**Super Characterization Document**

**Project 606 –**

**Software Development: Café recommendation system for coffee enthusiastic**

****

**Written By:**

Ahigad Genish

Omer Sela

**Project Supervisors**: Professor Geles Ran

Professor Izack Cohen

Date: 07/01/2024

Table of contents:

Objectives -------------------------------------------------------------- 3

Application System ----------------------------------------------------- 5

Technology and infrastructure --------------------------------------- 9

Realization (Implementation) ---------------------------------------- 12

**1.0 - Objectives**

The primary goal of this project is to develop a recommendation system for coffee shops that goes beyond the traditional approach of user ratings and incorporates machine learning to identify and recommend cafes based on quality factors. The specific targets include:

Addressing the reliability issues and biases in existing recommendation systems.

Focusing on the preferences of coffee lovers who appreciate handcrafted, professionally made coffee.

Developing a software system that utilizes machine learning to analyze images and identify key characteristics defining quality coffee shops.

Creating a user-friendly website that provides personalized recommendations, weighing machine learning results alongside other relevant scores.

Testing the system on coffee shops worldwide to assess its effectiveness and accuracy.

1.1 - General Target: The overarching goal of the recommender system for quality coffee shops is to redefine the traditional approach to recommendation systems, particularly in the context of coffee establishments. The system aims to cater to the preferences of discerning coffee enthusiasts by incorporating machine learning and image detection technologies to identify and recommend high-quality coffee shops worldwide.

1.2 - Extended Information about the Targets:

1.2.1 - Practical Targets:

\* Collect a diverse dataset of coffee shop images for training the machine learning model.

\* Use a machine learning model capable of accurately analyzing images to identify quality-related features.

\* Identify and locate specific objects within the images that are indicative of quality coffee preparation.

\* Design and implement a user-friendly website that seamlessly integrates the machine learning system with Google Maps.

\* Establish an efficient scoring mechanism that combines machine learning results with other relevant scores to provide personalized recommendations.

1.2.3 Future Targets:

Explore the potential integration of user feedback mechanisms to continuously enhance the system's recommendation accuracy.

Investigate opportunities to expand the recommendation system to include other features to rating according to the photographs of the coffee beans for example.

1.3 Problems:

1.3.0 General Problems:

Not all coffee shops may have images uploaded, limiting the system's coverage.

A Photograph of a coffee not always indicate about the tasty.

1.3.1 Problems that the System is Solving:

Addressing Reliability and Bias: Overcoming the reliability issues and biases associated with traditional recommendation systems that rely solely on user ratings.

Catering to Coffee Enthusiasts: Providing a recommendation system tailored to the preferences of coffee enthusiasts who appreciate the artistry and craftsmanship involved in quality coffee preparation.

1.3.2 Problems that the System May Cause:

Potential Exclusion: Coffee shops without uploaded images may not benefit from the recommendation system, potentially excluding certain establishments from user consideration.

**2.0 - Application**

The essence of the system lies in redefining the conventional recommendation approach by focusing on quality coffee attributes. The system will:

Analyze images uploaded by users on Google Maps to identify features indicative of quality coffee preparation.

Use machine learning algorithms to assess characteristics such as the presence of professional coffee machines, the skill demonstrated in coffee preparation, and other factors contributing to quality.

Integrate these machine learning results with other relevant scores to create a holistic recommendation suitable for coffee enthusiasts.

Cater specifically to individuals who appreciate the craftsmanship and skill involved in coffee preparation, rather than relying solely on general user ratings.

2.1 Users: The application is designed to cater to primary user groups:

Consumers:

Coffee enthusiasts seeking personalized recommendations for quality coffee shops.

Coffee Shop Owners/Managers:

Owners or managers of coffee shops interested in understanding and improving their establishments' quality perception.

Access insights into the factors contributing to their coffee shop's quality score.

Receive recommendations on potential improvements to enhance quality perception.

Engage with user feedback to address concerns or highlight positive aspects.

2.2 Tangential Systems: The recommender system seamlessly integrates with key external systems to enhance its capabilities and deliver an enriched user experience.

Google Maps API:

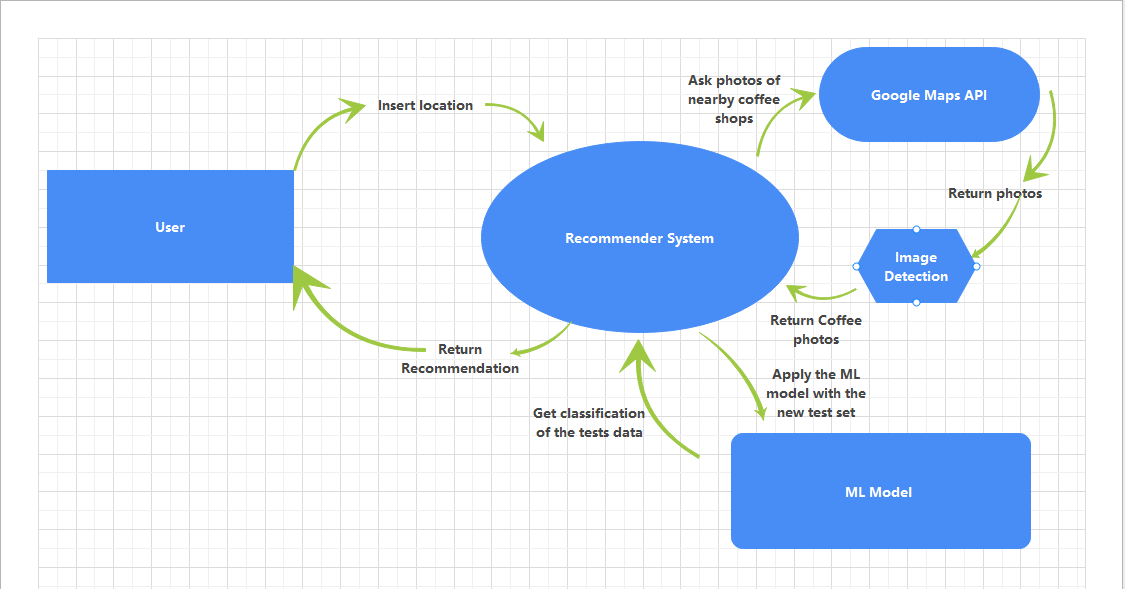
Purpose: The system leverages the Google Maps API to facilitate image uploads, retrieve geolocation data, and ensure a smooth user experience. This integration streamlines the process of incorporating location-based information and visuals directly from Google Maps, enhancing the accuracy of recommendations.

Machine Learning Model:

Purpose: Integration with a pre-trained machine learning model specifically tailored to analyze images from coffee shops. The model is designed to classify images based on quality-related features, extracting valuable insights that contribute to the determination of the coffee shop's overall quality. This collaboration empowers the recommender system to identify and recommend coffee shops with exceptional coffee, aligning with the preferences of discerning users.

ראש הטופס

2.3 Context Diagram: A context diagram illustrates the interactions between the recommender system and external entities.



2.3.2 Interactions:

Users interact with the Recommender System through the website.

The Recommender System communicates with the Google Maps API for image upload and geolocation data.

The image detection retrieve the relevant item in the returned photos.

The Recommender System create test set and apply the ML Model to classify this set.

The recommender system rating the results and return the top 5 results by rating to the user.

2.4 Logs Files - Central Information Entities: Log files serve as central information entities, capturing crucial data points for system monitoring, improvement, and debugging.

2.4.1 Types of Logs:

User Interaction Logs: Record user interactions with the system.

Machine Learning Logs: Capture details of the machine learning process, such as image analysis results, feature extraction, and quality scores assigned.

Error Logs: Document any system errors, exceptions, or issues encountered during operation.

Feedback Logs: Store user feedback, comments, and ratings for analysis and system refinement.

2.4.2 Log Management:

Utilize centralized log management tools to efficiently store, analyze, and monitor log data.

Implement secure and scalable log storage solutions to ensure data integrity and accessibility for system administrators and analysts.

Ensuring effective management and utilization of log files is essential for maintaining system performance, identifying areas of improvement, and enhancing the overall user experience.

**3.0 Technology and Infrastructure:**

The technology and infrastructure of the system will involve:

**Machine Learning Algorithms:** Implementing ML models capable of analyzing images and extracting relevant features that define quality coffee shops. This may include deep learning techniques for image recognition.

**Data Collection and Classification:** Collecting a diverse dataset of images from coffee shops worldwide, classifying them based on quality criteria, and using this dataset to train the machine learning system.

**Software Engineering:** Developing a robust website that incorporates the machine learning system, user interfaces for input, and algorithms for result interpretation. Ensuring scalability, security, and efficiency in the software design.

**Integration with Google Maps:** Utilizing the Google Maps platform for image upload and recommendation delivery. Ensuring seamless integration with the existing infrastructure for a user-friendly experience.

3.1 Databases: The system relies on robust databases to store, manage, and retrieve data efficiently. Two primary types of databases are employed:

Image Database:

Purpose: Stores a diverse dataset of coffee shop images used for training the machine learning model.

Features:

High-capacity storage for large image datasets.

Efficient retrieval mechanisms for quick access during model training.

Scalability to accommodate the continuous expansion of the dataset.

Features:

Transactional support for reliable user data handling.

Indexing and query optimization for quick retrieval of user-specific information.

Backup and recovery mechanisms to safeguard critical operational data.

3.2 Development Tools:

Programming Languages:

Python:

Role: Core programming language for system development.

Features:

Wide range of libraries and frameworks.

Support for machine learning, image processing, and web development.

Community-driven and extensively used in data science applications.

Machine Learning Framework:

PyTorch:

Role: Framework for building and training machine learning models.

Features:

Dynamic computation graph for flexibility.

Extensive support for neural network architectures.

Integration with popular deep learning libraries.

Image Processing Libraries:

OpenCV:

Role: Utilized for image processing tasks within the system.

Features:

Comprehensive library for computer vision tasks.

Efficient image manipulation and analysis functions.

Cross-platform support for diverse operating systems.

Web Development Framework:

React:

Role: The React library is utilized for building the user interface of the website.

Features:

Component-based architecture for modular and reusable UI elements.

Virtual DOM for efficient updates and rendering.

Seamless integration with other libraries and frameworks for a flexible development environment.

Features:

Model-View-Controller (MVC) architecture for structured development.

Built-in security features and scalability support.

Extensive documentation and an active community.

3.2.5 APIs:

Google Maps API:

Role: Facilitates integration with Google Maps for image uploads and geolocation data.

Features:

Access to mapping services and location-based functionalities.

Seamless integration with the system's user interface.

By leveraging these technologies and tools, the system ensures a robust, scalable, and efficient infrastructure for developing, training, and deploying the recommender system for quality coffee shops.

**4.0 Implementation:**

The project will be executed in several stages:

**Defining Software and Capabilities:** Clearly defining the functionalities and capabilities of the software based on the project idea. Outlining the specific features that the machine learning system will focus on.

**Collecting and Classifying Images:** Gathering a diverse dataset of coffee shop images, classifying them based on quality criteria, and using this dataset to train the machine learning system.

**Image Detection:** Use an Image detection model that retrieve relevant items from a photo. Specifically, a cup of coffee.

**Website Development:** Designing and developing a user-friendly website that integrates the machine learning system and provides personalized recommendations for coffee lovers.

**Testing on Cafes Worldwide:** Implementing the system on a global scale to test its effectiveness and accuracy in recommending quality coffee shops. Gathering feedback and refining the system based on real-world results.

**4.1 Management:** Regular meetings with project leaders will be conducted on a weekly basis to ensure effective communication, project alignment, and timely issue resolution. Meeting summaries will be compiled, outlining key discussions, decisions, and action items. Derived task actions will be assigned, tracked, and updated during subsequent meetings to maintain project momentum.

**4.2 Development Team:** The development team comprises two members, Ahigad Genish & Omer Sela, collaboratively working on different aspects of the project. The team's collaborative effort aims to leverage diverse skills, promote knowledge sharing, and ensure a well-rounded development approach.

**4.3 Scheduling:** *assume tasks scheduling:*

**1. Requirements and Design (3 weeks)**

* **Task1.1:** Requirements Gathering (e.g., finalizing functional and non-functional requirements).
  + **Timeline:** 08/01/24 - 15/01/24
* **Task 1.2:** System Design (e.g., architecture design, database schema).
  + **Timeline:** 16/01/24 - 29/01/24

**2. Machine Learning Model Development (5 weeks)**

* **Task 2.1:** Data Collection (e.g., gathering images of coffee shops).
  + **Timeline:** 30/01/24 - 06/02/24
* **Task 2.2:** Model Training (e.g., training the ML model on image data).
  + **Timeline:** 07/02/24 - 04/03/24
* **Task 2.3:** Model Evaluation (e.g., testing the model for accuracy).
  + **Timeline:** 04/03/24 - 11/03/24

**3. Integration with Third-Party Services (3 weeks)**

* **Task 3.1:** Google Maps API Integration (e.g., image upload, geolocation retrieval).
  + **Timeline:** 12/03/24 - 25/03/24
* **Task 3.2:** Integration with Pre-prepared Object Recognition Model.
  + **Timeline:** 26/03/24 - 01/04/24

**4. Web Development (6 weeks)**

* **Task 4.1:** Frontend Development (e.g., React development for user interface).
  + **Timeline:** 02/04/24 - 15/04/24
* **Task 4.2:** Backend Development (e.g., server, API development).
  + **Timeline:** 16/04/24 - 29/04/24
* **Task 4.3:** Integration of ML Model and object detection Model with Web Interface.
  + **Timeline:** 30/04/24 - 13/05/24

**5. Testing and Validation (2 weeks)**

* **Task 5.1:** System Testing (e.g., testing the complete system functionality).
  + **Timeline:** 14/05/24 - 27/05/24

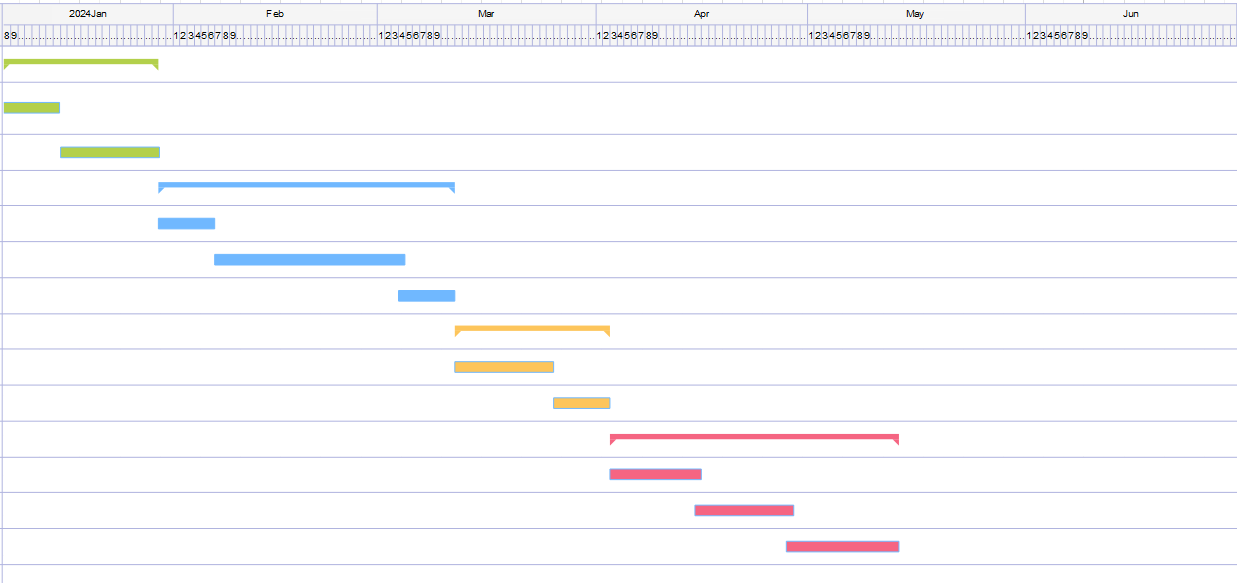
**6. Documentation and Reporting (1 week)**

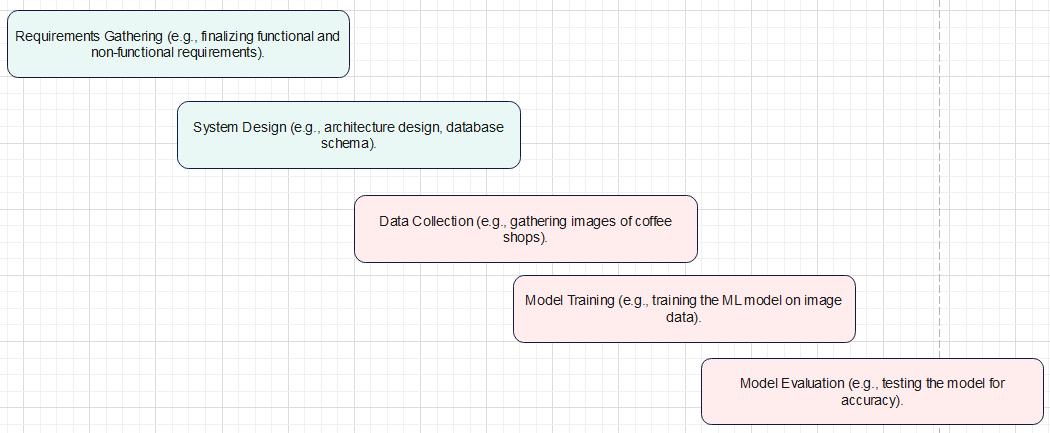
* **Task 6.1:** Technical Documentation (e.g., system documentation, API docs).
  + **Timeline:** 28/05/24 - 03/06/24

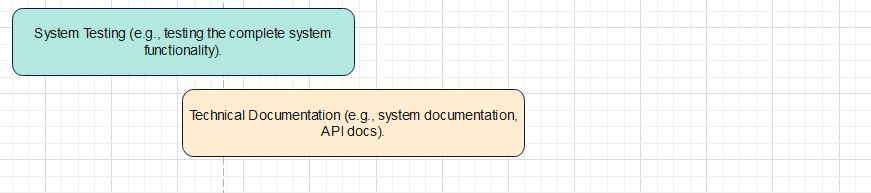
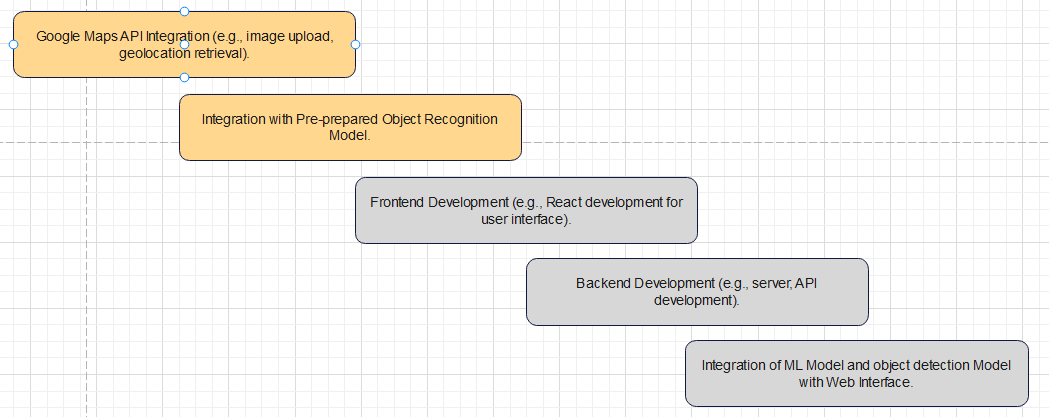
**7. Time buffer (4 weeks)**

* Extra time for changes.
  + **Timeline:** 03/06/24 - 01/07/24









**4.4 Development Method:** The Agile development methodology will be adopted to ensure adaptability and responsiveness to evolving requirements. Regular sprint cycles, typically on a weekly basis, will be employed to assess progress, evaluate priorities, and adjust the development trajectory accordingly. This iterative approach allows for flexibility in accommodating changes and prioritizing high-value features.

**4.5 Documentation:** Every sprint will include an update to the Git repository containing relevant design documents. These documents will encompass key aspects of the project, including system architecture, machine learning model specifications, and user interface design. Documentation updates will serve as a comprehensive reference for the team and stakeholders, facilitating effective collaboration and knowledge sharing.

By adhering to these implementation strategies, the project aims to foster efficient teamwork, adaptability to changing requirements, and transparent communication throughout the development lifecycle.

**4.6 Tests:** A robust testing strategy will be implemented to evaluate the system's performance and accuracy. The testing phase will involve comprehensive checks against coffee shops worldwide, comparing the system's recommendations with existing Google Reviews results. This process aims to validate the effectiveness of the recommendation system.

Test Scope:

Evaluate the system's recommendations for a diverse set of coffee shops globally.

Compare the system's results with existing Google Reviews for the same coffee shops.

Assess the accuracy, relevance, and alignment of the system's recommendations with user expectations.

Test Execution:

Location-Based Testing: Conduct tests in various geographic locations to ensure the system's adaptability to different coffee cultures.

Diversity in Coffee Shops: Include a mix of renowned, local, and niche coffee shops to validate the system's ability to cater to diverse preferences.

Test Criteria:

Accuracy: Measure the alignment between the system's recommendations and the actual quality of coffee shops.

Relevance: Evaluate how well the recommendations match the preferences of users seeking high-quality coffee experiences.

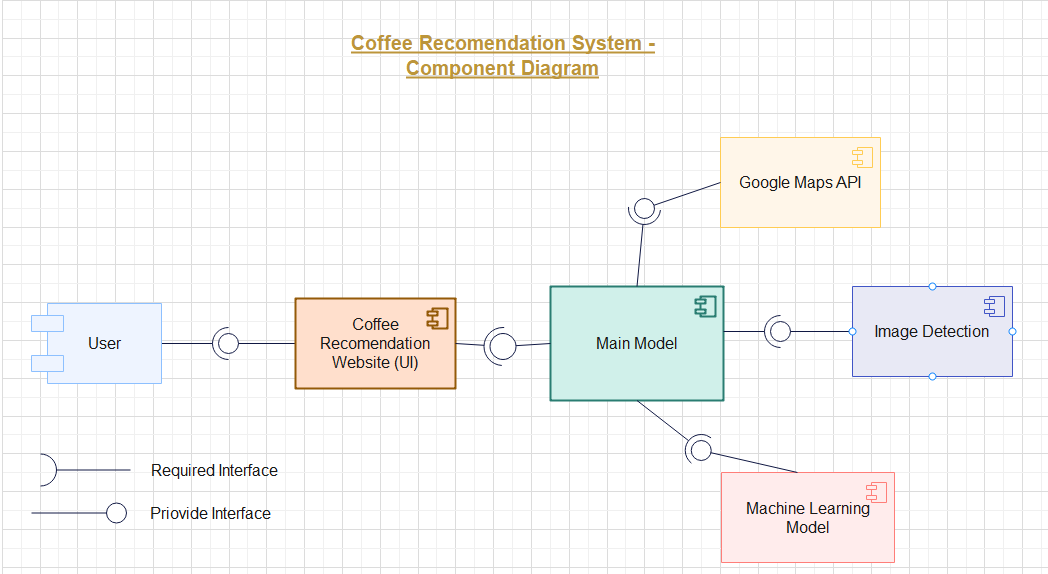
Consistency: Assess the system's performance across different locations and types of coffee shops.

Comparison with Google Reviews:

Benchmarking: Utilize Google Reviews as a benchmark to compare and validate the accuracy of the system's recommendations.

User Ratings: Analyze the correlation between user ratings on Google Reviews and the quality scores assigned by the system.

**Component Diagram:**

****