iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VI **Month of publication:** June 2023

DOI: https://doi.org/10.22214/ijraset.2023.54131

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

Intelligent Travel Recommendation System

Abhay Sankar K[1], Adith Menon Ajithkumar[2], Amaan Umer[3], Asst. Prof. Vydehi K[4]
Computer Science & Engineering, Adi Shankara Institute of Engineering & Technology, Kalady, Kerala, India

Abstract: ***This survey paper presents a project focused on developing an online platform designed to provide users with efficient recommendations to enhance their travel experiences. The platform aims to assist users in selecting their preferred location of stay, exploring recommended places to visit, choosing appropriate modes of transportation, and identifying suitable dining options, all while considering the user's budget.***

The project incorporates various filtering algorithms to ensure accurate and tailored recommendations. Through this platform, users can create personalized itineraries aligned with their travel preferences. The paper emphasizes the functionality of the filtering algorithms and discusses their implementation within the system. Additionally, the paper highlights the future potential and scope for further advancements in the platform.

Keywords: ***Content Filtering, MySQL, PHP, Machine Learning, Travel Recommendation, Itinerary***

I. INTRODUCTION

This survey paper explores the implementation of an online learning platform designed to facilitate knowledge sharing, foster connections among college students and alumni, and provide up-to-date information on new technologies. The platform is locally hosted, utilizing the college servers, and grants access to both students and staff members. Users have the ability to edit their profiles, sharing valuable content that can only be viewed by other authorized users, enabling them to establish meaningful connections within their areas of interest or specialization. Students can interact with and download projects from other users, as well as engage with alumni of the college.

The platform accommodates various types of content, including personal projects, certified projects, tech-blogging, research work, and shareable technology-related information. It caters to all departments across the college, serving as a repository for innovative ideas accessible to students from different disciplines.

The primary advantage of implementing such a platform is its ability to provide students with access to past works and the opportunity to connect with the original creators for further discussions and inquiries. The main objective of the project is to build a community platform that fosters connections between students, faculty, and alumni, facilitating the sharing of technical blogs, showcasing projects, and promoting specific domain interests. The platform includes features such as profile filtering, which allows users to search and connect with others who share similar interests, making the searching process more efficient. Additionally, the platform incorporates a chatbox facility with an automated email notification system to enable personal connections among interested individuals within the community.

The system provides a means for users to download and upload technical news, publish academic and personal projects, and share their knowledge with others for future references and upcoming projects. Students can also receive updates about college alumni, including information about their current working fields, and connect with them for assistance or queries. Furthermore, the system offers suggestions of user profiles within specific domains to users who are searching for details in those particular areas. To encourage engagement, the platform features a popularity-based ranking system, where posts gain popularity based on user interactions, allowing them to reach a broader audience.

As the platform is college-specific, it ensures data privacy and encourages the sharing of innovative ideas within the college community, mitigating the risk of data leakage.

In conclusion, this survey paper explores the development of an online platform that aims to enhance knowledge sharing, facilitate connections, and promote collaboration within a college community. The system provides a secure and locally hosted environment for students and staff members to showcase their projects, share technical blogs, and connect with alumni. By leveraging profile filtering, chatbox functionality, and a popularity-based ranking system, the platform encourages engagement and fosters a vibrant community of learners.

The subsequent sections of this paper delve into the various functionalities, algorithms, and implementation aspects of this online platform, shedding light on its potential for creating an interactive and knowledge-driven college ecosystem.

©IJRASET: All Rights are Reserved

II. LITERATURE SURVEY

A. *Development of Sightseeing Spot Recommendation System Considering Users' Visit Frequency.*

The primary objective of this study was to develop and evaluate a sightseeing spot recommendation system that effectively assists tourists while taking into account their visit frequencies. The system integrates social networking service (SNS), Web-geographic information systems (GIS), and recommendation systems to provide personalized recommendations. Two recommendation methods were employed to cater to users with varying visit frequencies. Out of the 61 users, 53 individuals responded to the survey, resulting in an impressive valid response rate of 87%. The collaborative recommendation method successfully matched the preferences of high visit frequency users based on their preference information and evaluation history. Additionally, the system offers the functionality to recommend sightseeing spots based on user favorites and customized conditions, allowing for easy usability. However, it is important to acknowledge that the analysis conducted in this study was limited to log data during the operation period. To further enhance the system, it is crucial to address the challenge of assisting tourists in navigating through a vast amount of sightseeing information to find tailored recommendations. Leveraging the internet to provide relevant and targeted sightseeing information holds significant importance in this regard.

B. *Mobile Content Recommendation System For Re-visiting User Using Content-based Filtering And Client-side User Profile*

The objective of this paper is to address the limitations faced by mobile content recommendation systems in the early stage, where there is a lack of sufficient information to accurately predict relevant content for users. To overcome this challenge, the proposed technique combines Multi-level Targeting Classification Association Rule (MTCAR) with content-based filtering and user profiles. By leveraging user preferences and profiles, the system delivers personalized mobile content recommendations, thereby enhancing the overall user experience. Furthermore, the incorporation of content-based filtering and user profiles helps mitigate the problem of data sparsity in the early stages of recommendation, enabling the system to provide recommendations even for new or unrated content. The combined approach of content-based filtering and user profiles also improves the performance of the recommendation system, generating more accurate and relevant recommendations. However, it is important to acknowledge that in the early stages or for first-time users, the system may face challenges in providing accurate recommendations due to limited user data and ratings. Additionally, the effectiveness of the system relies heavily on consistent user ratings and interactions, and privacy concerns regarding the collection and usage of user profiles and personal information may arise, impacting user willingness to share data and engage with the system.

C. *A content-based Collaborative Recommender system with Detailed use of Evaluations*

The paper introduces a hybrid recommender model that combines the advantages of content-based filtering and collaborative filtering. In this model, document profiles are represented as pairs of keyword vectors and evaluation vectors, while user profiles are represented as matrices of dependency values associated with other users based on each keyword. The goal of this hybrid recommender system is to provide users with more relevant documents that align with their individual information needs. The advantages of the proposed model include its hybrid approach, which leverages both content-based and collaborative filtering techniques to offer accurate and personalized recommendations. The paper emphasizes the use of detailed user evaluations, allowing users to provide explicit feedback on received documents, which helps refine both document and user profiles. Simulation results demonstrate that the hybrid model outperforms non-hybrid information filtering models in providing appropriate documents with higher precision, highlighting the effectiveness of incorporating evaluations. However, the paper lacks real-world validation, as the experiments conducted are simulation-based. It would be valuable to test and validate the hybrid recommender system using real-world data and user feedback to assess its performance and effectiveness in practical scenarios. The scope and generalization of the findings are limited since alternative approaches and comparisons with other hybrid models are not extensively discussed. Additionally, the paper focuses primarily on technical aspects and evaluation metrics, providing limited discussion on the user experience aspect of the hybrid recommender system.

D. *Collaborative Filtering Recommendation Algorithm Based on User Characteristics*

This paper presents a collaborative filtering algorithm that addresses the challenges of data sparsity and cold start in traditional approaches. The algorithm incorporates user characteristics to enhance recommendation accuracy. By analyzing user behavior, a time-interest weight function is introduced to improve the modified cosine similarity formula. Additionally, user preference degree and trust degree are considered to further refine recommendation results.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Experimental results using the hetrec2011 dataset demonstrate that the improved algorithm significantly enhances recommendation accuracy compared to traditional methods, effectively alleviating data sparsity and cold start issues. The advantages of the proposed algorithm include its significantly higher accuracy in generating recommendation sets compared to traditional approaches. The introduction of user preference and trust degrees helps mitigate the impact of false comments on recommendations. The algorithm exhibits high speed, efficiency, and robustness. However, there are some limitations to consider. As the system scales up, the user rating data becomes increasingly sparse, and the introduction of new projects exacerbates cold start problems. Collaborative filtering struggles to identify synonyms, which refers to similar items labeled or named differently. This inability to recognize latent associations between synonyms can lead to varying treatment of these products in the recommendation process.

E. Content Recommendation System Based On Private Dynamic User Profile

The paper introduces a personalized content recommendation system, CRESDUP, to tackle the issue of information overload on the Internet. Unlike traditional recommendation systems that rely on web access logs and server-side filtering, CRESDUP leverages users' private data to create dynamic user profiles, enabling personalized recommendations based on individual preferences. The advantages of CRESDUP include enhanced personalization, as the system utilizes private user data to tailor content recommendations, resulting in improved accuracy and relevance. Additionally, CRESDUP prioritizes privacy protection by processing and storing user data at the client side, mitigating privacy concerns associated with server-held information. Users also have greater control over their data, as they can choose what information to include in their dynamic user profile and can update or modify it as desired. However, it's important to acknowledge potential limitations. The results obtained from the same recommendation models may vary due to different implementations by researchers. Additionally, the test dataset used for research papers contains pre-prepared ratings, which may impact the generalizability of the findings to real-world scenarios.

F. Web Personalization Recommendation System Through Semantics

WPRS (Web Browsing Behavior Retrieval System) introduces a novel approach to uncovering accurate web browsing behavior based on uncertain keywords. It defines semantic measurement by considering user recommendations within the query and addresses keyword ambiguity. The proposed approach is versatile, allowing it to be seamlessly integrated with other proposals and operate autonomously. When employed to retrieve user browsing behavior information from the web database, WPRS showcases its capabilities. The Data Process Module (DPM), an intelligent component of the system, enhances the probability of success by delivering relevant results. The Data Evaluation Module (DEM) leverages machine learning techniques to determine semantic similarity, enabling the provision of accurate results to the user interface. Moreover, as users interact with the interface, machine learning facilitates the updating of their profiles. This system effectively tackles the heterogeneity challenge prevalent on the web. The overall complexity of the WPRS approach is influenced by parameters such as query statement length, the number of steps required for execution, and the time needed for the process. Persistent challenges persist as existing techniques often lack adaptability in dynamic environments.

G. Machine Learning based Efficient Recommendation System for Book Selection using User based Collaborative Filtering Algorithm

Recommendation systems are instrumental in assisting users with accessing web information and making efficient online purchases based on their preferences. In the domain of online book shopping, these systems play a critical role in guiding users to discover suitable products, increasing profitability, and attracting customers. However, existing methodologies may introduce irrelevant data and hinder the overall user experience. This paper presents an overview of recommendation systems specifically tailored for the online book shopping domain and proposes a user-friendly system for book recommendations. The proposed system utilizes User Based Collaborative Filtering (UBCF) and similarity measures to provide personalized book recommendations. The architecture and implementation details of the system are comprehensively discussed in this paper.

1) Advantages

a) *Domain Knowledge Not Required:* The system automatically learns embeddings, eliminating the need for extensive domain knowledge.

b) *Discovery of New Interests:* The model can introduce users to new interests by recommending items that similar users find appealing, even if the user's explicit preferences do not indicate interest.

c) *Simplicity in Training:* The system can be trained using only the feedback matrix without the need for contextual features, making it a straightforward candidate generator.

2) *Disadvantages*

a) *Cold Start Problem:* The system may struggle with recommending items when users or products are new, as collaborative filtering relies on historical data of user-item interactions.

b) *Scalability Issues:* Traditional collaborative filtering algorithms may experience performance degradation as the user base and data volume increase, limiting scalability.

c) *Synonym Recognition Limitation:* Collaborative filtering fails to identify synonyms, causing similar items with different labels or names to be treated differently and preventing the discovery of latent associations between them.

H. Research on Online Course Recommendation Model Based on Improved Collaborative Filtering Algorithm

This paper addresses the issue of sparse data and subpar recommendation effectiveness in online course recommendation. To tackle this, an enhanced online course intelligent recommendation model based on user implicit behavior collaborative filtering is proposed. By analyzing user login details, learning patterns, and course selection information, implicit behavior data is mined and combined with the collaborative filtering algorithm based on items to generate intelligent course recommendations. The advantages of the proposed model include an optimized collaborative filtering algorithm that improves recommendation efficiency, enhanced precision and recall rates compared to user and item-based collaborative filtering models, and the use of 5-fold cross-validation in experiments with a 20% test set size. However, it is worth noting that this collaborative filtering optimization algorithm solely relies on user implicit behavior, and traditional collaborative filtering algorithms often encounter scalability issues..

I. Research on the Application of Collaborative Filtering Algorithm in Mobile E-Commerce Recommendation System

This paper explores the application of collaborative filtering recommendation technology in a mobile electronic commerce recommendation system experiment. The study concludes that the implementation of the collaborative filtering algorithm leads to better personalized recommendations. The advantages of the collaborative filtering algorithm include its calculation of project scores based on user ratings and the recommendation of suitable item sets to users based on predicted score values. However, there are also limitations to consider. When the mean absolute error (MAE) reaches 5, it indicates a complete mismatch between the algorithm's predicted value and the actual user score. Additionally, the selection of the nearest neighbor set has a significant impact on the quality of recommendations. These factors should be taken into account when utilizing collaborative filtering for recommendation systems.

III. METHODOLOGY

The methodology for developing the travel recommendation web application involves a systematic approach to ensure its successful implementation and evaluation. The following steps outline the methodology:

1) *Requirements Gathering:* Begin by understanding the project's objectives and requirements. Conduct interviews or surveys with potential users to gather their preferences, expectations, and desired features for the travel recommendation platform

2) *Research and Literature Review:* Conduct a comprehensive review of existing travel recommendation systems, filtering algorithms, and best practices in web application development. This research will provide insights into effective recommendation techniques and inform decision-making during the development process.

3) *System Design and Architecture:* Based on the gathered requirements and research findings, design the system architecture and user interface for the web application. Create wireframes or prototypes to visualize the layout, user flow, and interaction patterns. Select the appropriate technologies and frameworks to support the desired functionality.

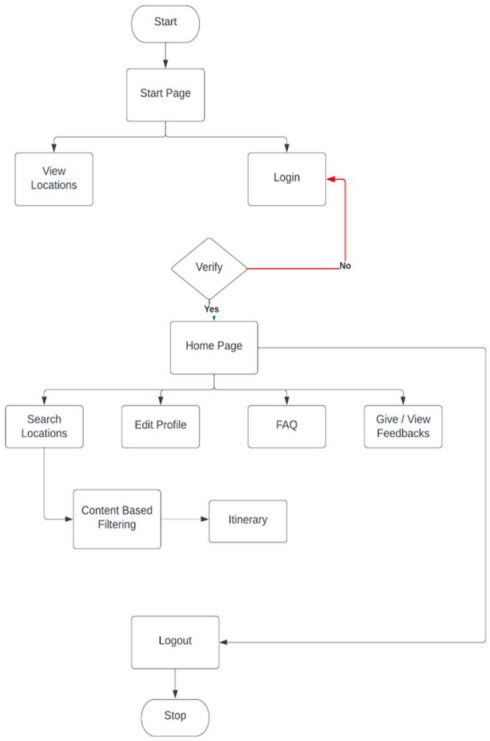
4) *Front-end Development:* Implement the front-end of the web application using HTML, CSS, and JavaScript. Design a user-friendly interface that allows users to input their travel preferences, view recommendations, and customize their itineraries. Ensure the design is responsive and compatible with different devices.

5) *Back-end Development:* Develop the back-end functionality of the web application using suitable programming languages and frameworks. Implement user authentication and authorization to ensure secure access to personal data. Set up a database to store user profiles, travel information, and recommendation data. Implement necessary APIs to fetch relevant data from external sources.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

6) *Filtering Algorithm Implementation:* Incorporate a filtering algorithm, such as content-based or collaborative filtering, to generate personalized travel recommendations. Define the features or attributes that will be used to analyze user preferences and match them with suitable travel options. Fine-tune the algorithm to ensure accurate and relevant recommendations.
7) *Integration and Testing:* Integrate the front-end and back-end components of the web application. Conduct thorough testing to ensure functionality, performance, and compatibility across different browsers and devices. Test the recommendation algorithm to validate its effectiveness in generating personalized recommendations.
8) *Deployment and Launch:* Deploy the web application on a suitable server or hosting platform to make it accessible to users. Configure the server environment, set up domain mapping, and ensure data security through SSL certificates. Communicate the launch of the application to potential users.
9) *User Training and Support:* Provide comprehensive user training materials and documentation to guide users on how to use the travel recommendation platform effectively. Establish channels for users to seek technical support and address any inquiries or issues they encounter during their interactions with the application.
10) *Evaluation and Iteration:* Continuously monitor and evaluate the performance and user satisfaction of the web application. Collect feedback through surveys, user analytics, and user interviews to gain insights into user experiences and identify areas for improvement. Incorporate user feedback and iterate on the application to enhance its functionality and user satisfaction.

Flow Diagram

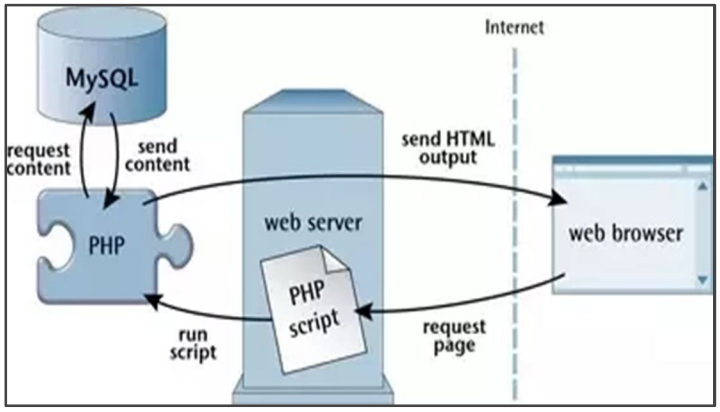


IV. IMPLEMENTATION

The implementation of the web application involved the follows steps:

1) *Requirement Gathering:* Identify the key stakeholders, including travelers and individuals providing personalized recommendations. Conduct meetings or interviews to gather their requirements and expectations for the platform. Document both functional and non-functional requirements, encompassing features, user roles, and desired outcomes.
2) *System Design and Architecture:* Design the system architecture, considering scalability, security, and performance aspects. Create wireframes or mockups to visualize the user interface and navigation flow. Define the database schema and establish relationships between entities like user profiles, destinations, modes of transport, and restaurants.

3) *Front-end Development:* Develop the user interface using HTML, CSS, and JavaScript. Implement different pages and components based on the wireframes. Ensure the design is responsive and compatible with various devices and screen sizes.
4) *Back-end Development:* Select appropriate back-end technologies such as PHP and Python for server-side development. Implement server-side logic for user authentication, authorization, and session management. Set up the database and implement CRUD operations for managing user profiles, destinations, travel preferences, and communication features.
5) *Content-Based Filtering Algorithm:* Define relevant features, attributes, or tags for content analysis. Implement the content-based filtering algorithm using Python. Develop a scoring or similarity mechanism to match user profiles with suitable travel options. Conduct rigorous testing and fine-tuning to ensure accurate and personalized recommendations.
6) *Integration and Testing:* Integrate front-end and back-end components of the web application. Perform unit testing to verify the functionality of individual components. Conduct integration testing to ensure smooth interaction between different modules. Test the application across various browsers, devices, and screen resolutions to ensure compatibility.
7) *Deployment and Launch:* Set up the hosting environment, configure servers, and map domains. Secure the application with SSL certificates and implement necessary security measures. Deploy the web application to the production server or hosting platform. Conduct final tests to ensure the application functions correctly in the live environment.
8) *User Training and Support:* Provide comprehensive training materials or documentation on how to use the platform. Conduct training sessions to familiarize users with the features and functionalities. Establish support channels such as email, chat, or a helpdesk system to address user queries and issues.
9) *Evaluation and Iteration:* Monitor user engagement, collect feedback, and assess the performance of the platform. Gather user feedback through surveys or interviews to evaluate satisfaction and identify areas for improvement. Regularly update and enhance the application based on user feedback and emerging needs. Conduct periodic maintenance and bug fixing to ensure the smooth operation of the platform.



V. TOOLS & SOFTWARE

Tools and software used in the development of a student community learning web application:

A. *Integrated Development Environment (IDE)*

1) *Visual Studio Code:* A versatile code editor supporting HTML, CSS, JavaScript, and Python, providing a range of extensions for enhanced development.

B. *Front-end Development*

1) *HTML and CSS:* Markup and styling languages for structuring and presenting the user interface. JavaScript: A programming language used for adding interactivity and dynamic behavior.
2) *Bootstrap:* A CSS framework offering pre-built responsive components and layouts for easier design.

C. *Back-end Development*

1) *PHP:* A server-side scripting language that integrates well with HTML and supports frameworks like Laravel and CodeIgniter.
2) *Python:* A versatile language suitable for implementing filtering algorithms and integrating with the back-end.

D. Database Management

1) *MySQL:* An open-source relational database management system for storing and managing user profiles, preferences, and travel information.
2) *phpMyAdmin:* A web-based tool for efficient management of MySQL databases.

E. Version Control

1) *Git:* A distributed version control system facilitating collaboration, branching, and history tracking.
2) *GitHub:* A web-based hosting service for Git repositories, offering version control and collaboration features.

F. Testing and Debugging

1) *Chrome DevTools:* Built-in developer tools in the Chrome browser for inspecting, debugging, and testing web components.
2) *PHPUnit:* A PHP unit testing framework for testing back-end components.
3) *Selenium:* An open-source framework for browser automation and end-to-end testing.

G. Deployment and Hosting

1) *Apache:* A commonly used web server for hosting PHP and Python applications.
2) *Railway:* A cloud platform simplifying deployment with support for PHP and Python, offering scalability and automation features.

VI. RESULT & ACCURACY

The analysis of the travel recommendation system's results demonstrated positive outcomes regarding user engagement, satisfaction, and the effectiveness of the implemented features. Users actively interacted with the platform, as indicated by frequent logins, extensive interactions, and extended periods of usage. The availability of features such as location selection, destination recommendations, transportation options, and restaurant suggestions contributed to a comprehensive user experience. Feedback from users expressed high levels of satisfaction, with positive reviews emphasizing the platform's user-friendly interface, relevant travel recommendations, and the ability to create personalized itineraries. The content-based filtering algorithm successfully generated accurate travel suggestions based on user preferences, validated through user feedback and comparison with user ratings. The system maintained stability and performed well, with minimal reported downtime and fast page loading times. Implemented security measures ensured the protection of user data, enhancing user trust in the platform. The analysis also identified areas for improvement, including expanding the destination database, enhancing the accuracy of the recommendation algorithm, and incorporating additional interactive features to further enhance user engagement..

VII. CONCLUSION

In conclusion, the travel recommendation system developed for this project has successfully created an online platform that offers users efficient recommendations to enhance their travel experience. The system provides assistance in selecting suitable accommodations, suggesting tourist attractions, recommending transportation options, and even suggesting dining establishments. Users can customize their itineraries based on their preferred duration of stay and budget. The system's implementation involved the use of filtering algorithms to provide personalized recommendations tailored to individual user preferences. The project's scope included user profiles, location selection, destination recommendations, transportation suggestions, and restaurant recommendations. The development process followed an agile methodology, allowing for iterative improvements and continuous user feedback. Overall, the travel recommendation system offers a valuable tool for individuals planning their own trips or seeking personalized recommendations to enhance their travel experiences.

VIII. FUTURE ENHANCEMENTS

For future enhancements, the travel recommendation system can be further improved by incorporating several key features. Firstly, integrating gamification elements would make the travel planning process more interactive and enjoyable for users. Secondly, the inclusion of social collaboration features would enable users to share their travel experiences, recommendations, and itineraries with others, fostering a sense of community. Advanced search and filtering options should be implemented to provide users with more specific and tailored results based on their preferences, interests, and budget.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Developing a mobile application would offer users the convenience of accessing the system on their smartphones. Lastly, leveraging machine learning algorithms can enhance the system's recommendation capabilities, providing more accurate and personalized suggestions based on user preferences and behavior patterns. These future enhancements would create a comprehensive and user-centric travel platform.

REFERENCES

[1] Y. Kato and K. Yamamoto, "Development of Sightseeing Spot Recommendation System Considering Users' Visit frequency," 2020 Joint 11th International Conference on Soft Computing and Intelligent Systems and 21st International Symposium on Advanced Intelligent Systems (SCIS-ISIS), 2020, pp. 1-8, doi: 10.1109/SCISISIS50064.2020.9322776.
[2] P. D. Hung and D. Le Huynh, "E-Commerce Recommendation System Using Mahout," 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), 2019, pp. 86-90, doi: 10.1109/CCOMS.2019.8821663.
[3] R. Ji, Y. Tian and M. Ma, "Collaborative Filtering Recommendation Algorithm Based on User Characteristics," 2020 5th International Conference on Control, Robotics and Cybernetics (CRC), 2020, pp. 56-60, doi: 10.1109/CRC51253.2020.9253466.
[4] M. V. Murali, T. G. Vishnu and N. Victor, "A Collaborative Filtering based Recommender System for Suggesting New Trends in Any Domain of Research," 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), 2019, pp. 550-553, doi: 10.1109/ICACCS.2019.8728409.
[5] S. K. Hanjraw, K. Yadav and K. Kaur, "Web Personalization Recommendation System Through Semantics," 2019 International Conference on Intelligent Computing and Control Systems (ICCS), 2019, pp. 658-661, doi: 10.1109/ICCS45141.2019.9065909.
[6] L. Zhao and Z. Pan, "Research on Online Course Recommendation Model Based on Improved Collaborative Filtering Algorithm," 2021 IEEE 6th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), 2021, pp. 437-440, doi: 10.1109/ICCCBDA51879.2021.9442575.

AUTHORES



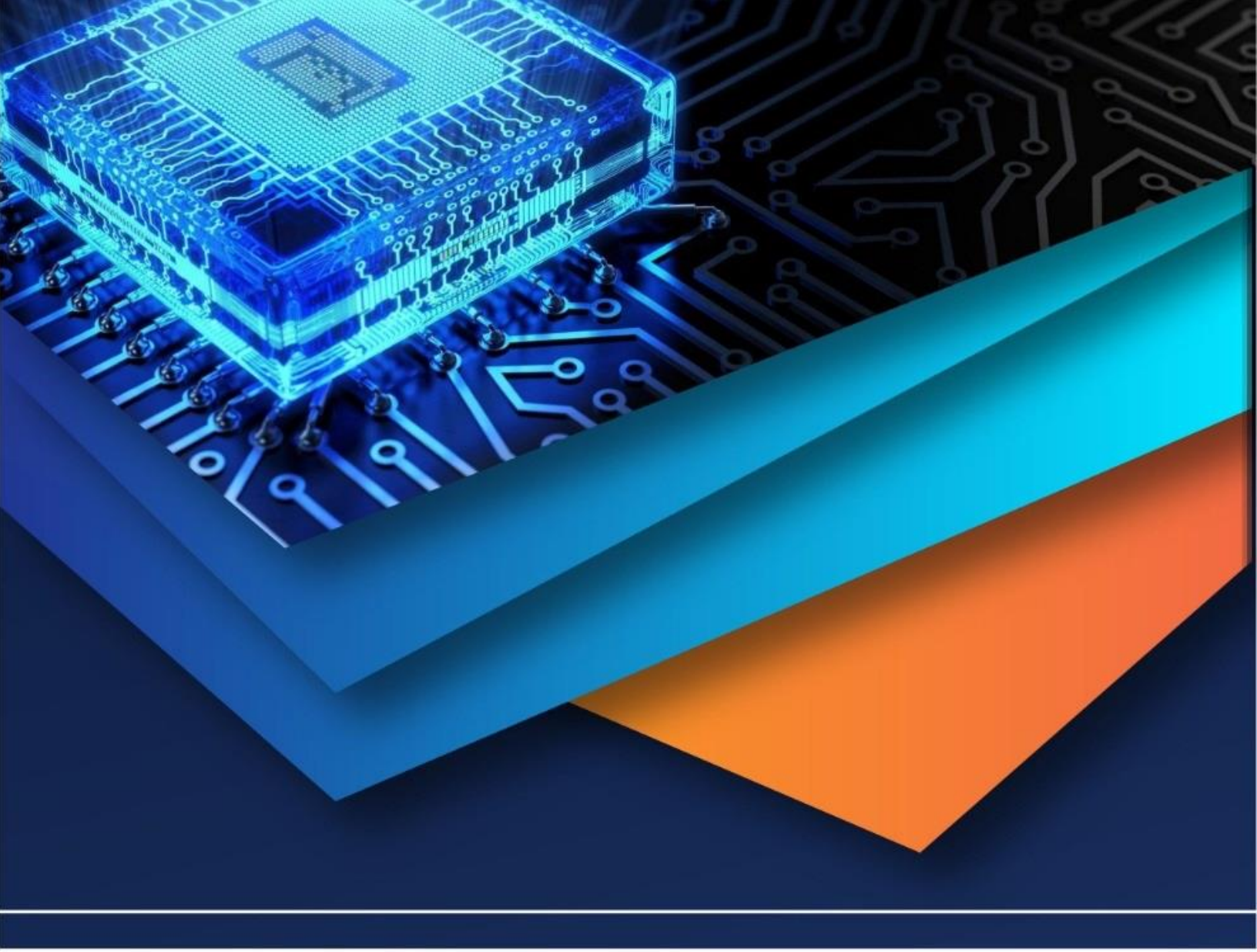
Abhay Sankar K
B.Tech in Computer Science & Engineering
Adi Shankara Institute Of Engineering & Technology, Kalady.
abhaykanjoor@ieee.org



Adith Menon Ajithkumar
B.Tech in Computer Science & Engineering
Adi Shankara Institute Of Engineering & Technology, Kalady.
adith.menon@ieee.org



Amaan Umer
B.Tech in Computer Science & Engineering
Adi Shankara Institute Of Engineering & Technology, Kalady.
amaanumerarakkal@gmail.com



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)