify the optimal model is Mean Absolute Error (MAE).

F. *Grid Search for Optimization:*

A set of hyperparameters and their values are feed to it first and then run an exhaustive search overall all possible combination of given values then training the model for each set of values is done. Then Grid Search algorithm will compare the score of each model it trains and keeps the best one. A common extension of Grid Search is to use cross-validation i.e., training the model on several different folds with different hyperparameter combinations to find more accurate results [9] .

A dependable method for determining the ideal hyperparameter combination is Grid Search. However, it can take a lot of time to evaluate each and every combination in cases with a large number of hyperparameters. Another strategy known as Randomized Search appears to address this.

G. *Best Model Selection via MAPE:*

Following the meticulous fine-tuning of ARIMA models for each language under investigation, a critical step emerges: the selection of the optimal model for forecasting purposes. While the selection process might intuitively gravitate towards the model exhibiting the lowest absolute error, a more nuanced approach is necessary. This is where the utility of Mean Absolute Percentage Error (MAPE) becomes apparent.

VII. Forecasting and Visualization:

We produce projections for language counts in the years 2023, 2024, and 2025 using the best models that were chosen. After that, compelling visualizations are made to convert the numerical data into distinct trends and show possible increases, decreases, or stability in language popularity.

A. *Results*

By optimizing the model parameters, the algorithm seeks to increase the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) values. Improved visuals are created to accurately depict the predicted facts.

SARIMAX Results

Dep. Variable:	y	No. Observations:	10
Model:	SARIMAX	Log Likelihood	3.426
Date:	Sun, 29 Oct 2023	AIC	8.852
Time:	13:41:22	BIC	9.154
Sample:	0	HQIC	2.520
	- 10		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
sigma2	1.68e+11	3.64e+10	4.614	0.000	9.66e+10	2.39e+11

Ljung-Box (L1) (Q):	0.24	Jarque-Bera (JB):	20.39
Prob(Q):	0.63	Prob(JB):	0.00
Heteroskedasticity (H):	36620.88	Skew:	2.56
Prob(H) (two-sided):	0.00	Kurtosis:	7.76

Fig. 3. SARIMAX Results

Plots with the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) are useful tools for analyzing and modeling time series data. These plots serve as helpful visual aids for understanding the relationship between a time series and its lagged variants, providing information about the underlying structure of the data.

Peaks in the ACF plot can point towards two potential phenomena:

a) Seasonal Patterns : There may be seasonal components in the data if the peaks show a recurrent pattern at regular intervals (e.g., every 12 lags for monthly data). These seasonal elements might be patterns that repeat themselves according to the season, holidays, or other cyclical elements.

b) Moving Average (MA) Components : The influence of previous errors (residuals) on present results can also be shown via ACF peaks. This case implies the existence of an MA component, in which forecast errors from the past affect future values.

Significant peaks, on the other hand, clearly show how previous values—apart from the impact of lags—have influenced the current value in the PACF plot. These peaks show that the data has Autoregressive (AR) components. According to AR models, values in the future can be predicted by combining past values in a linear fashion.Improved illustrations are produced to efficiently portray the predicted data.

Through the simultaneous analysis of the ACF and PACF plots, we are able to obtain a thorough comprehension of the autocorrelation structure present in the time series.

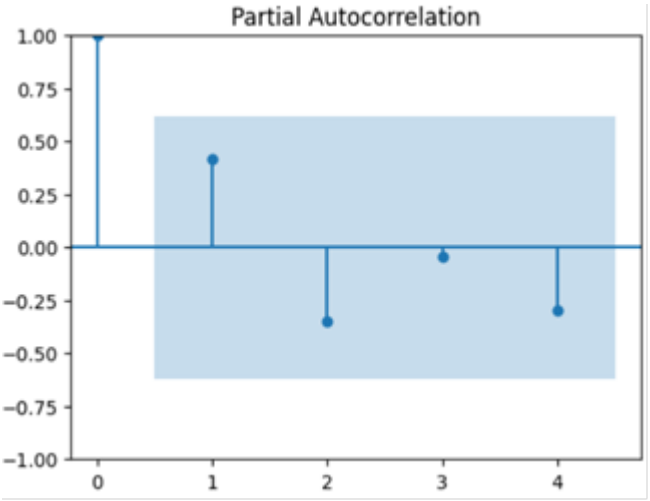


Fig. 4. Results based on testing of ARIMA models

When choosing the right models for time series forecasting, this combined knowledge proves to be extremely helpful. These recognized structures—AR, MA, and seasonality—are used by models like as ARIMA (Autoregressive Integrated Moving Average) and SARIMA (Seasonal ARIMA) to create predictive models that are highly successful.

In essence, ACF and PACF plots serve as a bridge between the raw data and the selection of optimal forecasting models. By interpreting the peaks and patterns within these plots, we can decipher the inherent relationships within the data, ultimately leading to more accurate and informed time series forecasting.

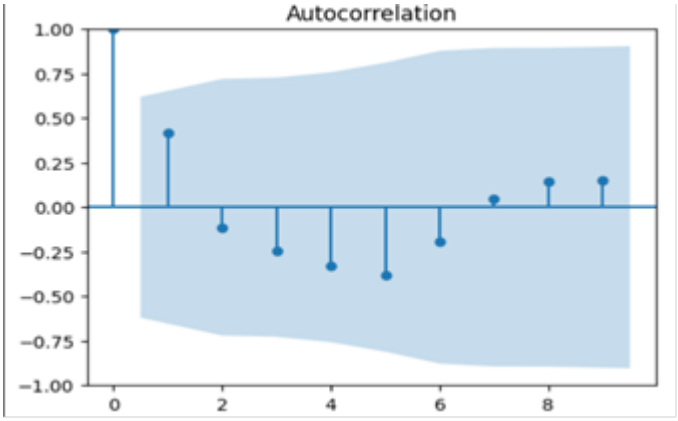


Fig. 5. Results based on testing of ARIMA models

VIII. Advantages of this Approach

Unlike conventional, static recommendations, our method is dynamic. We take advantage of the collective knowledge within the programming community, which is a wealth of information about language usage patterns, market demands, and trends. Through the use of content-based approaches and collaborative filtering, our system learns from the interactions and language preferences of experienced programmers. Our comprehensive analysis of user behavior enables us to suggest languages that are both highly valued in the ever changing technology landscape and in line with students' interests.

However, our light is not limited to this moment. By using time series analysis and ARIMA models to forecast future patterns in language popularity, we use the art of prophecy. This provides pupils with the ability to traverse the ever-changing terrain by giving them languages that have the potential to become future dominant as well as presently prominent.

Our platform has the potential to completely transform the way that programming languages are learned because to its diverse approach. In our ideal future, students go out on their adventures not fearfully but with the firm assurance that their route would be illuminated by statistics, trends, and the collective wisdom of the programming community.

This approach offers several advantages over existing methods:

A. *Data-driven Recommendations*

By basing our recommendations on the objective realities of language usage and industry trends, we transcend subjective opinions.

B. *Future-Oriented Insights*

We enable students to make well-informed decisions that are in line with the rapidly changing technology world by forecasting future trends.

C. *User Friendly Interface*

By predicting future trends, we empower students to make informed decisions that align with the quickly evolving digital landscape.

This innovative method may help prospective programmers find languages that will satisfy their current interests while also giving them the tools they need to succeed in the rapidly changing field of technology.

IX. Conclusion

For aspirant learners, the ever-expanding field of programming languages poses a formidable challenge: making the appropriate decision in a wide and constantly changing terrain. This research has suggested a new platform that goes beyond simple guidance, acting as an interactive road map that shows programmers what to learn.

Unlike conventional, static recommendations, our method is dynamic. By utilizing data on language usage, industry trends, and developer interactions, we are able to harness the collective intelligence of the programming community. Using a blend of content-based methods and collaborative filtering, our approach suggests languages that are both highly valued in the workplace and correspond with a student's interests.

The foundation for a strong platform that may enable students to choose their programming language with knowledge is laid by this research. Subsequent tasks will comprise putting the suggested method into practice, testing it with users, and making adjustments in response to their input. We think this platform has a great deal of potential to change the way that programming is taught and help students pursue successful careers in the rapidly changing technology industry.

REFERENCES

[1] A review and analysis of technologies for developing web applications, BY-SOLOMON ANTONY, 2012.

[2] Web Application Development -a study on UML Web Application Extension, Andreas Oskarsson, Martin Kling, Tobias Norberg, 2002.

[3] Research on HTML5 in Web Development, Ch Rajesh, K S V Krishna Srikanth, 2014.

[4] Review of WordPress website vs coding website By-Swapnil S More, 2018.

[5] Website Development Technologies: A Review By - Pratiksha D Dutonde , Shivani S Mamidwar , Monali Sunil Korvate, 2022.

[6] Wide and Deep Learning for Recommender Systems Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, Hemal Shah, 2016.

[7] Research Paper Recommendation System Using Natural Language Processing Submitted in partial fulfilment of the requirements for the degree of Bachelor of Technology in Computer Science and Engineering by Siddharth Sanjay Gandhi 19BCE0005 Under the guidance of Dr. Akila Victor School of Computer Science and Engineering VIT Vellore Dr. Felix Dietrich, Department of Informatics Technical University of Munich, 2023.

[8] RESEARCH ON COVID-19 EPIDEMIC BASED ON ARIMA MODEL Li Zhihao et al 2021 J. Phys.: Conf. Ser. 2012 012063, 2012.

[9] Research Paper On Recommendation System Dr. Alka Singhal Assit Professor,CSE department, IPEC Ghaziabad, India Shivangi Rastogi Student, CSE department, IPEC Ghaziabad, India Nikhil Panchal Student, CSE department, IPEC Ghaziabad, India Shivani Chauhan Student, CSE department IPEC Ghaziabad, India Shradha Varshney Student, CSE department, IPEC Ghaziabad, India, 2021.

[10] Forecasting with Machine Learning Dr. Osman Mohamed Abbas, 2018.

[11] An Overview of Data Analysis and Interpretations in Research. Dibekulu, Dawit. (2020).

[12] HTML5 -Unleashing the Power of the Web Lubbers, Peter Albers, Brian Salim, Frank Kumar, Mahesh. (2023).

[13]] Use of React.js in Modern Web Development and its Comparative Analysis Kunal Goyal, Dr. Vishal Shrivastava , Dr. Akhil Pandey, 2024

[14]] A Comparative Analysis of Database Management Systems for Time Series Data TOVE VERNER-CARLSSON VALERIO LOMANTO, 2023

[15] An Analysis of Time Series Analysis and Forecasting Techniques. Malik, Pankaj Dangi, Aditya Singh, Aditya Asst, Thakur Pratap, Aditya Parihar, Singh Sharma, Utkarsh Mishra, Lakshya. (2023)

[16] IMPLEMENTATION AND COMPARISON OF MERN STACK TECHNOLOGY WITH HTML/CSS, SQL, PHP MEAN IN WEB DEVELOPMENT Er. Vikas Goyal, Ashish Kumar Mishra, Daljeet Singh, 2023

[17] React-JS: A Cutting-Edge Framework for Web Designing Navratan Mal, Dr. Vishal Shrivastava, Dr. Akhil Pandey. 2024

[18]] Predicting Stock Prices Using Hybrid LSTM andARIMA Model Chi Ma, Jie Wu* , Hui Hu, YueNai Chen, JingYan Li, 2023