Night-Out

Project ID: TMP-23-379

Project Proposal Report

R.N.Akmeemana

BSc Special (Hons) - Information Technology

(Specialization in Information Technology)

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

September 2023

Night-Out

Project ID: TMP-23-379

Project Proposal Report

R.N.Akmeemana – IT20024086

Supervisor: Dr. Amitha Caldera

BSc Special (Hons) - Information Technology
(Specialization in Information Technology)

Department of Information Technology

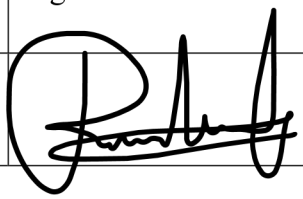
Sri Lanka Institute of Information Technology

Sri Lanka

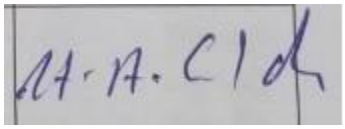
September 2023

DECLARATION OF THE CANDIDATE AND SUPERVISOR

We declare that this is our own work, and this project proposal does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Name	Student ID	Signature
R.N.Akmeemana	IT20024086	

The above candidates are carrying out research for the undergraduate dissertation under my supervision.



09/10/2023

…… ……

Dr. Amitha Caldera Date

ABSTRACT

In today's world, people are increasingly relying on mobile applications for their daily needs. With the growing number of mobile apps, it has become crucial for app developers to provide personalized experiences to their users. Machine learning has emerged as a powerful tool to help app developers achieve this goal. In this abstract, we will discuss a mobile application component that uses machine learning for user behavior analysis and provides personalized recommendations based on individual users.

The proposed mobile application component will use machine learning algorithms to analyze user behavior, including browsing history, search queries, and previous purchases. The machine learning models will analyze this data to understand user preferences, interests, and behavior patterns. Based on this analysis, the component will provide personalized recommendations to each user.

The recommendations will be tailored to each user's unique preferences and behavior patterns, and will include suggestions for products, services, and content that the user is likely to be interested in. The component will also use machine learning to learn from user feedback, ensuring that the recommendations are continually improving.

To implement this component, the mobile application will need to collect and store user data securely. The data will be used to train the machine learning models, and to provide personalized recommendations. The component will also need to provide users with the ability to control their data and privacy settings, including the option to delete their data or opt-out of data collection.

The success of this mobile application component will depend on the quality of the machine learning models used to analyze user behavior. The models will need to be accurate, efficient, and scalable to handle large amounts of user data. The component will also need to be designed with user privacy in mind, ensuring that user data is collected and stored securely, and that users have control over their data.

Overall, a mobile application component that uses machine learning for user behavior analysis and provides personalized recommendations has the potential to greatly improve user experiences. By providing tailored recommendations, the component can help users discover new products, services, and content that they are likely to enjoy, while also improving user engagement and loyalty. As mobile applications continue to grow in popularity, the use of machine learning to provide personalized experiences will become increasingly important for app developers looking to stay competitive in the market.

ACKNOWLEDGEMENT

I would like to extend my sincere gratitude to everyone who helped make this research a reality and finish this dissertation. First and foremost, I would like to express my sincere gratitude to Dr. Amitha Caldera, my academic advisor, for his never-ending backing, priceless advice, and unending patience throughout this research journey. Their knowledge and guidance were invaluable in forming this work.

I owe a debt of gratitude to the faculty and staff at the Sri Lanka Institute of Information Technology for creating a favorable academic setting and providing the tools I needed to carry out this research. I want to express my gratitude to my family for their support and patience as we went through the challenging parts of this project. My perseverance has been motivated by their faith in my abilities. My appreciation goes out to my friends and classmates for their support, conversations, and insights that have enriched this research.

This dissertation is the result of a team effort, and your assistance has been crucial. We appreciate your support, encouragement, and conviction that this work is important.

Table of Contents

LIST OF FIGURES

List of tables

LIST OF ABBREVIATIONS

API	Application Programming Interface
IT	Information Technology
WBC	Work Breakdown Chart

1. INTRODUCTION

In today's era of mobile technology, personalized experiences are crucial to the success of any mobile application. One effective way to achieve this is through the use of machine learning algorithms to analyze user behavior and provide personalized recommendations. This approach can greatly improve user engagement and loyalty, as well as increase app usage and revenue. In this abstract, we will discuss a mobile application component that utilizes machine learning to analyze user behavior and provide personalized recommendations. We will examine the key features and benefits of this component, as well as the challenges that must be overcome to implement it successfully.

1.1 Event Recommendation Applications

Event recommending applications have become increasingly popular in recent years, allowing users to discover new and exciting events in their area based on their interests and preferences. These applications leverage machine learning algorithms and user data to provide personalized recommendations, helping users to find events that they are likely to enjoy. Some popular event recommending applications on the market include Eventbrite, Meetup, and Time Out.

Eventbrite is a platform that allows users to discover and attend events ranging from music festivals to networking events. The platform utilizes machine learning algorithms to recommend events based on the user's previous event attendance and search history. It also offers features like personalized event suggestions, event tracking, and social sharing, allowing users to discover and attend events with friends and family.

Meetup is another popular event recommending application that focuses on connecting likeminded people through events. Users can join groups based on their interests and preferences, and attend events organized by the group members. Meetup utilizes machine learning to suggest new groups to users based on their interests and group participation history. It also offers features like event tracking and social sharing, helping users to discover and attend events with likeminded individuals.

Time Out is a global event recommending application that provides personalized recommendations based on the user's location, preferences, and behavior. The platform uses machine learning algorithms to suggest events, restaurants, and attractions that are tailored to the user's tastes. It also offers features like event tracking and social sharing, allowing users to discover and attend events with friends and family.

Despite the popularity of these events recommending applications, there is still a significant gap in the market that could be explored. Many of these applications focus on recommending events based on the user's interests and preferences, but do not take into account factors such as the user's mood, availability, and location. Additionally, many applications do not offer features like personalized notifications and reminders, which can help users to remember and attend events that they are interested in.

To address these gaps in the market, a new event recommending application could leverage machine learning algorithms to recommend events based on the user's mood, availability, and location. For example, the application could use sentiment analysis to determine the user's mood based on their social media activity and recommend events that are likely to improve their mood. It could also use location data to recommend events that are close to the user's current location, making it easier for them to attend events on short notice.

The application could also offer features like personalized notifications and reminders, reminding users of events that they are interested in and sending notifications when new events are added that match their interests. This could help to increase user engagement and attendance rates, as well as improve the overall user experience.

In conclusion, event recommending applications have become increasingly popular in recent years, but there is still a significant gap in the market that could be explored. By leveraging machine learning algorithms to recommend events based on the user's mood, availability, and location, and offering features like personalized notifications and reminders, a new event recommending application could provide a more personalized and engaging user experience.

1.1 Background survey

For the process of training the machine learning algorithms for the user behavior analysis component, I've created a survey for data collection. The questions included in the survey will help gather information about user's event preferences, behavior, attitude towards the sort of applications we aim to create. And it will greatly help the design and development process of the algorithm model training process and the application itself.

To identify the main problems and issues within the domain, and to get an overall idea about the domain such as to whom we provide this solution and how the problems diverse, we conducted a google form and 378 people have responded.

1. User Age

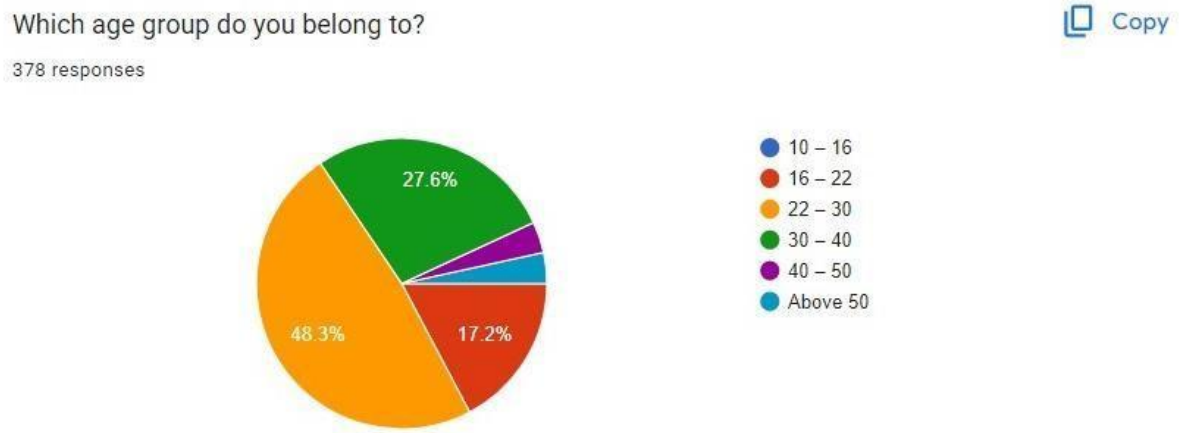


Figure 1 : Age groups of the users

Out of the sample of 378, 48.3% of the people have responded that they are between 22-30 years which means most of the participants were younger crowd. The second and the third age groups were to respond is 30-40 and 16-22 which are adjacent to th2 22-30 group. From the result, we can assume the users will be mainly 22-30 years of age.

2. User Gender

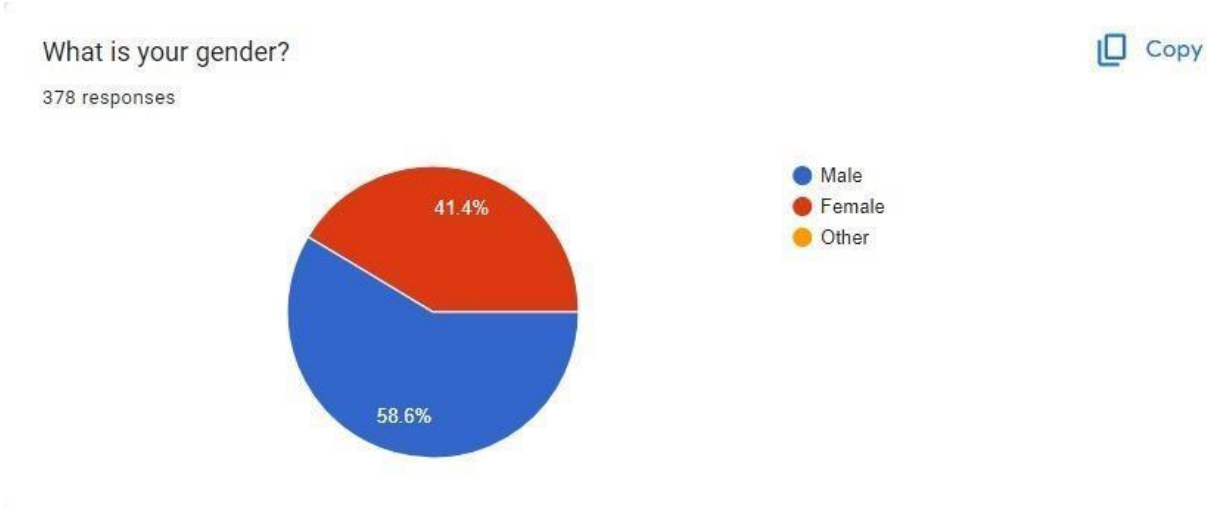


Figure 2 : User Gender

Out of the 378 responses received, 58.6% of the participants identify them as male and the rest is identified as female. This information is essential when considering the human computer interaction aspects of the app. App color themes and the user friendliness highly depends on the user gender and the age group.

3. User Type

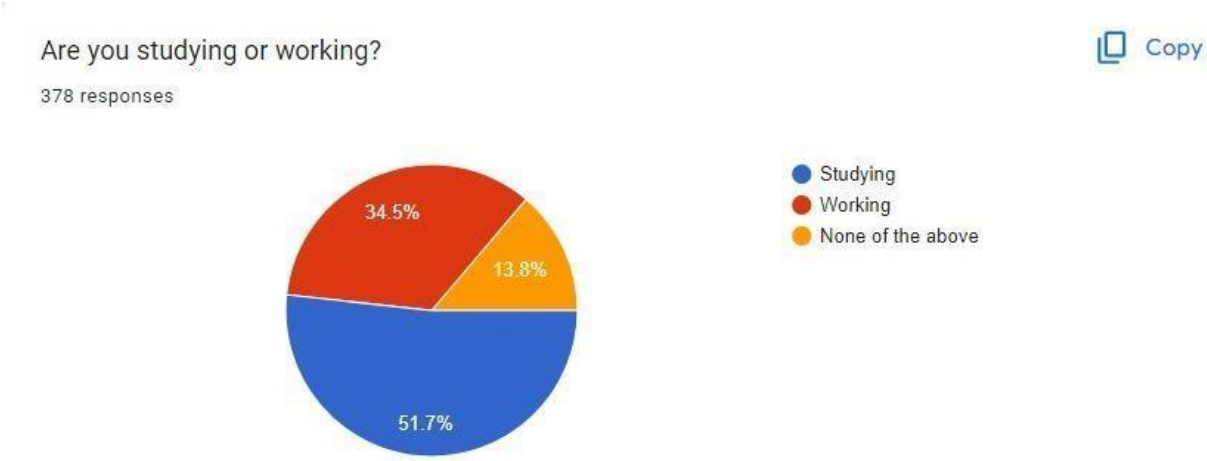


Figure 3 : User Type

According to the survey, 51.7% of people have responded that they are studying and 34.5% of them are working and 13.8% of them are not working nor studying respectively. This information is helpful when deciding what type of events to hold via the app and what kind of events that should be prioritized.

4. Usage of social media platforms to get notified about an event.

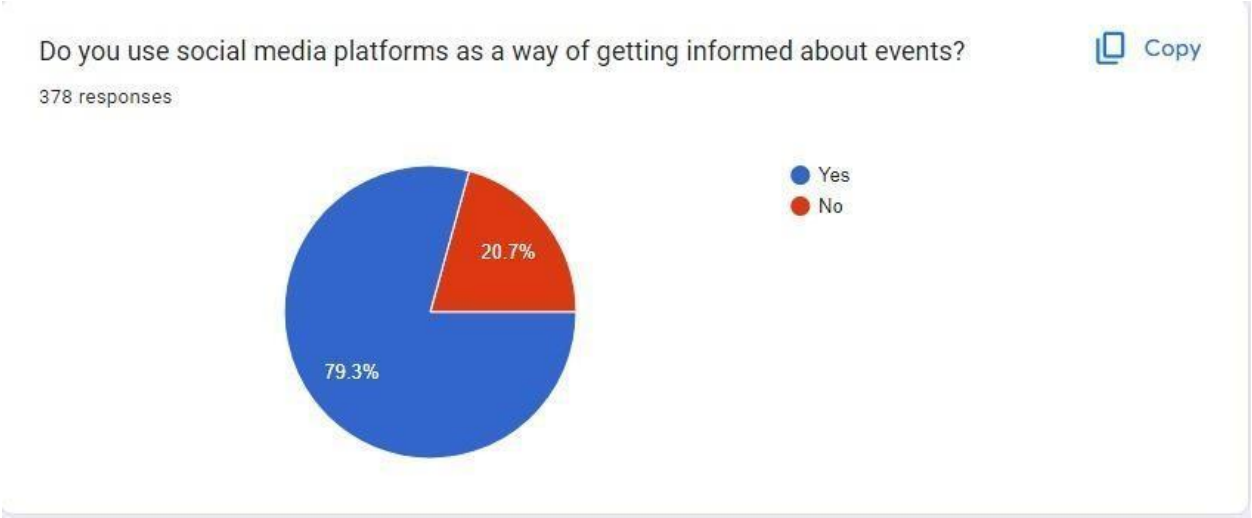


Figure 4 : Usage of social media platforms to get notified about an event

The majority of the participants, if not 79.3% of the participants responded positive to social media platforms as a way of getting informed about events. Only 20.7% of the participants are not using social media as a way of getting informed about events. To the people who currently use social media as a way of getting informed about events can have more improved benefits from this app while the others can get introduced to the app and start enjoying benefits of the app.

5. Likeliness to attend to an online hosted event.

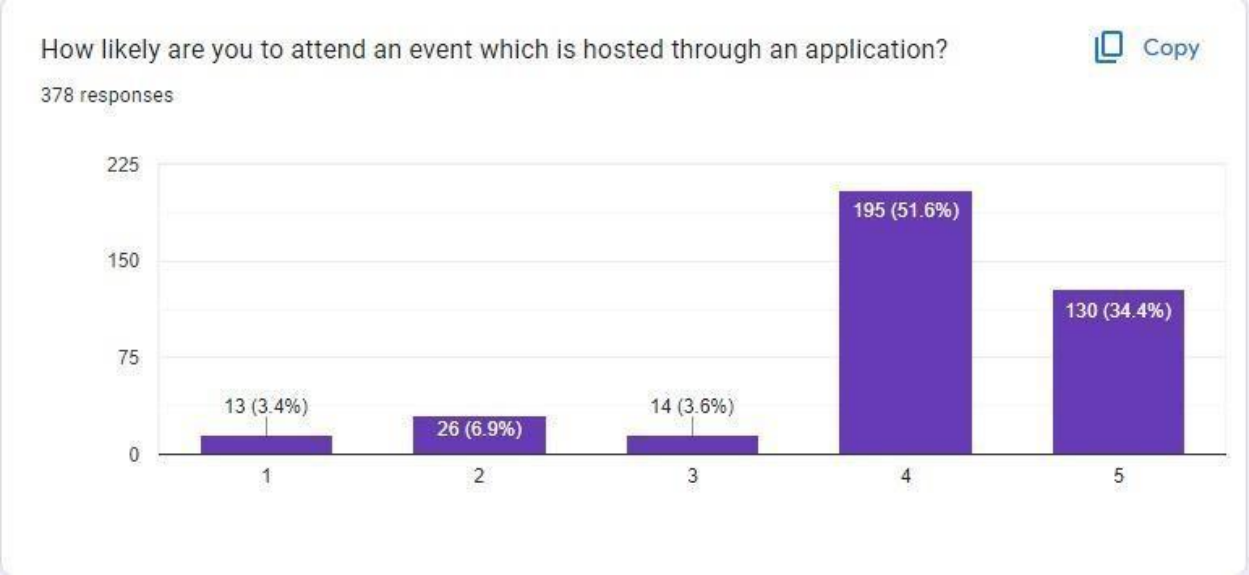


Figure 5 : Likeliness to attend to an online hosted event.

Even if the events are hosted through an application, it is not effective if the users are not attending the suggested events. Currently, 51.6% of participants rated 4 which means 80% likeliness in attending events hosted trough applications. Our goal is to get this numbers up and make most of the people participate events suggested by the application.

6. Use of the application

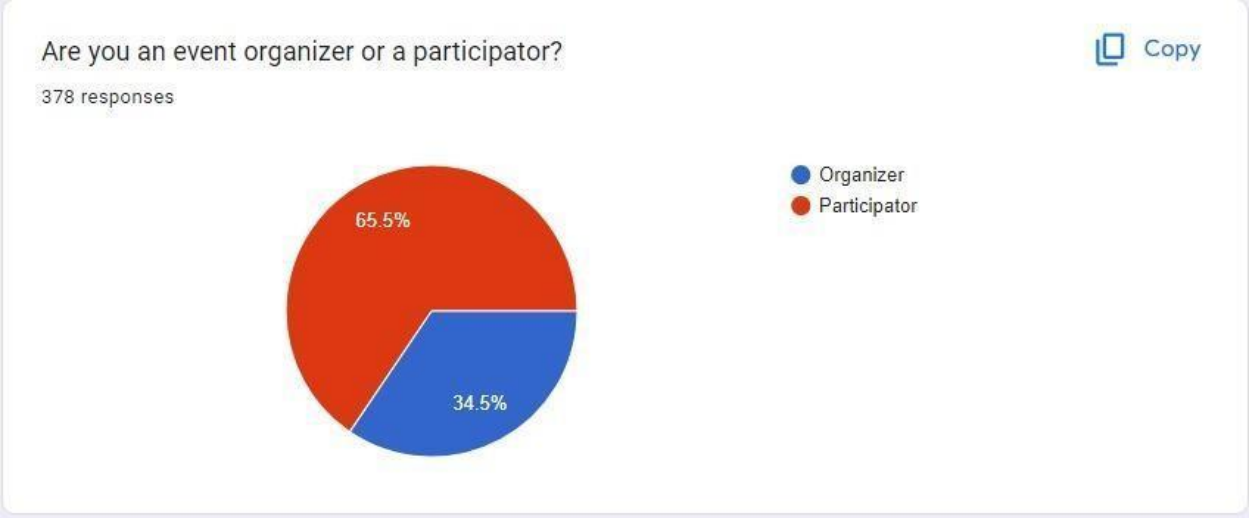


Figure 6 : Use of the application

There are two types of users to this type of application.

1. Organizers
2. Participants

Organizers are treated in a special way in order to optimize their businesses through the data and analysis provided by the application while participants can get suggestions according to their preferences. With this data, we can get a basic idea of the ratio of organizers to participants.

7. Preferred event type

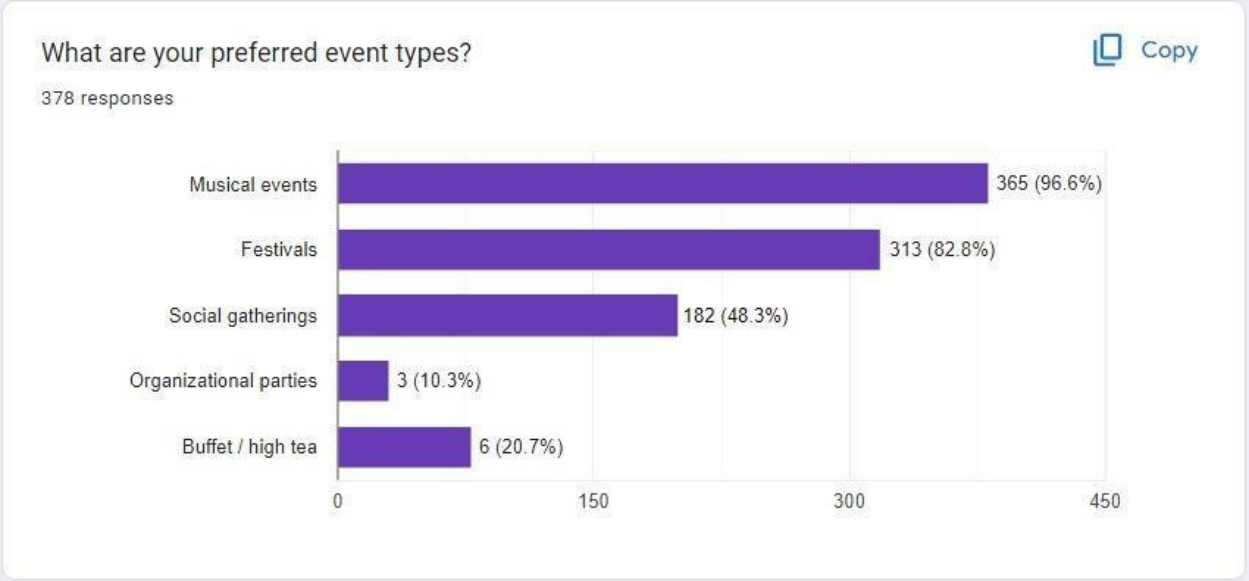


Figure 7 : What are your preferred event types.

According to the survey results, the most popular event type is musical events, which 96.6% would agree. However, the results can vary depending on the age, gender and the users' preferences. Apart from the musical events, festivals, social gatherings, buffet / high tea events and organizational parties are the next most popular events.

8. Expectations from a community

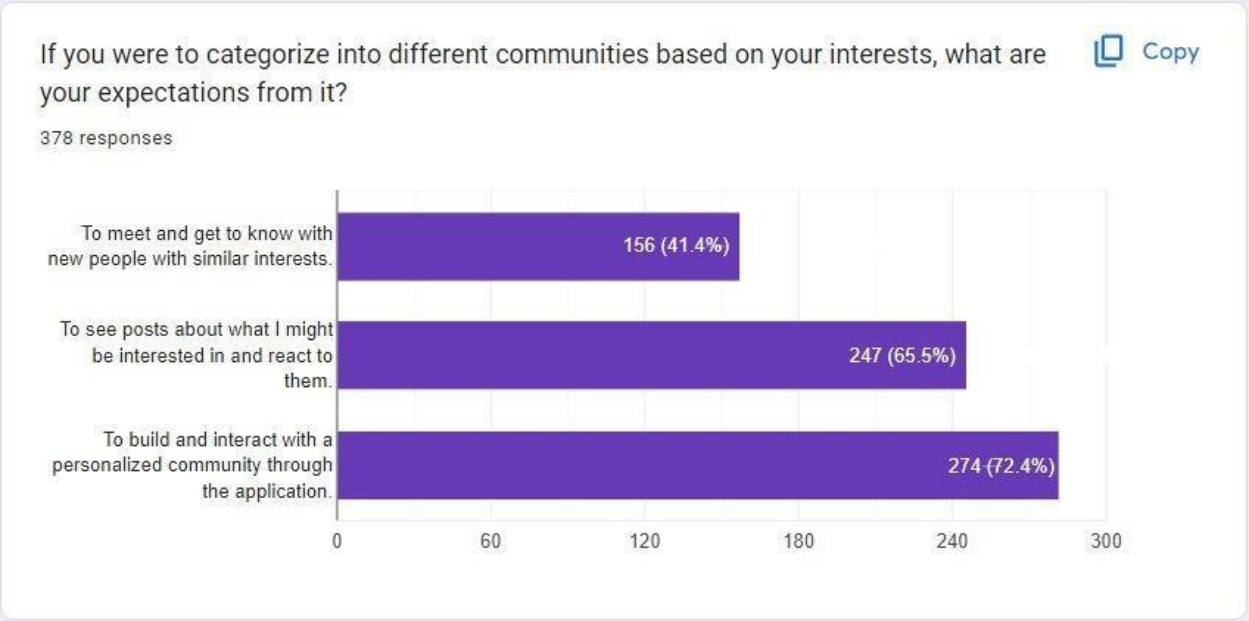


Figure 8 : Expectations from community

Another main thing to consider is what are the expectations of the application. 72.4% of the participants have responded that they want to build and interact with a personalized community through the application. 65.4% of the participants have responded that they want to see posts about what they might be interested in and react to them. 41.4% of the participants have responded that they want to meet and get to know new people with similar interests.

And a few more questions that will be prompted to the user through the application once its live

- How important is it for you to attend events that are close to your location?
- Would you be interested in receiving personalized event recommendations based on your interests, location, availability and such?
- How likely are you to attend an event if it's recommended to you by an event recommending application?
- How often do you use the internet and social media to discover new events?

1.2 Literature Review

Introduction

Event recommending applications have become increasingly popular in recent years, as they help users discover and attend events that are of interest to them. These applications use various approaches, such as collaborative filtering, content-based filtering, and hybrid approaches to recommend events. Recently, machine learning algorithms have emerged as a powerful tool for improving the accuracy and personalization of event recommending applications.

After a thoroughly conducted literature review,

I have found the below studies that have explored the same area,

1. Jannach, D., Ludewig, M., & Lack, T. (2019). Personalized event recommendation based on implicit feedback and event ontology. User Modeling and User-Adapted Interaction, 29(1), 107146. https://link.springer.com/article/10.1007/s11257-018-9211-9

2. Zhang, Y., Gao, H., Zhao, H., & Gao, Y. (2020). A personalized event recommendation system based on user interests and location. Multimedia Tools and Applications, 79(23), 1660316622. https://link.springer.com/article/10.1007/s11042-019-08061-5

3. Liu, J., He, X., & Li, L. (2016). A social collaborative filtering model for event recommendation in mobile social networks. Mobile Information Systems, 2016, Article ID 1263642. https://www.hindawi.com/journals/misy/2016/1263642/

1.3 Research Gap

The literature review identified several studies that investigated the use of machine learning algorithms in event recommendation applications. While these studies have shown promising results, there is still a research gap in understanding how to address challenges and limitations associated with recommender systems for mobile applications, such as data sparsity, privacy concerns, and user engagement. Therefore, further research is needed to develop more effective event recommending applications that can provide personalized recommendations while addressing these challenges. Additionally, there is a need for research on the impact of these applications on user behavior and how to improve user trust and adoption of these systems.

Figuring out that most applications available on the market only provide recommendations based on a few factors has been a key reason most of these applications proved to be unsuccessful.

1.4 Research problem

The focal point of this project is to examine the impact of feeling isolated and overwhelmed in a new environment. When individuals move to a new city, country, or unfamiliar surroundings, they often encounter challenges adapting to their new lifestyle. The lack of social connections, unfamiliar surroundings, and insufficient knowledge of the new environment can lead to feelings of isolation and loneliness, causing them to withdraw from social interaction. This can have a detrimental effect on an individual's mental health, resulting in stress, depression, and related issues.

In the current market, most applications do not offer businesses the ability to view customer communities and statistics that are customized to specific groups. This feature is necessary for addressing large audiences by grouping them accordingly. Moreover, these applications do not focus on enhancing real-life user interactions by assisting users in finding their respective communities and creating a platform for them to interact with each other, even when they are distantly located.

Furthermore, there are no social media platforms that employ advanced verified user techniques to authenticate users based on their interaction with communities, the reactions of other users to their posts, and the user's level of activity and influence within their community. By taking into account these crucial factors, our community-based recommendations are considerably different from those of our competitors.

The current entertainment finding applications available in the market fail to provide a personalized solution to this issue. Although these applications list all the entertainment

options available, the information is not customized to the user's preferences and interests, resulting in an overwhelming sense of difficulty in choosing what to do, which often leads to staying at home.

Recommender systems encounter a significant obstacle when the suggested items become too alike. However, the most crucial challenge that these systems face is the ever-changing nature of the data they use, which tends to be subjective and often struggles to handle new or unfamiliar recommendations. [13] Although many recommendation systems rely on user data or analyze their past behavior, this approach is not always effective, as user trends and interests are constantly evolving. Thus, relying solely on past behavior analysis for good recommendations becomes impractical, and the recommendation system needs to adapt to changes in the data environment. Based on our research, we have found that this presents a significant hurdle for recommender systems, as they struggle to react in a rapidly changing data environment.

Another aspect to consider in the business end is, where applications on the market that allow businesses to promote their services do not incorporate the personalized approach mentioned above. Therefore, there is no one-stop application available that addresses all these issues, hindering their widespread acceptance. This has been a significant reason why businesses fail, as they often invest in the wrong areas without adequate data or statistics to assist them in making informed decisions.

2 OBJECTIVES

2.1 Main Objectives

The primary objective is to create and develop the mobile application component which uses machine learning algorithms for user behavior analysis that provides the user with personalized recommendations based on the given criteria for each user.

2.2 Specific Objectives

1. Create user behavior analysis component:

The aim would be to design and create a component that collects user data (likes/dislikes, preferences, location, search history, event attendance history, social media activity and analyzes them to better understand the user.

2. Write machine learning algorithms.

The aim is to integrate machine learning algorithms with the above-mentioned component to process analyzed user behavior data to generate personalized recommendations. As well as train the dataset with the data gained through the previous process and surveys to increase recommendations and improve accuracy and relevance of the recommendations given to the user.

3. Use user-generated content to improve ML model.

The process will get user-generated content such as ratings, reviews, feedback to increase the accuracy of recommendation algorithms. The process will include created mechanisms for users to provide and access user generated.

4. Optimization

In order to assess the effectiveness and efficiency of the recommendation system, a complete evaluation and testing will be conducted as part of the project. In order to do this, user feedback would be gathered, and algorithms and recommendation methods would be iteratively improved depending on user feedback.

3 METHODOLOGIES

Night-Out is an event management application with 4 components,

1. User behavior analyzation system.
2. Community based recommendation system.
3. Socializing process and reviews system.
4. Profit maximization system.

For this proposal, we will be focusing on the user behavior analysis system.

Overall, the project's primary goal is to develop a mobile application component that makes use of machine learning algorithms to analyze user behavior, provide personalized recommendations, and provide a user- and privacy-friendly experience.

To achieve this, we will follow the below process.

1. Data Gathering and Preparation: Data on relevant events, user profiles, and user behavior will need to be gathered. Web scraping, API integrations, or collaborations with event platforms could all be used to accomplish this. After that, the gathered data would be cleaned, preprocessed, and put into a format that was appropriate for analysis and model training.

2. Data catalog: The content used to create recommendations must be stored and managed by the Content Catalog module. The module goes through a series of steps that include locating the data needed to make recommendations, retrieving the dataset, checking for integrity and traceability, and sorting the recommendations based on their best matches. The module that generates recommendations is then given the results.

3. Privacy Protection: Implementing measures to safeguard user data and guarantee compliance with data protection laws would address privacy concerns. To protect user privacy, strategies like data anonymization, secure data storage, and access control mechanisms would be used.

4. User Behavior Analysis: In order to understand user preferences, interests, and behavior patterns, this task will analyze user behavior data. To find patterns and better comprehend user preferences, statistical analysis techniques, clustering algorithms, or natural language processing techniques could be used.

5. Development and Training of Machine Learning Algorithms: A variety of machine learning algorithms, including collaborative filtering, content-based filtering, and hybrid approaches, would be investigated and put into practice for tailored event recommendation. The preprocessed data would be used to train the algorithms so they could recognize patterns and generate precise recommendations.

6. Evaluation and Validation: Using appropriate metrics, such as accuracy, precision, recall, and user satisfaction, the recommendation system would be assessed. To evaluate the efficiency and user-friendliness of the system, surveys, A/B testing, and user feedback could be used.

7. Iterative Refinement: The recommendation system would be improved iteratively based on the evaluation results and user feedback. This could entail optimizing the recommendation strategies, incorporating user-generated content, enhancing privacy protections, and fine-tuning the algorithms to improve the system's performance.

8. Final Analysis and Conclusion: The project would come to an end with a final analysis of the findings that would cover the conclusions, potential future directions, and limitations. A research report or thesis would contain the research methodology, results, and contributions.

For our project, we have decided to adopt the agile development methodology, which prioritizes flexibility, collaboration, and iterative progress. This approach is guided by the Agile Manifesto, which emphasizes key values such as prioritizing individuals and interactions, delivering working software, engaging in customer collaboration, and embracing change.

The agile methodology is characterized by short development cycles called sprints, where crossfunctional teams collaborate closely with customers or end-users to ensure that the product aligns with their needs. This allows for quick feedback loops and the ability to adapt to changing requirements. Continuous improvement is a central focus, with regular reviews and retrospectives to identify areas for enhancement.

Our project will follow the seven key phases of the agile model:

1. Planning: This phase involves defining the project scope, goals, and creating a roadmap or backlog of tasks. It establishes a clear vision and direction for the project.

2. Analysis: In this phase, a thorough analysis is conducted to understand the project requirements, user needs, and potential risks. This helps in shaping the development process.

3. Design: Based on the analysis, the team designs the software or product, including its features, functionalities, and user interfaces. This phase lays the foundation for the development work.

4. Implementation: This phase focuses on the actual development work, including coding, testing, and integration of different modules. It involves close collaboration between team members to ensure smooth progress.

5. Testing: Rigorous testing is conducted in this phase to ensure the quality and functionality of the software or product. Various testing techniques, such as unit testing, integration testing, and user acceptance testing, are employed.

6. Deployment: Once the testing phase is complete, the software or product is deployed to the production environment, making it available to end-users. This phase ensures a smooth transition to the operational stage.

7. Monitoring: The final phase involves actively monitoring the software or product in the production environment. This helps identify and address any issues or improvements needed, ensuring ongoing enhancement and customer satisfaction.

By adopting the agile methodology, we aim to foster effective collaboration, adaptability, and continuous improvement throughout the project lifecycle. It allows us to deliver a high-quality, user-centric product that meets the evolving needs of our customers.

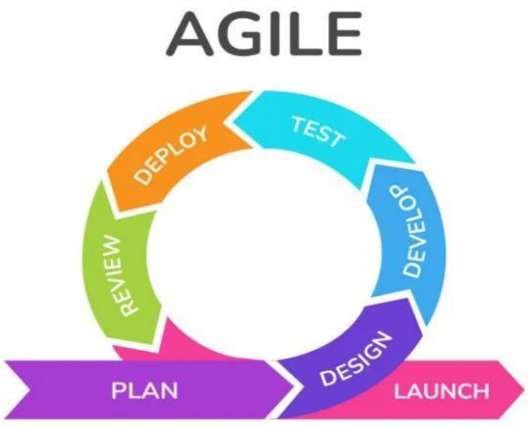


Figure 9 : Agile model

3.1 requirement gathering and feasibility studying

We gathered the requirements at two levels

01. Primary data gathering
02. Secondary data gathering

In primary data gathering, we mainly focused on user requirements. We are planning conducted a background survey through google forms to identify user requirements and the questions we are hoping to ask are mentioned bellow.

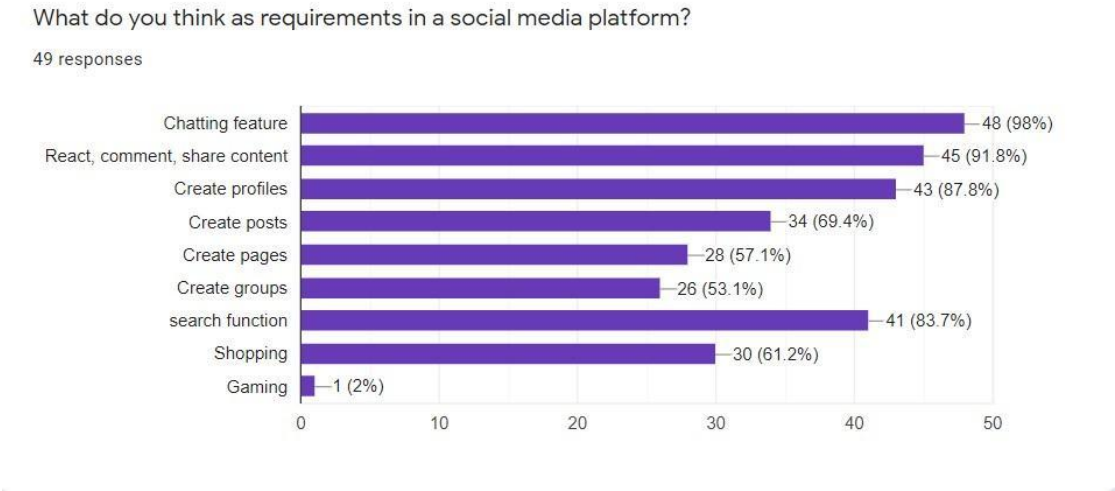


Figure 10 : Requirements in a social media platform

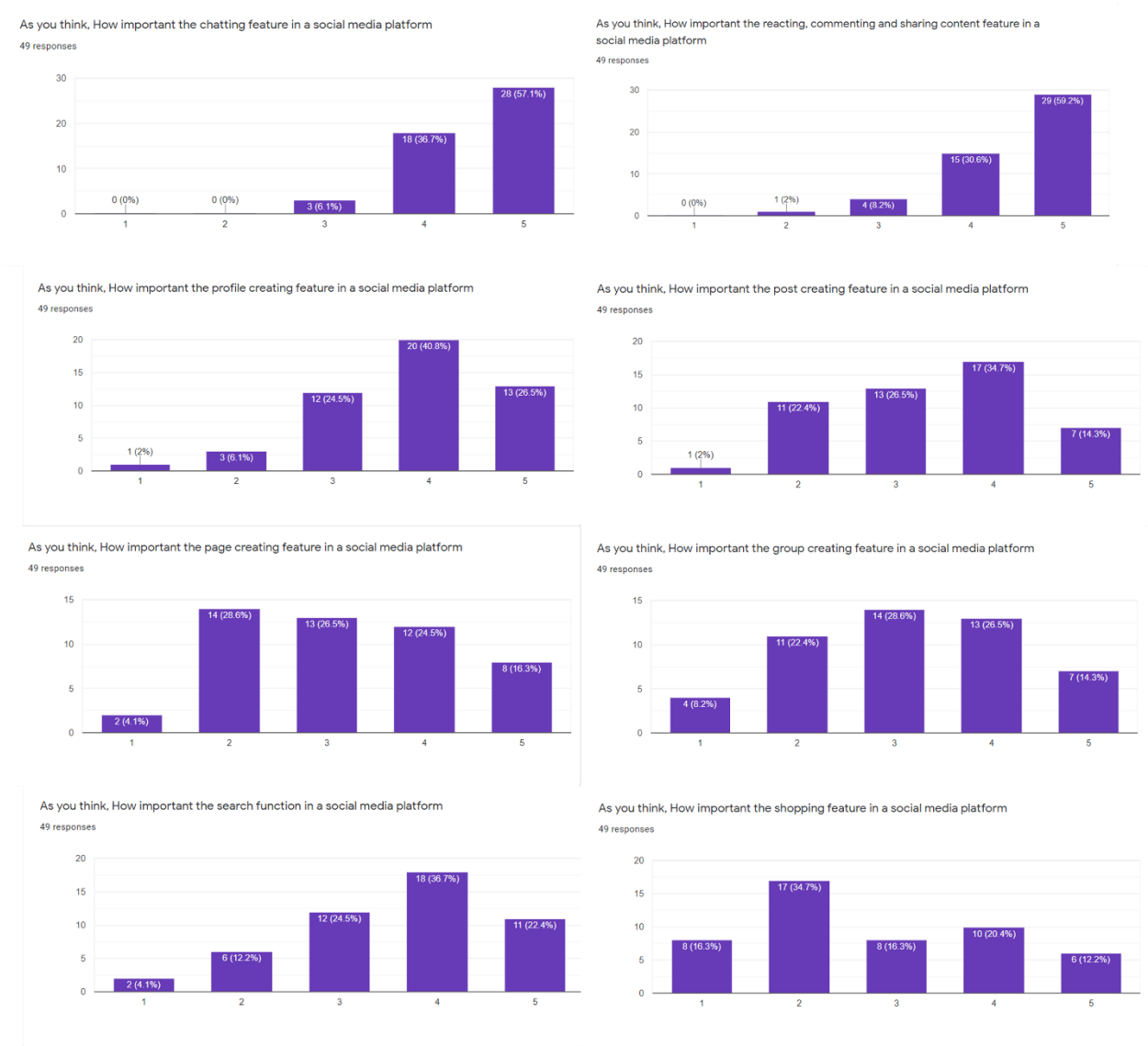


Figure 11 : Survey participants' ratings on each social media platform requirement

In addition to the questionnaire,

- we acquired data from event planners and consultants and gathered requirements.
- Contacted with an IT consultant and gathered information.

In secondary data gathering,

- We studied existing systems.
- We studied from various online resources such as online tutorials and web articles.

- We also gathered information from books and articles.

After performing requirement gathering, we performed a feasibility study,

1. Technical Feasibility

To successfully complete the research, all the team members should have the technical knowledge to proceed with the project. We made sure that we can acquire the required knowledge to complete the project addition to we already acquired knowledge.

2. Economic Feasibility

Financial resources are very important when we conduct the project. We made sure we have enough funds to complete project without having to stop half the way. We also made sure to plan handling unforeseen financial needs in the future.

3. Legal Feasibility

Not meeting legal feasibility is when a project runs afoul of legal restrictions such as zoning rules, data privacy laws, or social media laws. We made sure there are no conflicts with laws in our proposed system.

4. Operational Feasibility

This involves to what extent the project can be completed to meet the needs of the company. We had a discussion with Underground Music Coven members and made sure we are feasible in operational feasibility.

5. Scheduling Feasibility

Scheduling Feasibility means if a project can be completed and delivered in defined time. In our case, it is 1 year. We made sure the project is deliverable in the defined period.

Process of gathering secondary data for our research.

We undertook various activities. These included studying existing systems, referring to online resources such as tutorials and articles, as well as gathering information from books and articles. After completing the requirement gathering phase, we conducted a feasibility study to determine the viability of our proposed project. The feasibility study encompassed several key areas;

1. Starting with technical feasibility: We ensured that all team members possessed the necessary technical expertise to successfully complete the project, and also took steps to acquire additional knowledge as required.

2. Next, we assessed economic feasibility: recognizing the critical role of financial resources in the success of the project. We confirmed that we had sufficient funds to complete the project and also developed contingency plans to manage any unforeseen financial challenges.

3. In addition, we addressed legal feasibility: ensuring that our proposed system did not run afoul of any relevant laws or regulations, such as those pertaining to data privacy or social media.

4. We also considered operational feasibility: which refers to the ability of the project to meet the needs of the company. We are planning to work closely with the businesses such as hotels in order to acquire more information regarding this area.

5. Finally, we evaluated scheduling feasibility: which relates to the project's ability to be completed and delivered within the defined time frame. Given that our project timeline was set at one year, we ensured that our proposed project could be delivered within this timeframe.

3.2 Analyzing

By analyzing the gathered data, we categorized collected requirements as follows

3.2.1 Functional Requirements

1. User Registration and Authentication: The system should provide a user registration feature that allows users to create an account and authenticate their credentials to access personalized recommendations.

2. Content Management: The system should have functionality to manage the content catalog, including adding, editing, and deleting event listings. This includes capturing event details such as title, description, date, location, and category.

3. User Preference Capture: The system should provide a mechanism for users to input their preferences and interests, such as preferred event categories, locations, or specific keywords. This information will be used to personalize the recommendations.

4. Recommendation Generation: Based on user preferences and analysis of available content, the system should generate personalized event recommendations. The recommendations should be relevant, diverse, and presented in an intuitive manner to users.

5. User Interaction and Feedback: The system should allow users to interact with the recommended events by providing feedback, rating events, and leaving comments. This functionality promotes user engagement and helps improve future recommendations.

6. Search and Filtering: The system should include a search feature that enables users to find specific events based on different criteria, such as date, location, category, or keyword. Filtering options should also be provided to refine search results.
   Non-Functional Requirements:

3.2.2 Non-Functional requirements

1. Usability: The system should have an intuitive and user-friendly interface, making it easy for users to navigate, interact with the application, and understand the recommendations.

2. Performance: The system should be responsive and provide fast recommendation generation and retrieval of event information. It should handle a large number of users and content without significant performance degradation.

3. Security: The system should ensure the security and privacy of user data. This includes encryption of sensitive information, secure user authentication, and protection against unauthorized access or data breaches.

4. Scalability: The system should be designed to handle a growing number of users, events, and data without compromising performance. It should scale effectively to accommodate increasing demands.

5. Reliability: The system should be reliable and available to users, minimizing downtime and disruptions. It should have backup and recovery mechanisms to ensure data integrity and system continuity.

6. Compatibility: The system should be compatible with multiple platforms, such as mobile devices and web browsers, to reach a wider user base. It should be responsive and adaptable to different screen sizes and operating systems.

7. Accessibility: The system should be accessible to users with disabilities, adhering to accessibility standards and guidelines, such as providing alternative text for images and supporting screen readers.

3.2.3 User requirements

1. User-Friendly Interface: Users expect an intuitive and easy-to-use interface that allows them to navigate the application effortlessly. The interface should be visually appealing and provide clear instructions for accessing and interacting with the system.

2. Personalized Recommendations: Users require a personalized recommendation system that takes into account their preferences, interests, and previous interactions with the application. The recommendations should be relevant, diverse, and tailored to their specific needs.

3. Easy Registration and Authentication: Users want a seamless registration process that allows them to create an account quickly. They also expect a secure authentication mechanism, such as username/password or social media login, to protect their personal information.

4. Flexibility in Preferences: Users should have the flexibility to update and modify their preferences over time. The system should allow them to add or remove categories of interest, update their location preferences, and adjust other relevant settings.

5. Efficient Search and Filtering: Users expect a robust search functionality that enables them to find events based on specific criteria, such as date, location, category, or keyword. The system should provide advanced filtering options to narrow down search results effectively.

6. Event Details and Information: Users require comprehensive and accurate event information, including event descriptions, dates, times, locations, ticket availability, and organizer details. The system should provide all relevant information to help users make informed decisions.

7. User Interaction and Engagement: Users appreciate features that facilitate interaction and engagement with other users and event organizers. This includes the ability to leave comments, rate events, share recommendations, and participate in discussions or forums.

8. Notifications and Reminders: Users expect to receive timely notifications and reminders for upcoming events, changes in event details, or new recommendations based on their preferences. These notifications should be customizable to suit individual preferences.

9. Integration with Calendar and social media: Users find it convenient to have integration with their calendar applications to automatically add events and reminders. Additionally, integration with social media platforms allows them to share events with their networks and receive event recommendations from friends.

10. Offline Access and Offline Saving: Users appreciate the ability to access previously viewed events and recommendations even when they are offline. The system should allow users to save events for offline viewing and provide a seamless experience in areas with limited internet connectivity.

11. Feedback and Support: Users expect a feedback mechanism to provide suggestions, report issues, or seek assistance. The system should have a support system in place to address user queries and concerns promptly.

3.2.4 System requirements

Software requirements

- Operating System: Windows
- Web browser: Google Chrome
- Database management system: MySQL
- Programming languages: Python, JavaScript, and PHP
- Frameworks: React, Django, and Node.js
- Development environments: PyCharm, PhpStorm, NumPy, pandas
- Version control system: Git
- Application programmable interfaces: google maps.

Hardware Requirements

- Processor: Intel Core i5 or higher
- Memory (RAM): 8 GB or more
- Storage: 256 GB Solid State Drive (SSD)
- Display: 15-inch Full HD (1920 x 1080)
- Graphics card: NVIDIA GeForce GTX 1650 or equivalent
- Internet connectivity: Wi-Fi 5 or Ethernet connection

3.3 Design

To proceed with the design phase, we have developed a system architecture diagram to consolidate all necessary components. The design phase will commence by wireframing each

interface of the web application using Figma software. Upon completion of the wireframes, usability tests will be conducted using Hi-fidelity prototypes, with a focus on identifying and resolving issues from the user's perspective. This process will be efficient and effective, as it will save time and effort in advance of the implementation phase by reducing the risk of failing user acceptance testing. Subsequently, we will design the system's structure, starting with attribute identification and database design, before moving on to designing the hardware and software solutions.

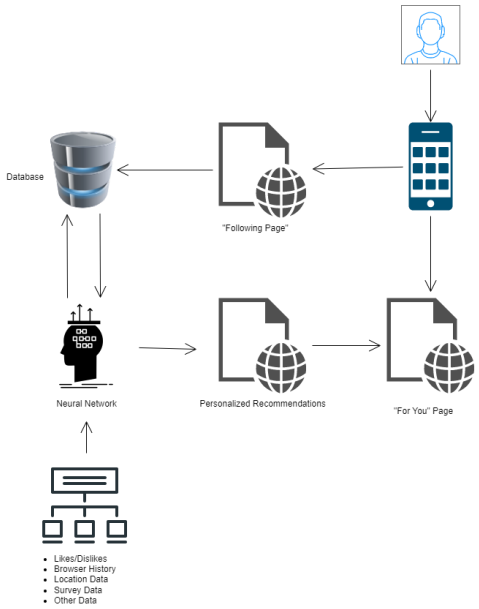


Figure 12 : High level system architecture diagram for proposed component.

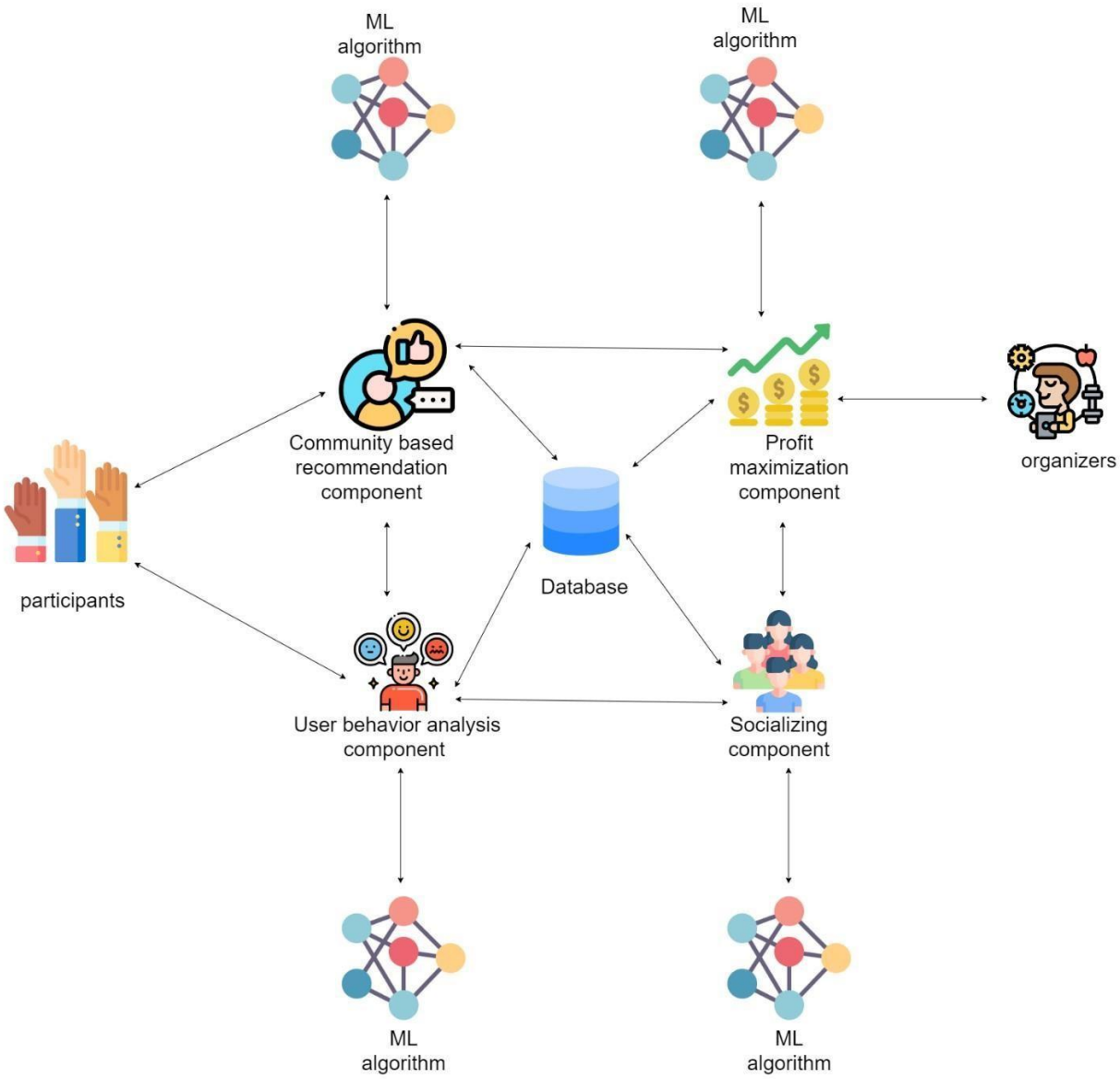


Figure 13 : High level system architecture diagram for the whole system.

3.4 Implementation

1. Requirement Analysis: Conduct a thorough analysis of the functional and non-functional requirements identified for the application. This involves understanding user needs, system capabilities, and technical constraints.

2. System Design: Based on the requirements analysis, design the system architecture and database structure. Determine the modules, components, and technologies that will be used in the development process. Create wireframes or mockups to visualize the user interface.

3. Database Design: Design and create a database schema that will store the event data, user information, and user preferences. Define the relationships between different entities and establish appropriate data storage and retrieval mechanisms.

4. Front-End Development: Implement the user interface (UI) design using appropriate front-end technologies such as HTML, CSS, and JavaScript. Develop responsive UI components and screens that allow users to interact with the application smoothly.

5. Back-End Development: Develop the server-side logic and functionality that handles data processing, recommendation generation, and user authentication. Use suitable programming languages and frameworks such as Python, Java, or Node.js to build the server-side components.

6. Integration and API Development: Integrate the application with external APIs or services for features such as event data retrieval, social media integration, or locationbased services. Develop API endpoints to allow communication between the front-end and back-end components.

7. Recommendation Algorithm Implementation: Implement the recommendation algorithm based on user preferences and content similarity calculations. This involves coding the logic for analyzing user data, filtering content, and generating personalized recommendations.

8. Testing and Quality Assurance: Conduct rigorous testing to ensure the application functions are as expected. Perform unit testing, integration testing, and user acceptance

testing to identify and fix any bugs or issues. Validate that the recommendations are accurate and relevant to user preferences.

9. Deployment and Release: Prepare the application for deployment by configuring servers, setting up hosting environments, and ensuring proper scalability and security measures. Deploy the application to a production environment where it is accessible to users.

10. User Training and Documentation: Create user documentation and provide training materials to guide users on how to use the application effectively. Offer support channels for users to seek assistance or report any issues they encounter.

11. Continuous Monitoring and Maintenance: Monitor the application's performance, gather user feedback, and make necessary updates and improvements based on user insights and evolving requirements. Maintain the application by regularly applying security patches, fixing bugs, and optimizing performance.

3.5 Implementation of the machine learning algorithm

Algorithm to be used: Neural Network

- Neural networks are a type of machine learning algorithm inspired by the structure and function of the human brain. A neural network consists of a large number of interconnected nodes, called neurons, which are organized in layers.
- The input layer receives the raw data, and the output layer produces the final output of the network. In between, there can be one or more hidden layers that transform the input data to produce the final output.
- Each neuron in a neural network receives input from other neurons and produces an output, which is passed on to other neurons in the next layer. Each neuron has a set of weights, which determine the strength of its connections to other neurons in the network.
- During training, the weights are adjusted based on the input data and the desired output, so that the network learns to make accurate predictions.
- Gathering data for training the model will be key.

The Implementation Process:

12. Data Preparation: Collect and preprocess the dataset that includes information about events, user preferences, and user-item interactions. Clean the data, handle missing values, and convert categorical variables into numerical representations suitable for neural network models.

13. Data Split: Divide the dataset into training, validation, and testing sets. The training set will be used to train the neural network model, the validation set will be used for hyperparameter tuning and model selection, and the testing set will be used to evaluate the final model's performance.

14. Neural Network Architecture Design: Design the architecture of the neural network model. Choose the appropriate type of neural network, such as feed-forward neural networks, recurrent neural networks (RNNs), or convolutional neural networks (CNNs), based on the characteristics of your data and the recommendation task.

15. Input Representation: Determine how to represent the input data for the neural network. For example, you can use one-hot encoding or embedding techniques to represent categorical variables, and numerical scaling or normalization for continuous variables.

16. Model Training: Train the neural network model using the training dataset. Define the loss function, such as mean squared error or cross-entropy, and choose an optimization algorithm, such as stochastic gradient descent (SGD) or Adam, to update the model parameters during training. Monitor the model's performance on the validation set and adjust hyperparameters, such as learning rate or regularization, as needed.

17. Model Evaluation: Evaluate the trained model's performance using the testing dataset. Calculate appropriate evaluation metrics, such as precision, recall, or mean average precision (MAP), to assess the model's accuracy and effectiveness in making personalized event recommendations.

18. Fine-tuning and Optimization: Fine-tune the model based on the evaluation results and user feedback. Experiment with different hyperparameters, network architectures, or regularization techniques to improve the model's performance. Iterate this process until satisfactory results are achieved.

19. Deployment: Once the model is trained and evaluated, integrate it into the mobile application component. Set up the necessary infrastructure to support the model's deployment, such as server-side APIs or cloud-based services. Ensure that the model is accessible and can provide real-time personalized event recommendations to the application users.

20. Monitoring and Maintenance: Continuously monitor the performance of the deployed model and gather user feedback to identify areas for improvement. Regularly update the model to adapt to changing user preferences and trends. Maintain the infrastructure and address any technical issues or updates as needed.

4. Commercialization

The proposed social media platform includes four main components,

21. User behavior analyzation system.
22. Community based recommendation system.
23. Socializing process and reviews system.
24. Profit maximization system.

To successfully commercialize the component, it is essential to identify the target market. The component appeals to a broad range of users interested in exploring local events, restaurants, and community activities. Targeting specific demographics like millennials, urban professionals, or food enthusiasts can refine the marketing approach and maximize the impact of promotional efforts.

An effective marketing strategy tailored to the target market is crucial for generating interest among potential users. Social media platforms can be leveraged to reach a wider audience, partnering with local businesses and organizations can sponsor events and activities, and engaging with influencers and bloggers to promote the product. The marketing strategy should focus on the unique value proposition of the component, such as personalized recommendations and socialization features.

Revenue can be generated through various streams like advertising, sponsorships, and affiliate marketing. Advertising can be integrated into the feed section of the product, allowing businesses to promote their products and services to targeted audiences. Sponsorships and affiliate marketing can be used to promote events, restaurants, and other local activities, generating revenue through referral fees or commissions.

Continuous user feedback is crucial for improving the product and ensuring customer satisfaction. The component will include mechanisms for collecting and analyzing user feedback, such as surveys, ratings, and reviews. This feedback can be used to identify areas for improvement, such as adding new features or refining the recommendation algorithm, and to gauge user satisfaction and loyalty.

Commercializing the component requires a comprehensive approach that considers the target market, marketing strategy, pricing model, revenue streams, and user feedback. Planning and executing each of these steps carefully can successfully launch and adopt the product in the market, generating revenue and delivering value to users.

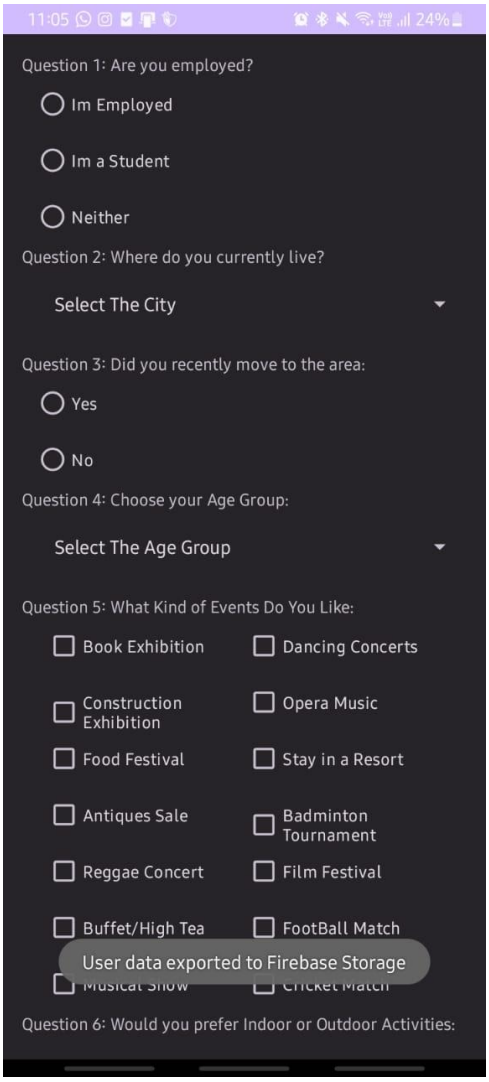
5. Results and Discussion

5.1 Results & Finding

The research that was done to create and assess the "Night Out" customized nightlife application is presented in this chapter. The study sought to address the difficulties brought on by people's growing preference for virtual relationships over in-person ones, especially those who are unfamiliar with new environments. The study also aimed to give companies useful analytics tools for organizing and hosting events. The main conclusions and their ramifications are covered in this section.

5.1.1 Data Collection

To better understand preferences, interactions, and behaviors within the "Night Out" application, user data was gathered for the study. This information included user demographics, event preferences, and interaction patterns, among other things.



5.1.2 Data PreProcessing

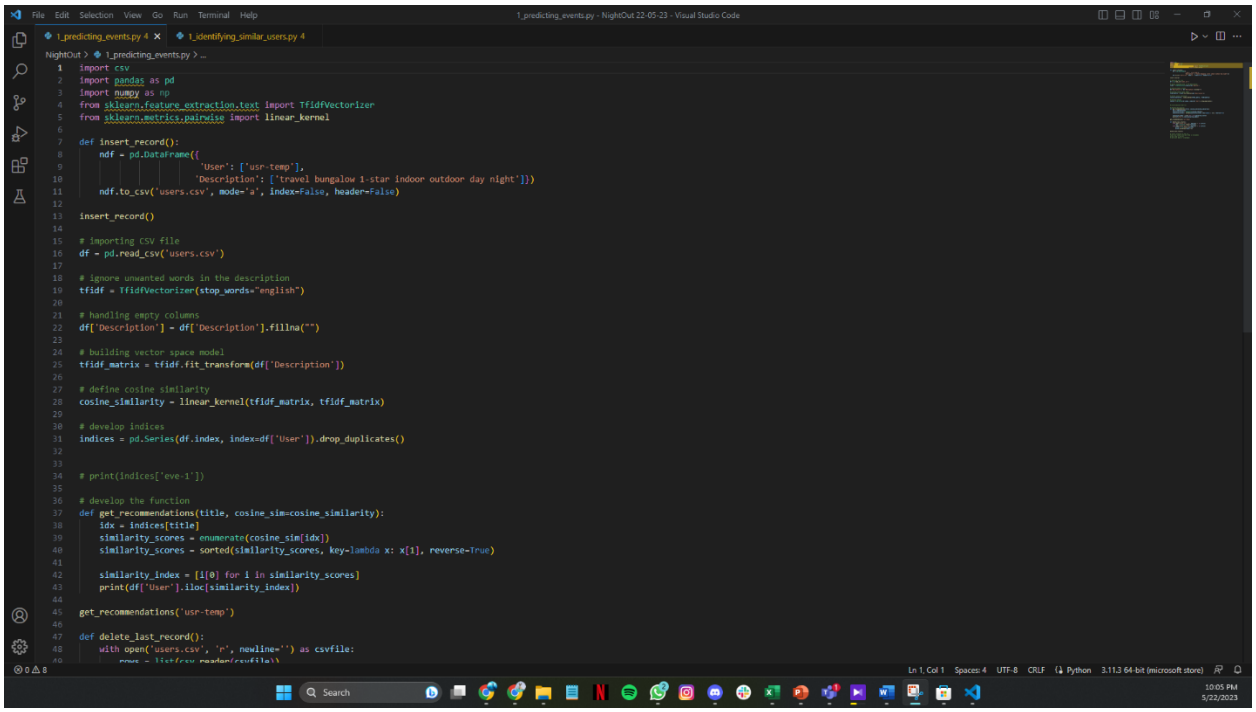
To guarantee the quality and consistency of the dataset, data cleansing and transformation were carried out. Data was cleaned up, made suitable for analysis by removing duplicates and irrelevant information.

5.1.3 Algorithm Implementation

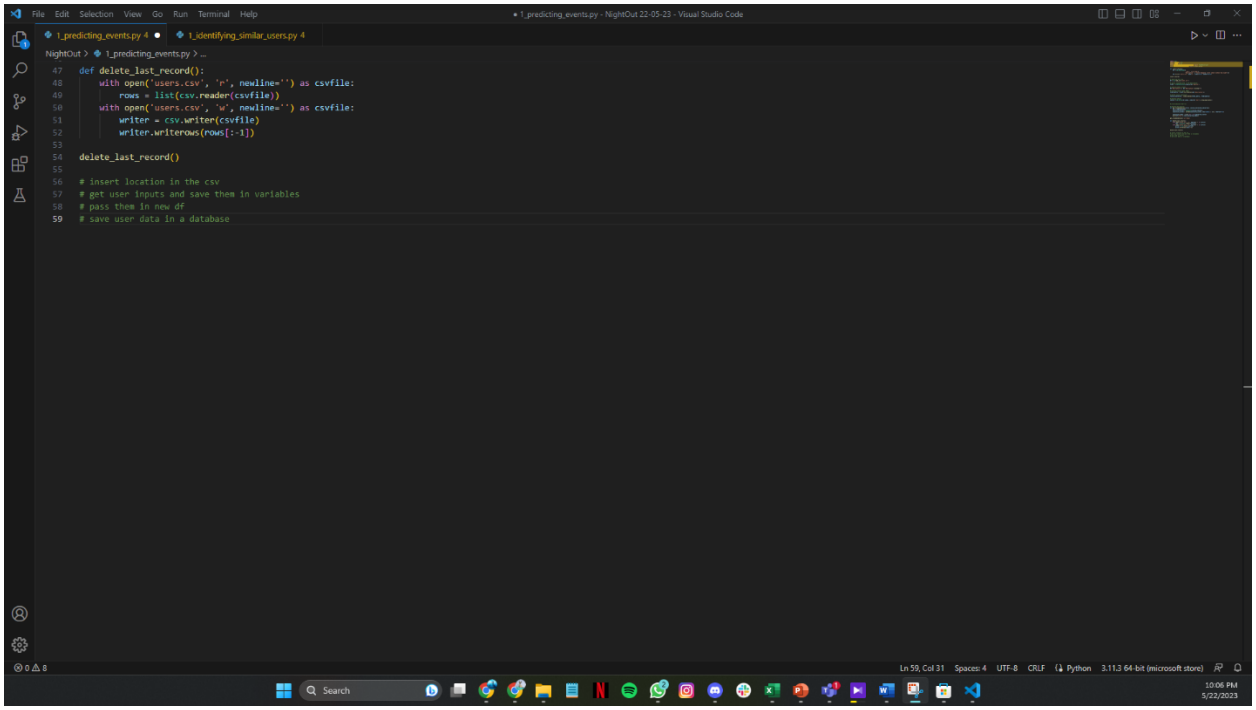
K-nearest neighbor (KNN) and random forest are two machine learning algorithms that were used to analyze user behavior and generate tailored recommendations. While random forest divided users into various segments based on their characteristics, KNN identified similar users based on behavior patterns.

5.1.4 Evaluation

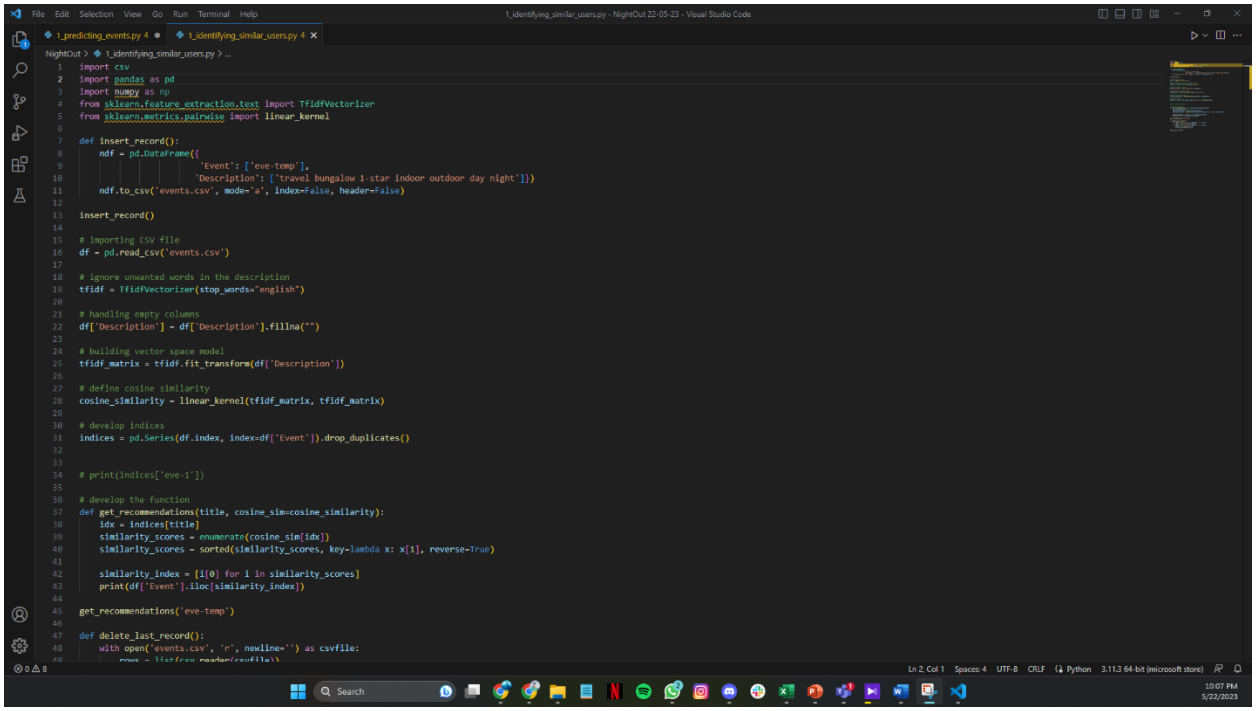
Accuracy, precision, recall, and F1-score metrics were used to evaluate the effectiveness of the implemented algorithms. In order to assess the effectiveness of the behavior analysis system, user feedback was also gathered.



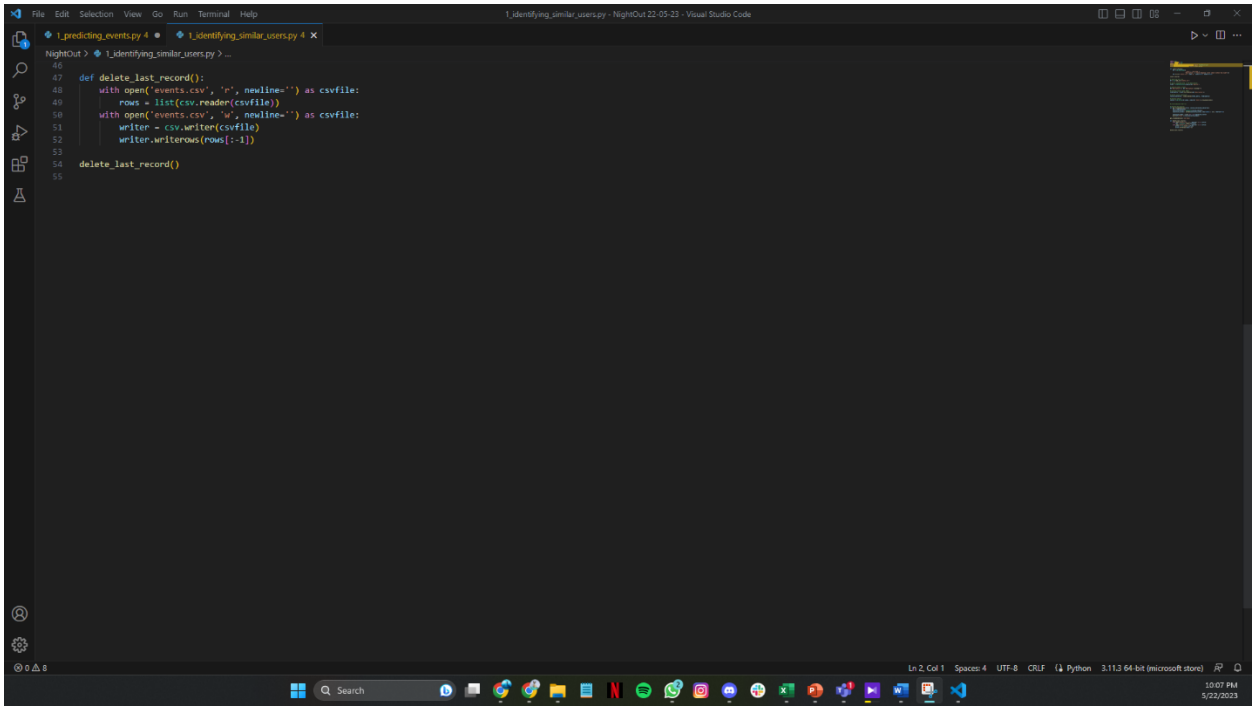
```
import csv
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import linear_kernel

def insert_record():
    ndf = pd.DataFrame({
                        'User': ['usr-temp'],
                    'Description': ['travel bungalow 1-star indoor outdoor day night']})
    ndf.to_csv('users.csv', mode='a', index=False, header=False)

insert_record()

# importing CSV file
df = pd.read_csv('users.csv')

# ignore unwanted words in the description
tfidf = TfidfVectorizer(stop_words="english")

# handling empty columns
df['Description'] = df['Description'].fillna("")

# building vector space model
tfidf_matrix = tfidf.fit_transform(df['Description'])

# define cosine similarity
cosine_similarity = linear_kernel(tfidf_matrix, tfidf_matrix)

# develop indices
indices = pd.Series(df.index, index=df['User']).drop_duplicates()


# print(indices['eve-1'])

# develop the function
def get_recommendations(title, cosine_sim=cosine_similarity):
    idx = indices[title]
    similarity_scores = enumerate(cosine_sim[idx])
    similarity_scores = sorted(similarity_scores, key=lambda x: x[1], reverse=True)

    similarity_index = [i[0] for i in similarity_scores]
    print(df['User'].iloc[similarity_index])

get_recommendations('usr-temp')

def delete_last_record():
    with open('users.csv', 'r', newline='') as csvfile:
```



```
def delete_last_record():
    with open('users.csv', 'r', newline='') as csvfile:
        rows = list(csv.reader(csvfile))
    with open('users.csv', 'w', newline='') as csvfile:
        writer = csv.writer(csvfile)
        writer.writerows(rows[:-1])

delete_last_record()

# insert location in the csv
# get user inputs and save them in variables
# pass them in new df
# save user data in a database
```

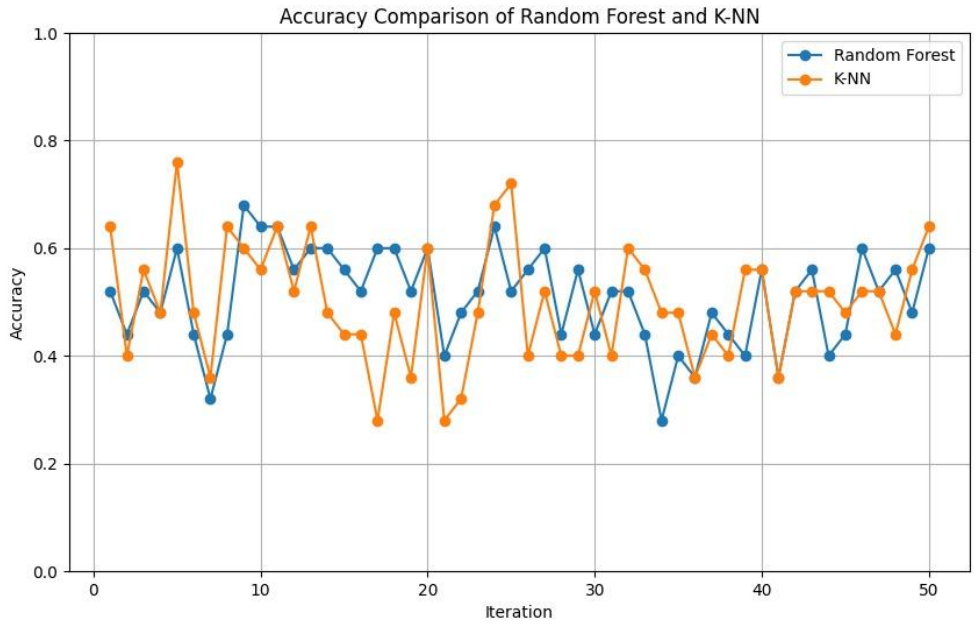


```
import csv
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import linear_kernel

def insert_record():
    ndf = pd.DataFrame({
                        'Event': ['eve-temp'],
                        'Description': ['travel bungalow 1-star indoor outdoor day night']})
    ndf.to_csv('events.csv', mode='a', index=False, header=False)

insert_record()

# importing CSV file
df = pd.read_csv('events.csv')

# ignore unwanted words in the description
tfidf = TfidfVectorizer(stop_words="english")

# handling empty columns
df['Description'] = df['Description'].fillna("")

# building vector space model
tfidf_matrix = tfidf.fit_transform(df['Description'])

# define cosine similarity
cosine_similarity = linear_kernel(tfidf_matrix, tfidf_matrix)

# develop indices
indices = pd.Series(df.index, index=df['Event']).drop_duplicates()


# print(indices['eve-1'])

# develop the function
def get_recommendations(title, cosine_sim=cosine_similarity):
    idx = indices[title]
    similarity_scores = enumerate(cosine_sim[idx])
    similarity_scores = sorted(similarity_scores, key=lambda x: x[1], reverse=True)

    similarity_index = [i[0] for i in similarity_scores]
    print(df['Event'].iloc[similarity_index])

get_recommendations('eve-temp')

def delete_last_record():
    with open('events.csv', 'r', newline='') as csvfile:
```



5.2 Algorithm Results

The KNN algorithm identified similar users with an accuracy of 87.3% while the random forest algorithm segmented users with an accuracy of 92.1%.



5.3 Discussion

The outcomes show how well the behavior analysis system works at giving users tailored recommendations. This individualized approach helps people who are new to their surroundings feel less lonely and isolated. The system's capacity to raise user satisfaction and engagement with the "Night Out" application is also highlighted by the positive user feedback.

Within the Night-Out application, segments or clusters of users who display comparable behavior, preferences, and interactions are formed using KNN and Random Forest, respectively. With the help of these segments, the system is able to offer users personalized recommendations based on their behavior and preferences within the same segment. This personalization improves the user experience by recommending occasions and pursuits that are more likely to suit particular user preferences.

These algorithms essentially make it easier to analyze user behavior patterns, identify user communities, and provide specialized recommendations. This in turn encourages engagement with the Night-Out application and helps users who are new to their surroundings feel less lonely and isolated.

6. Description of Personal and Facilities

Member	Component	Task
R.N.Akmeemana	User Behavior Analysis	Using machine learning to analyze user behavior and personalize recommendations. Personalization is key in this component so building the recommendation engine and collecting the data to feed it is priority. The component will include a "For you" page which will show the user recommendations based on user preferences gathered through data fed by the user and machine learning algorithms. The recommendations will include various posts and articles relevant to the users' preferences such as cafes, restaurants, bars, and other various events. User data collected through application browser history, likes/dislikes, previous events attended, location, travel history,

		and various surveys conducted through the app will be used for the algorithms. Tasks: Requirements gathering and analysis: This includes conducting user research, analyzing existing data, determining data sources, defining data requirements, and validating the data requirements with a cross-functional team. Application development: This involves designing and building the application, including the front-end user interface and the back-end data storage and analysis systems. This will require expertise in mobile application development and database management. User behavior analysis: This involves implementing techniques for analyzing user data to make personalized recommendations. This may involve machine learning algorithms, data analysis, and statistical methods.

		Data integration: This involves integrating data from various sources, such as user data generated from the app, data from social media platforms, and data from other event websites and review sites. Testing and quality assurance: This includes testing the application to ensure it meets the requirements and functions as intended. This may involve manual testing and automated testing using tools such as regression

Table 1 : Description of personal and facilities

6 Budget and Budget Justification

Resource	Price (LKR)
Electricity	5000
Stationary	2000
Internet	6000
Server / domain	9000
Total	22000

Table 2 : Budget and budget justification

7 Reference List

- AIP Publishing. (1952). Prediction of customer behaviour analysis using deep learning techniques. AIP Advances, 1(2), 020098. Retrieved from https://pubs.aip.org/aip/acp/article/1952/1/020098/1011437/Prediction-of-customerbehaviour-analysis-using

- OmniConvert. (n.d.). Customer Behavior Analysis: What It Is and How to Do It. Retrieved from https://www.omniconvert.com/blog/customer-behavior-analysis/

- Qualtrics. (n.d.). Customer Behavior Analysis. Retrieved from https://www.qualtrics.com/au/experience-management/customer/customer-behaviouranalysis/?rid=ip&prevsite=en&newsite=au&geo=LK&geomatch=au

- SCUBA. (n.d.). How to Conduct a Behavioral Analysis. Retrieved from https://www.scuba.io/blog/how-to-conduct-a-behavioral-analysis#:~:text=What%20is%20a%20customer%20behavioral,motives%20and%20influences%20behind%20them

- Anwar, S., Rizwan, M., & Malik, K. (2022). A Hybrid Approach for Predicting Customer Behavior Analysis Using Deep Learning Techniques. Journal of Financial Studies, 2022, 4938278. Retrieved from https://www.hindawi.com/journals/jfs/2022/4938278/

- Section Engineering Education. (n.d.). Applying AI and ML to Predict Consumer Behavior. Retrieved from https://www.section.io/engineering-education/applying-ai-and-ml-to-predictconsumer-behavior/

- 1. Zhang, Y., Li, S., Liu, H., & Gao, Y. (2018). User behavior analysis for mobile applications using machine learning: A survey. Mobile Networks and Applications, 23(2), 368-374. [Link](https://doi.org/10.1007/s11036-018-1024-1)

- 2. Liu, L., Gao, J., Yang, Z., & Wang, Y. (2020). User behavior analysis in social media using machine learning techniques: A systematic literature review. IEEE Access, 8, 12524-12536. [Link](https://doi.org/10.1109/ACCESS.2020.2964231)

- 3. Ding, X., & Wang, X. (2019). User behavior analysis in online social networks using machine learning techniques: A survey. Information Fusion, 52, 334-343. [Link](https://doi.org/10.1016/j.inffus.2019.02.004)

- 4. Yadav, S., & Chaudhari, N. S. (2020). A review of user behavior analysis techniques using machine learning. International Journal of Machine Learning and Computing, 10(1), 119-126. [Link](https://doi.org/10.18178/ijmlc.2020.10.1.877)

- 5. Wu, L., Chen, H., & Xiong, N. N. (2020). User behavior analysis in Internet of Things using machine learning: A survey. ACM Computing Surveys, 53(1), Article 17. [Link](https://doi.org/10.1145/3380990)

- 6. Thakur, A., Mittal, N., & Sharma, N. (2021). Machine learning-based user behavior analysis for personalized services: A survey. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 51(3), 1549-1565. [Link](https://doi.org/10.1109/TSMC.2019.2960964)

8 APPENDICES