GRADUATE CERTIFICATE
INTELLIGENT REASONING SYSTEM (IRS)

Project Report
Indoor Green Guide

GROUP 8

CAI Jingjing (A0285758H)

GAO Zihui (A0285749H)

LI Dongshuo (A0285829J)

LI Jixuan (A0285733W)

WANG Jiangqianyi (A0285765L)

Table of Contents

Chapter 1: Introduction

In the era of information and technology, digital solutions that improve our experiences and offer personalized suggestions are becoming more and more integrated into our everyday lives. In this regard, our project aims to create a connection between the technological and natural worlds. Our goal was to develop an intelligent system that assists customers in selecting indoor plants that are appropriate for their homes based on plant size and environmental preferences. The idea was to simplify the often-overwhelming process of selecting the right indoor plants and make it accessible to a broad audience.

This report presents the culmination of our efforts to design, develop, and evaluate this intelligent plant recommendation system. We will delve into the architecture, algorithms, and technologies employed to bring this system to life. Moreover, we will discuss the validation of the system, sharing insights into how well it performed in different aspects, including frontend usability and backend recommendation accuracy. By the end of this report, we aim to provide a holistic view of our system, its achievements, and the potential for further enhancements in the realm of intelligent reasoning and recommendation systems.

Chapter 2: Business Justification

This chapter explores the challenges consumers face when choosing indoor plants and identifies the potential for personalized recommendation systems to fill the gap in the market, offering tailored guidance for plant selection.

2.1 Background

Indoor air pollution can lead to several health issues at home or work. Potentially toxic gases and particular matter (PM) can be released by a variety of indoor sources and activities of occupants, including furnishings, paints, varnishes, waxes, carpets, solvents, cleaning supplies, office equipment such as copiers and printers, gas cooktops, and cigarettes [1].

Nitrogen oxides (NO and NO_2), polycyclic aromatic hydrocarbons (PAHs), carbon monoxide and dioxide (CO and CO_2), and volatile organic compounds (VOCs; such as formaldehyde and benzene) are prevalent indoor air pollutants [2]. Since people in industrialized countries spend more than 80% of their lives indoors, the build-up of air pollutant concentrations to dangerous levels, especially in modern energy-saving but air-tight constructions, represents one of the priority concerns for human health today [3].

Continuous exposure to air pollutants, the concentration of which indoors can even be higher than outdoors, may cause respiratory and cardiovascular diseases eventually contributing to the so-called ‘sick building syndrome’ (SBS; see Glossary) and ‘building-related illnesses’ (BRI). One of the major concerns regards formaldehyde, a widespread hazardous air pollutant that is released over the long term from aging furniture and pressed-wood products, and that is likely to have carcinogenic effects in humans [4].

Indoor plants offer an effective and natural means to improve indoor air quality. NASA's 1989 demonstration of plants' ability to dramatically lower carbon dioxide and volatile organic compound (VOC) concentrations in sealed Spaces [5] sparked a number of studies on plants' capacity to absorb chemicals due to the increased awareness that houseplants could contribute to better air quality. Since then, research on the effects of houseplants on different environments, especially on living environments, has progressed by leaps and bounds. Numerous studies have indicated that indoor plants can have several positive effects, such as boosting moods, mitigate noise levels, enhancing productivity, sharpening focus, reducing stress and fatigue, purifying indoor air by absorbing toxins, elevating humidity, and generating oxygen.

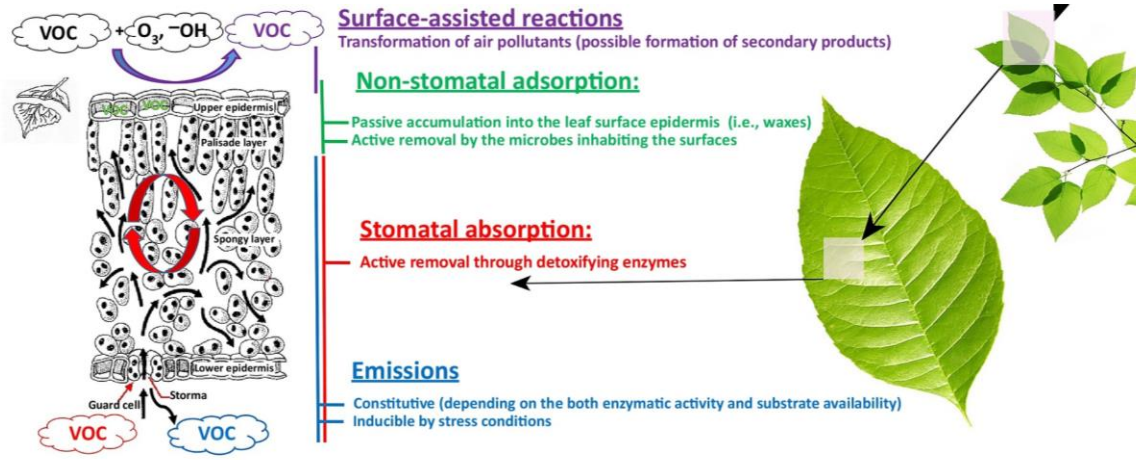

Figure 1. Details of the various processes potentially involved in the interaction between indoor volatile organic compounds (VOCs) and the plant foliage [6]

2.2 Problem Statement

Since the onset of the COVID-19 pandemic, the indoor plant market has experienced remarkable growth. Post 2020, web searches for indoor plants have surged significantly in comparison to pre-pandemic levels (shown in Figure 2). This surge can be attributed to people spending more time indoors and a heightened emphasis on enhancing indoor air quality.

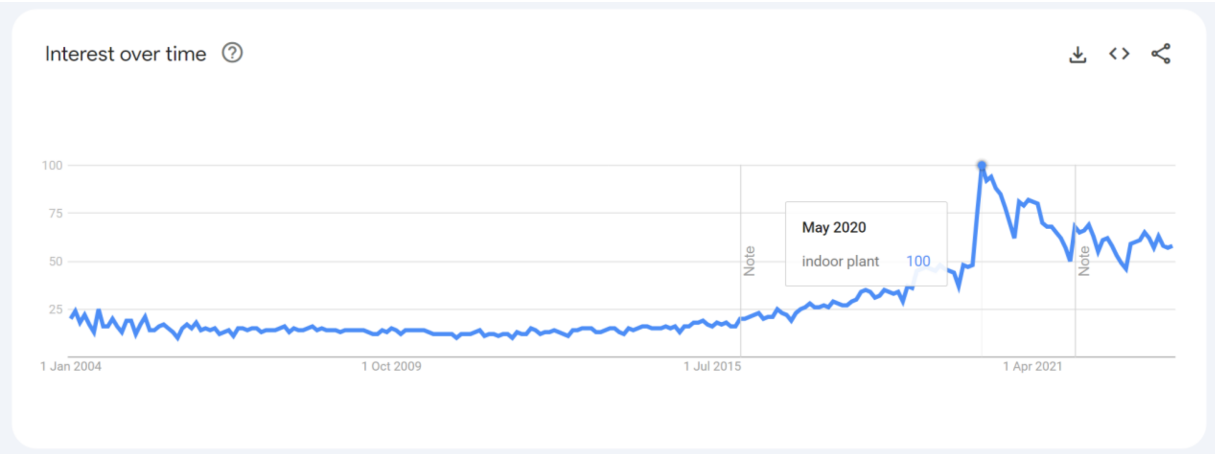

Figure 2. Google trends of indoor plant in worldwide from 2004 to 2023[7]

Furthermore, the increasing popularity of indoor plants is not limited to web searches alone. The global import values of living plants by country in 2022, as depicted in Figure 3, reveal a substantial uptick in the demand for indoor plants. This surge in imports is indicative of the rising global interest in indoor plant cultivation and the associated economic impact.

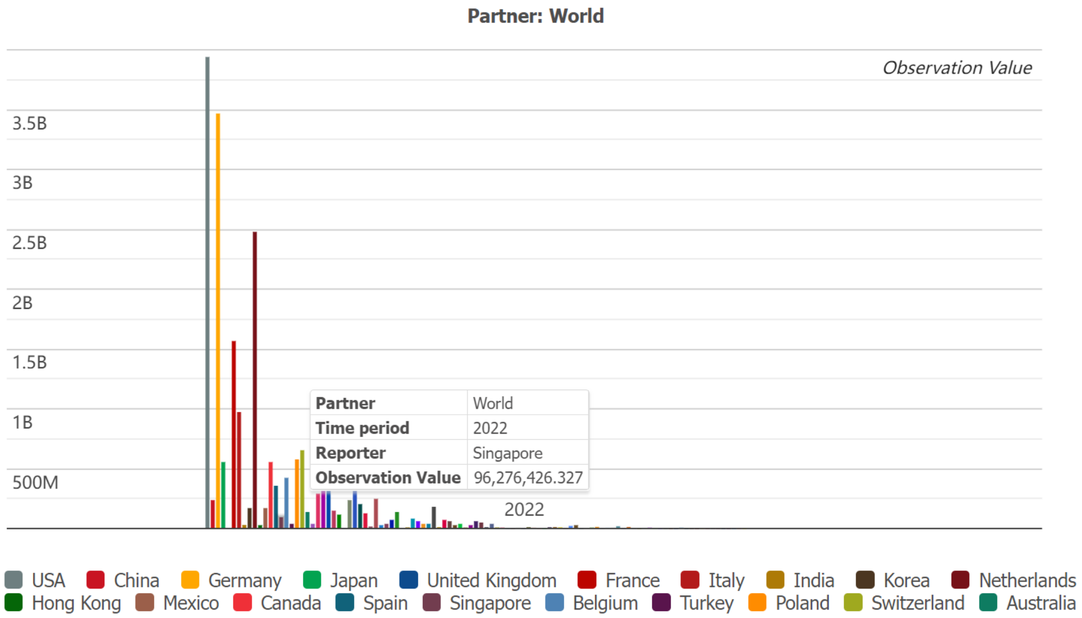

Figure 3. Column chart showing import values of living plant by country in 2022 [8]

When individuals embark on the journey of acquiring indoor plants, they are confronted with two primary avenues for decision-making. The first path involves turning to the vast landscape of the internet, where they seek out recommended plant selections through their web browsers. The second route leads them to physical stores, where they rely on the expertise of store personnel to guide them in making their choices.

However, this seemingly straightforward task of choosing the perfect indoor plant is often laden with challenges. Consumers may feel uncomfortable and hesitant when making decisions due to a lack of information about plants and a limited comprehension of the distinctive qualities of different species. The lush and diverse world of indoor flora can seem overwhelming, and consumers may find themselves ill-equipped to make informed choices.

Intriguingly, the advice they get is often impacted by a variety of outside variables, chief among them being the alluring power of marketing, even when they use the internet or speak with in-store specialists. Customers' decisions may not always reflect their true preferences or the particular features of their living spaces due to these subtle yet potent effects.

Given these challenges, it becomes evident that there is a compelling need for a system that can simplify this process and provide consumers with tailored recommendations. Such a system would not only alleviate the confusion and uncertainty associated with plant selection but also empower consumers with the knowledge they need to make choices that genuinely resonate with their preferences and living environments.

2.3 Business Opportunity

The burgeoning indoor plant market has paved the way for a burgeoning trend and, indeed, a significant business opportunity: personalized and customized services. As we've conducted extensive research through methods like Google web searches, it's clear that customers now have a multitude of convenient avenues to access plant-related information. The primary sources for plant information have been gardening websites and plant sales platforms, as illustrated in Figure 4.

Gardening websites typically serve as valuable hubs for disseminating horticultural knowledge, aiming to educate and engage plant enthusiasts. In contrast, plant sales websites are primarily focused on furnishing customers with intricate details about the plants available for purchase, providing in-depth information that facilitates informed buying decisions.

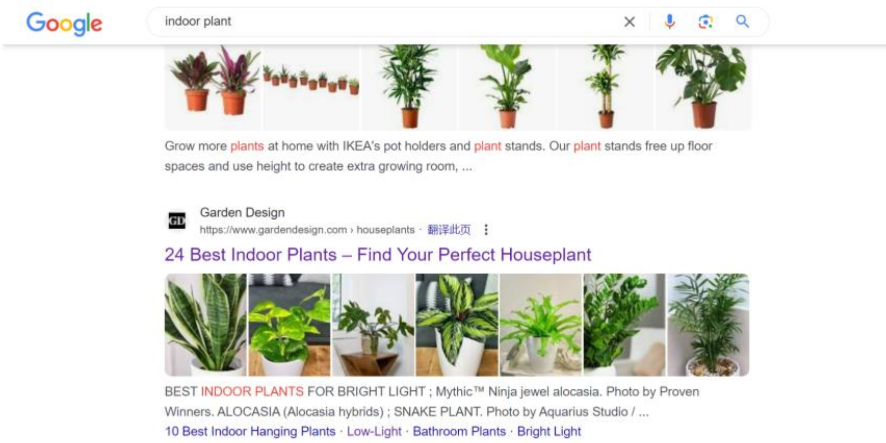

Figure 4. Google result of indoor plant

Despite the wealth of information at the fingertips of plant aficionados, there remains a noticeable gap in the market. There's a notable absence of websites or systems that can provide relatively objective and personalized recommendations for indoor plants. In this context, we see a compelling business opportunity.

By developing a houseplant personalized recommendation system, we aim to bridge this gap, offering a unique and valuable service to a growing audience of plant enthusiasts. Our objective is to create a system that can cater to consumers' specific needs and offer recommendations for suitable plant varieties. The system will also give pricing references and vital details about each facility in addition to these recommendations.

This system will take into account various factors comprehensively, including the plant's lighting requirements, size, care duration (considering watering and fertilization needs). This system will also categorize plants into various types, offering a curated selection that aligns with consumer tastes. Whether someone is a fan of succulents, tropical plants, or flowering varieties, our system will offer tailored suggestions. Furthermore, for those with unique preferences, it will consider special plant characteristics such as the ability to flower, or the adaptability to thrive without traditional soil. Through this process, we hope to help customers choose indoor plants that best suit their needs and tastes.

In essence, our system aspires to be the ultimate companion for anyone embarking on their indoor plant journey. It is not just about recommending a plant; it's about empowering users with knowledge, offering them choices that perfectly suit their space, preferences, and lifestyle. By doing so, we aim to make the process of selecting indoor plants an enjoyable and informed experience, enriching the lives of plant enthusiasts and fostering a deeper connection with the world of indoor plant.

Chapter 3: Data Collection and Preprocessing

This chapter delves into the critical aspect of data collection and the meticulous process of dataset construction. We provide insights into the methods employed for web data scraping and data preprocessing.

3.1 Data Source

The data source is a website, Candy Floriculture (https://www.candy.com.sg), that offers clients indoor plants for sale. They specialize in plants or gardening products and provide various plant types of indoor plants, such as tropical plants, succulents, and so on.

Customers' confidence in their purchases can be boosted and their level of satisfaction can rise when websites offer comprehensive information on the upkeep and care of each indoor plant. Furthermore, providing a large selection of indoor plant sizes and kinds can assist clients in selecting the ideal plants for their requirements and tastes. You may provide clients the chance to design their own distinctive and lovely interior environments by providing a wide assortment of indoor plants.

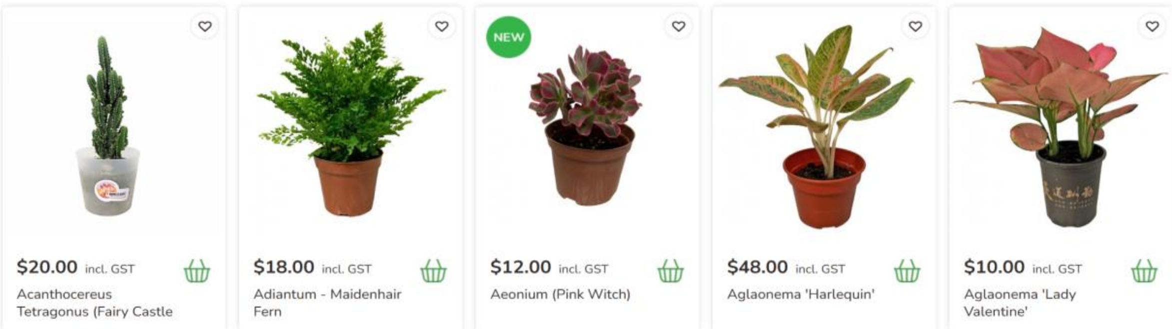

Figure 5. Items displayed on Candy Floriculture

3.2 Data Scraping

To collect the necessary data for our project, a Python script called **dataCollect.py** was developed. This script scrapes data from a specific website by utilizing Chrome developer tools and the XHR object, which typically contains data in JSON format. We extract crucial header information, including 'accept,' 'accept-encoding,' 'host,' 'cookies,' and more. With this header information, along with page and URL details, we systematically scrape data from the website and save it in CSV format. The script is designed to aggregate data from various website pages, playing a vital role in our data collection and preprocessing workflow.

3.3 Data Preprocess

Raw data from the indoor plant website was cleaned and structured by a Jupyter Notebook file during the data preprocessing phase. The data was transformed into a structured dataframe format using the Pandas library for further analysis. During this preprocessing step, columns that were deemed irrelevant were removed. Notably, the 'price' column required special handling; it was converted into a string format to eliminate non-numeric characters.

The content within the 'content' column was rich and diverse, containing a wealth of information. To enhance data clarity and analysis, we applied a process to separate this content based on the '/n/n' annotation. After this separation, each section of content was appended as individual columns to the dataframe. These new columns included product measurements, general information, soil requirements, water requirements, etc. Following this extraction and transformation, the original 'content' column, which was now redundant, was removed from the dataset. This meticulous data preprocessing procedure allowed us to work with a clean and well-structured dataset for our analysis, as shown in Figure 6.

content	processed	num_columns	DISCLAIMER	GENERAL INFORMATION	SOIL REQUIREMENT	FEEDING FERTILIZER? REQUIREMENT	LIGHT REQUIREMENT	WATER REQUIREMENT	PRODUCT MEASUREMENT
DISCLAIMER\n\nThe actual plant may differ as e...	20.00	16	The actual plant may differ as each pot has it...	The Acanthocereus tetragonus ¡®Fairy Castle Ca...	Suitable for light (sandy), medium (loamy)?soi...	Fertilize?with a good cactus? fertilizer?in spr...	It does well in partial?sun?to partial shade.	The best way to?water?your? Acanthocereus tetra...	NaN
DISCLAIMER\n\nThe actual plant may differ as e...	18.00	18	The actual plant may differ as each pot has it...	Adiantum, the maidenhair fern, is a genus of a...	Most ferns grow best in slightly acidic soils;...	Fertilize monthly year-round with half-strengt...	Maidenhair ferns need shady settings.\nDo not ...	While it needs to be kept moist as part of its...	Plant Overall Height APPROXIMATELY:-\n16cm for...

Figure 6. Data basic info

Cause some plant detailed page have uncomplete information, lack some information such as product measurement and detailed information title which is useless, we should drop them and add empty column to those who do not have complete detailed information. And after splitting, some plants have 16 detailed information columns while some other have 18 columns. These two have 354 items while total are almost 400 items, so items whose detailed info columns number is not 16 or 18 are ignored. We calculate the number and accumulate a new dataframe. Finally, concat them with the origin dataframe.

Next, we move on to the table text. For the text in the general information, soil requirements, water requirements, light requirements, fertilizer requirements and product measurement sections, convert it to all lowercase and remove non-alphanumeric characters. Load the stop words list, split it into separate words and remove stop words. TF-IDF feature extraction is performed on the processed text using TF-IDF Vectorizer and the result is stored in a Pandas data frame. Then the total TF-IDF score of each term is calculated and sorted in descending order to find the most frequent terms, so that the top 50 most frequent terms are output. The top 50 words of TF-IDF (Term Frequency-inverse Document Frequency) and TF (Term Frequency) are merged and then deduplicated to obtain a list of independent words.

3.4 Knowledge Representation Using Neo4j

For the text in the environmental demand and size information section, data frames and dummy variables are created, the text is splitted into separate words and the word frequency of each word is calculated, and the top five high-frequency words are extracted to create a new column. For numeric information, the number is matched with a regularization expression and printed. We use neo4j graph to show the obtained data as shown below.

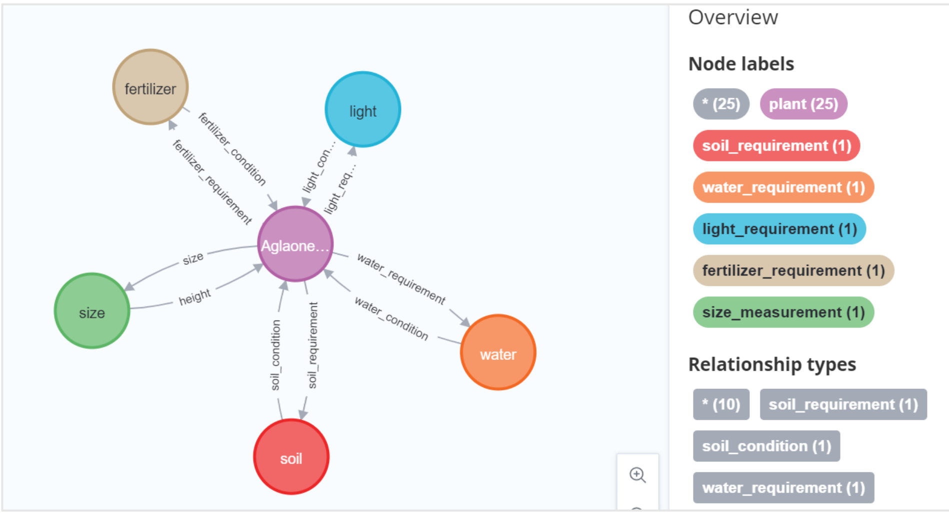

Figure 7. Sample of all plant information categories

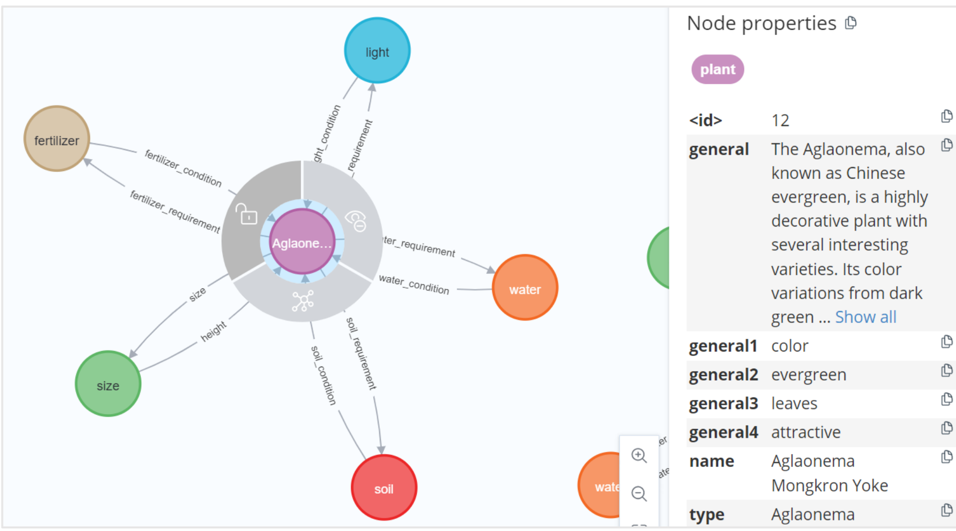

Figure 8. Sample of General information

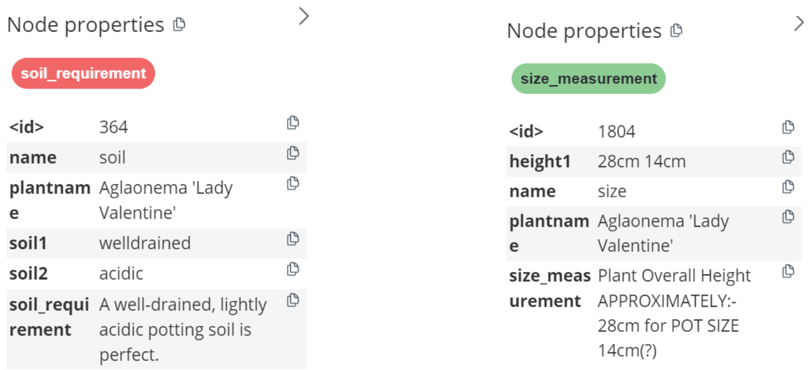

Figure 9. Soil requirement information sample (left) and product measurement sample (right)

3.5 Data Correction

Finally, based on the observation of the data set and the discussion of user requirements, we decided to divide the water requirement into occasionally and usually, the fertilizer requirement into easy and careful, for subsequent reasoning about the required care time. The light requirement is divided into three categories: direct, indirect and shade, so that users can quickly filter the plant species according to the location of the plant they want to place. The General information contains all the information that you want to show the user at the end, such as pictures of the plant, general information, and links to the purchase page. The data were all manually corrected for classification accuracy.

Chapter 4: System Design and Implementation

This chapter provides an in-depth exploration of the frontend and backend architecture, system functionality, and the underlying recommendation algorithms.

4.1 System Architecture

This section describes the overall design of our plant recommendation system, encompassing both frontend (Part I) and backend (Part II) components and how they interact to provide the desired functionality, as shown in Figure 10.

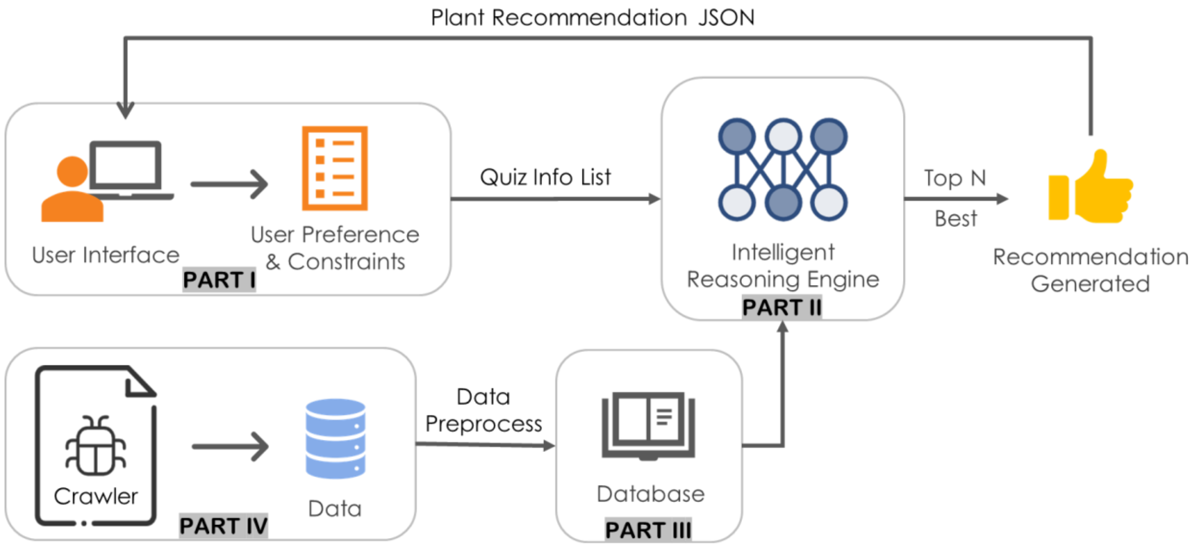

Figure 10. Overall Architecture

4.1.1 Frontend Architecture

The frontend is designed to provide an interactive and visually appealing user interface. It primarily consists of HTML, CSS, and Python Script to create web pages that users interact with in their browsers. The key components in the front-end architecture include:

1) **HTML Pages**: These serve as the structural foundation of the user interface, defining the layout and content structure of the web pages.
2) **CSS Stylesheets**: Cascading Style Sheets are used for styling HTML elements. They define the visual presentation of the web pages, including layout, colors, fonts, and other aspects.
3) **Quiz.py**: A Python script using Streamlit serves as the entry point for handling user requests. It processes user inputs, such as preferences, and communicates with the recommender component to provide personalized recommendations.
4) **Fonts**: Custom fonts are embedded in the front-end using *@font-face*. This ensures that the user interface maintains a unique and visually appealing design.

5) **Animations**: The front-end utilizes CSS animations for elements like stars and ribbons to create dynamic and engaging user experiences.

4.1.2 Backend Architecture and Data Storage

The backend is responsible for processing user requests, managing data, and ensuring the system's functionality. It consists of server-side components, including:

1) **Recommender**: The recommender component analyzes user preferences and generates recommendations. It collaborates with the data storage and retrieval components to provide relevant suggestions.
2) **Data Storage**: Data storage mechanisms like databases or file systems store information related to users, products, and recommendations. These components support data retrieval and persistence.
3) **User Preferences**: The user's preferences, including light, height, care, species, and other preferences, are passed to the system through **user_answers**. The backend processes this data to understand user requirements.

4.1.3 Interactions and Workflow

When a user accesses the system, the front-end collects their preferences and sends the data to the backend through **quiz.py**. The backend component, **recommender.py**, analyzes the user's preferences and queries the data storage to generate personalized recommendations. These recommendations are then sent back to the front-end for presentation to the user.

Overall, the system architecture ensures a seamless flow of information and interactions between the front-end and backend components, creating an engaging and personalized experience for the users.

The integration of fonts, animations, and interactive elements in the front-end enhances the user experience, while the backend handles the complexity of processing preferences and providing relevant suggestions.

This architecture ensures the efficient functioning of the system, making it capable of delivering tailored recommendations to users based on their preferences. Figure 11 is an illustration of the modules in the system and their interactions.

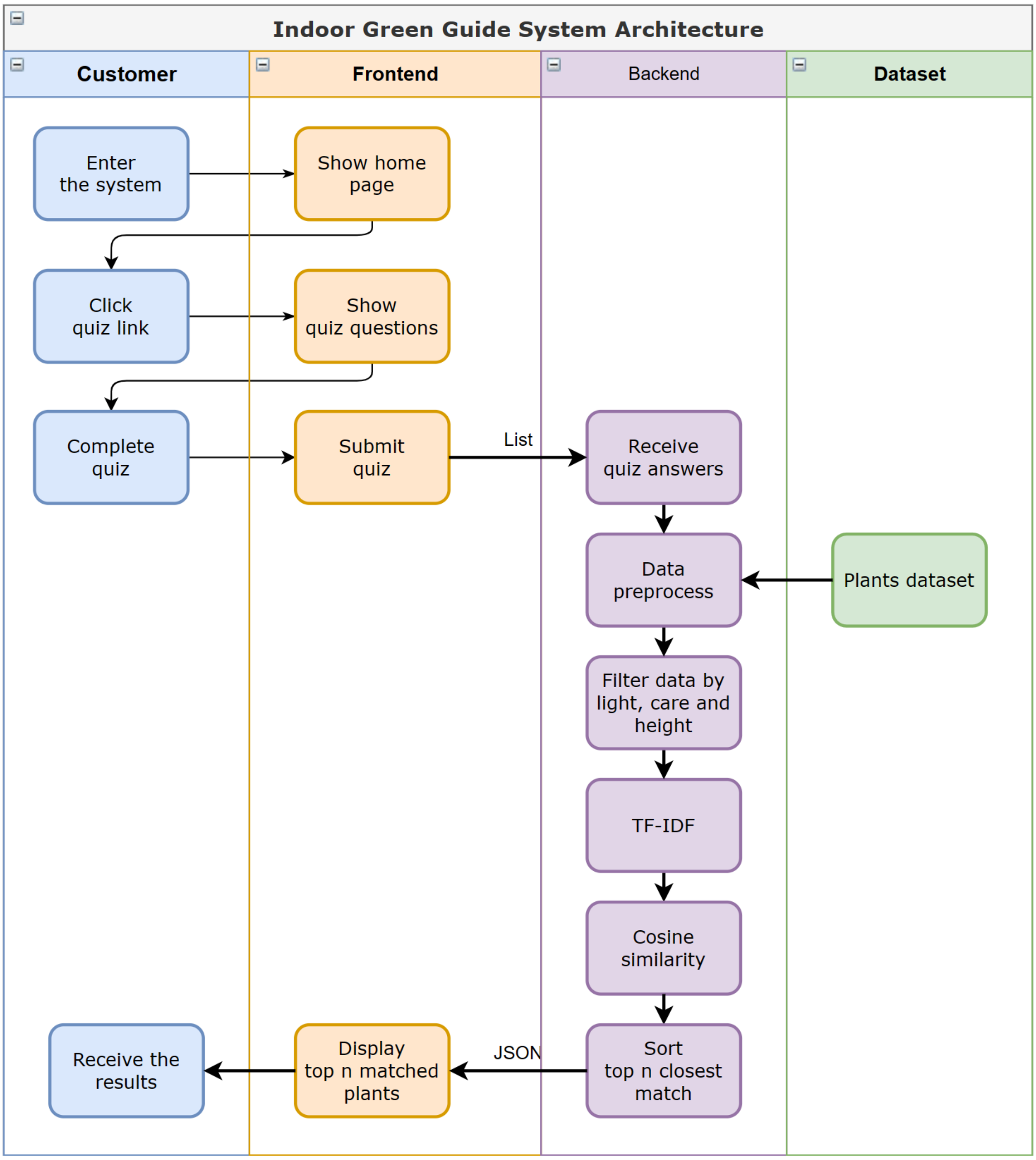

Figure 11. Different modules in the system and their interactions

4.2 Functionality and Implementation

In this section, we will outline the main functionalities and modules of the system and provide a detailed description of each module's responsibilities, interactions with other modules and how they were implemented:

4.2.1 User Interface (Frontend)

The User Interface is responsible for presenting the application to the user. It includes various components for user interaction, consists of several HTML pages and a Streamlit application. To be specific, HTML pages primarily serve to display information, while the Streamlit app is responsible for receiving user input, transmitting it to the backend, receiving output from the backend and presents it to the user.

HTML Pages

- ***Initial page***: Presents the main title, allows users to navigate to other sections of the application, and directs them to the quiz page for indoor plants recommendation.
- ***'Home' page****:* Provides the landing page of the application, introducing the system's purpose and inviting users to select indoor plants.
- ***'About' Page****:* Offers information about the project's designers.
- ***'From' Page****:* Acknowledges data sources.

Figure 12. Initial page

Streamlit Application

Streamlit is an open-source Python library for creating web applications with minimal effort. It serves as the user interface through which users interact with the system. Below are the key aspects of the frontend development:

- **Collect User Preferences**: Users are presented with a series of questions to determine their plant preferences. These questions cover aspects such as light condition, plant height, care time, plant species, and other preferences. By recording the user's answers, user preferences are collected.
- **Trigger Backend Processing**: A button labeled "Get Recommendations" is provided, allowing users to submit their preferences and initiate the backend processing for plant

recommendations.

- **Display Recommendations**: The recommendation results returned in JSON by the backend are displayed, including the plant's name, picture, URL link by which can navigate to its sales interface, and a description. Results are presented in columns using Streamlit's *st.columns* component.

Questionnaire

The questionnaire is the main part displayed on the Streamlit app. It consists of five questions that help us determine the customer's preferences on plants, as shown in the Table 1 and Figure 13 below:

Table 1. Questions and options on the page

Questions	Options
What is your preferred light condition?	Direct Sunlight/ Indirect Sunlight/ Shade
What height of plants do you prefer?	Short/ Medium/ Tall
How much time can you devote to plant care?	Low/ Medium/ High
What plant species do you prefer?	None/ Cactus/ Climber/ Creeper/ Fern/ Herbs/ Orchid/ Shrub/ Succulent/ Tree/ Tropical
Do you have other preferences?	None/ Has Flowers/ Fit for Family/ No soil/ Decorative

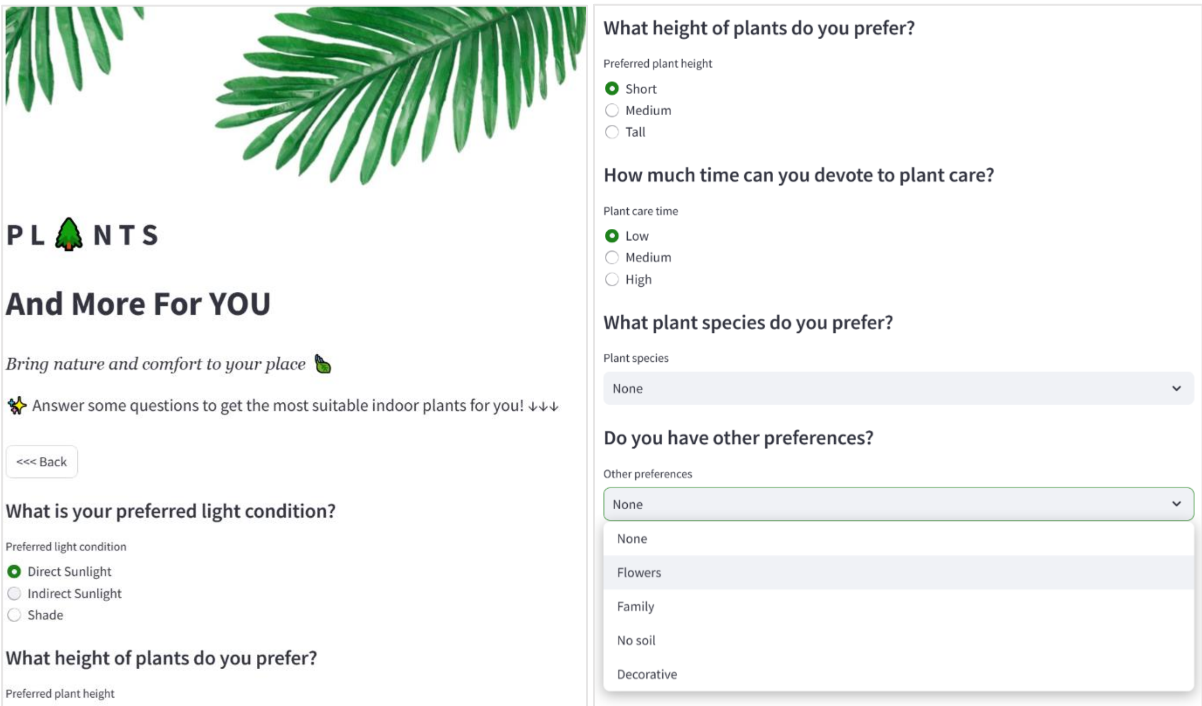

Figure 13. Questions and options displayed on the page

4.2.2 Recommendation Engine (Backend)

Serving as the core of the whole system, the Recommendation Engine generates personalized recommendations based on user preferences. It integrates with the Streamlit frontend, processes user preferences, matches them with plant data, calculates similarity scores, and generates plant recommendations based on those scores. Followed is a detailed explanation of the intelligent reasoning functionality:

1) **Data Retrieval**: The script starts by loading plant-related data from several CSV and Excel files, including information on general plant characteristics, light requirements, water needs, fertilizer preferences, measurements (height), and more. These datasets are stored in dataframes for further processing.
2) **User Preference Processing**: It processes the user's input preferences, which are collected through the Streamlit frontend. These preferences include the user's choice for light conditions, plant height, level of care, preferred species, and other special preferences. The script maps these preferences to specific criteria suitable for plant selection.
3) **Matching Plants**: Using the processed preferences, the script matches the user's choices with plant data in the loaded datasets. This involves filtering the dataset to find plants that align with the user's preferences for light, height, care, and other criteria. The matching process results in a set of plant candidates.
4) **Text Vectorization**: To perform content-based recommendation, the script uses the TF-IDF (Term Frequency-Inverse Document Frequency) vectorization technique to convert textual plant descriptions into numerical vectors. This allows for quantifying the similarity between the user's preferences and plant descriptions.
5) **Cosine Similarity Calculation**: The script computes the cosine similarity between the user's preferences (vectorized) and the vectorized plant descriptions. This similarity score indicates how well each plant matches the user's input.
6) **Recommendation Generation**: The plants are ranked based on their cosine similarity scores, and the top N (e.g., 3) plants with the highest similarity scores are selected as recommendations. For each recommended plant, the script collects essential information such as the plant's name, picture, URL, and a brief description. These recommendations are then organized into a list.
7) **Output**: Finally, the script returns the list of recommended plants in JSON format to the Streamlit frontend. These recommendations are presented to the user, including the plant's name, picture, URL, and a brief description.

4.3 Recommendation Algorithm

Our system utilizes Content-Based Filtering to recommend plants to users based on their preferences, where item (plant) similarity is determined through TF-IDF vectorization, and cosine similarity is used for the calculation. The following is an explanation of the algorithms employed in the system.

4.3.1 Content-Based Filtering

Content-Based Filtering uses item features to recommend other items similar to what the user likes, based on their previous actions or explicit feedback. In this case, the filtering bases on the characteristics of plants without recording user historical behaviour. It profiles items based on their features and recommends items similar to what the user has shown an interest in.

This algorithm is considered ideal for our system in virtue of its usability, scalability and capability of personalized recommendation. As the model doesn't need any data about other users since the recommendations are specific to this user, which makes it easier to scale to a large number of users. Besides, the content-based filtering model can capture the specific interests of a user, and can recommend niche items that very few other users are interested in.

4.3.2 TF-IDF (Term Frequency-Inverse Document Frequency)

TF-IDF is a fundamental concept in natural language processing and information retrieval. TF-IDF is a numerical statistic that reflects the importance of a term within a document relative to a collection of documents (corpus), consisting of two components:

1) **Term Frequency (TF)**: Measures how often a word appears in a document, normalized by the total number of words in the document.

$$TF = \frac{\text{number of times a term appears in a document}}{\text{total number of words in the document}} \tag{1}$$

2) **Inverse Document Frequency (IDF)**: Measures how unique or rare a term is across all documents in the corpus. It's calculated as the logarithm of the total number of documents divided by the number of documents containing the term.

$$IDF = \log\left(\frac{\text{total number of documents}}{\text{number of documents containing the term} + 1}\right) \tag{2}$$

TF-IDF is calculated by multiplying the two components for each word in a document. The higher the TF-IDF score, the more relevant the word is for the document in the context of the corpus. TF-IDF can be used to rank documents by their relevance to a query, or to extract keywords from a document.

$$TF - IDF = TF \times IDF \tag{3}$$

TF-IDF is utilized by the recommendation engine to vectorize textual data. The textual data, which includes plant descriptions, is processed and transformed into a matrix where each row corresponds to a document (plant description) and each column corresponds to a unique term. Then TF-IDF scores are calculated for each term in each document, which represents the importance of that term in the context of that document.

4.3.3 Cosine Similarity

In data analysis, cosine similarity is a measure of similarity between two non-zero vectors defined in an inner product space. Cosine similarity is the cosine of the angle between the vectors; that is, it is the dot product of the vectors divided by the product of their lengths. It follows that the cosine similarity does not depend on the magnitudes of the vectors, but only on their angle. Cosine Similarity ranges from -1 (completely dissimilar) to 1 (perfectly similar). A value of 0 indicates no similarity.

According to the recommendation engine, Cosine Similarity is employed to identify the similarity between the user's preferences (represented as a TF-IDF vector) and the plant descriptions (also represented as TF-IDF vectors). After the user's preferences are transformed into a TF-IDF vector, Cosine Similarity is calculated between this vector and the vectors representing plant descriptions. The result is a similarity score for each plant description, and the plants are ranked based on these scores.

Chapter 5: Testing and Evaluation

To ensure the effectiveness and reliability of the indoor plant recommendation system, evaluation and validation are needed. This assessment encompassed two main components: frontend and backend testing. Besides, an evaluation on the system as a whole is conducted.

5.1 Backend Evaluation

The evaluation of the backend primarily relied on a similarity scoring approach. When different user preferences were input into the system, it provided corresponding plant recommendations, each with an associated similarity score. This score quantifies how closely the recommended plant aligns with the user's preferences and serves as a measure of the recommendation quality. Here is an example of input, its corresponding output, and the associated similarity score:

Input		Output (Plant: Similarity)
Light condition:	indirect sunlight	
Plant height:	medium	Potted Assorted Orchid Plant: 0.31
Time to care:	low	Potted Assorted Mokara Orchid Plant: 0.29
Plant species:	orchid	Aloe 'Mini Belle': 0.11
Other preference:	has flowers	

5.2 Frontend Testing and Evaluation

In the testing of the system's frontend, we primarily assessed the navigation between different frontend pages, the correct collection and transmission of user input to the backend, and the proper display of results returned by the backend. According to the test results, the system has effectively achieved its fundamental functionalities. Here are some examples of test results:

Test Case ID:	Frontend-1	Test By:	LI Dongshuo
Test Description:	Test the interconnection of webpages	Test Date:	18-Oct-2023
Pre-Conditions:	Open the Indoor Green Guide home (initial) page		

Step #	Step Details	Expected Result	Actual Result	Statas
1	Click ‘Home’ button	Show ‘Home’ page	Show ‘Home’ page	Pass
2	Click ‘About’ button	Show ‘About’ page	Show ‘About’ page	Pass
3	Click ‘From’ button	Show ‘From’ page	Show ‘From’ page	Pass

4	Click 'Choose your own plants!' link	Show Questionnaire page	Show Questionnaire page	Pass

Test Case ID:	Frontend-2		Test By:	LI Dongshuo
Test Description:	Test the 'Back' function of webpages		Test Date:	18-Oct-2023
Pre-Conditions:	Open the Indoor Green Guide subpages (i.e., 'Home', 'About', 'From' and Questionnaire pages)			
Step #	Step Details	Expected Result	Actual Result	Statas
1	Click 'Back' button from 'Home' page	Show initial page	Show initial page	Pass
2	Click 'Back' button from 'About' page	Show initial page	Show initial page	Pass
3	Click 'Back' button from 'From' page	Show initial page	Show initial page	Pass
4	Click 'Back' button from Questionnaire page	Show initial page	Show initial page	Pass

Test Case ID:	Frontend-3			Test By:	WANG Jiangqianyi
Test Description:	Test the validity of the recommendation with 3 inputs			Test Date:	19-Oct-2023
Pre-Conditions:	Open the Questionnaire page				
Step #	Step Details	Test Data	Expected Result	Actual Result	Statas
1	Answer the first three questions (light condition, plant height preference, time to care)	Direct sunlight, short, low	Show top 3 matched plants	Show top 3 matched plants: Tillandsia Usneoides, Culantro Thai Parsley, Dieffenbachia Starbright	Pass
2		Indirect sunlight, short, high	Show top 3 matched plants	Show top 3 matched plants: Zebra Plant, Dieffenbachia Exotica, Crassula Ovata	Pass
3		Shade, tall, medium	Show top 3 matched plants	Show top 3 matched plants: Cordyline, Dieffenbachia Cheetah, Alocasia Cucullata 'Crinkles'	Pass

When getting recommendations, not only the general information of the suggested plants will be displayed, but a link to its source website is provided by our system, as shown in Figure 14. Through this link, users can access more detailed information (Figure 15) to assist them in making purchasing decisions.

Figure 14. Recommendations given by our system

Figure 15. The website that the first link in Figure 14 navigates to

5.3 System Evaluation

This plant recommendation system has several notable strengths and potential shortcomings.

Advantages:

1) Personalized Recommendations: The system employs content filtering algorithms to provide users with personalized indoor plant recommendations based on their preferences and requirements, which helps users find plants that align with their specific needs more easily.
2) Comprehensive Information: The system offers extensive plant information, including features, care requirements, and related images, allowing users to gain in-depth knowledge about plants they may be interested in.

3) Simplified Selection: For users with limited knowledge of plants, the system simplifies the decision-making process, providing a more streamlined way to choose suitable plants.

Shortcomings:

1) Data Completeness: Incomplete or erroneous data in the data source may affect the accuracy of recommendations. The system needs to ensure data accuracy and completeness.
2) Algorithm Limitations: The system employs a content-based recommendation algorithm primarily based on user input and plant characteristics, which means factors like user profile or the popularity of plants among other users are not considered. Introducing collaborative filtering and other algorithms could enhance recommendation diversity and accuracy.
3) User Feedback: The system appears to lack user feedback and reviews. Incorporating user feedback into the recommendation process can better cater to user needs.

Chapter 6: Conclusion and Further Work

6.1 Conclusion

Throughout the project, we conducted a comprehensive exploration of the market landscape, engaged in rigorous business analysis, and carefully evaluated user requirements. This process illuminated a distinct demand within the market for a plant recommendation system, leading us to conceive, design, and realize the Indoor Plant Recommendation System.

This system leverages content-based filtering, TF-IDF, and cosine similarity algorithms to provide personalized recommendations to users. Its primary strength lies in the ability to provide personalized recommendations, comprehensive plant information, and a simplified selection process.

6.2 Further Work

As mentioned before in section 5.3, there are areas for improvement for our system, such as data completeness, algorithm enhancements, and user feedback integration. Therefore, the system needs further improvements and it can benefit from addressing the following aspects:

1) Data Enhancement: Ensuring data completeness and accuracy in the data source is crucial to improving recommendation quality.
2) Algorithm Diversification: Incorporating collaborative filtering and other recommendation algorithms can enhance recommendation diversity and accuracy.
3) User Feedback: Collecting and incorporating user feedback and reviews into the recommendation process can better cater to user needs.
4) Graphical User Interface: Future work can focus on enhancing the system's usability, making it more intuitive and visually appealing.

Overall, this plant recommendation system is a promising tool to assist users in selecting indoor plants that suit their preferences and requirements. Further development focusing on data integrity, algorithm enhancements, user feedback integration and GUI can enhance the system's utility and user experience.

References

[1] S.J. Maisey, et al. An extended baseline examination of indoor VOCs in a city of low ambient pollution: Perth, Western Australia Atmos. Environ., 81 (2013), pp. 546-553

[2] O. Geiss, et al. The AIRMEX study – VOC measurements in public buildings and schools/kindergartens in eleven European cities: statistical analysis of the dataAtmos. Environ., 45 (2011), pp. 3676-3684

[3] N.G. Bruce, et al. The Health Effects of Indoor Air Pollution Exposure in Developing Countries World Health Organization (2002)

[4] D.E. Hu, et al. Formaldehyde in residences: long-term indoor concentrations and influencing factors Indoor Air, 20 (2010), pp. 196-203

[5] B.C. Wolverton, et al. Foliage plants for removing indoor air pollutants from energy-efficient homes Econ. Bot., 38 (1984), pp. 224-228

[6] F. Brilli, et al. Plants for Sustainable Improvement of Indoor Air Quality, Trends in Plant Science,23-6(2018), pp. 507-512

[7] G. Trend, https://trends.google.com/trends

[8] T. Economy, "Live trees and other plants | Imports and Exports | 2022," Publication date: 2023-05-14. Available: https://trendeconomy.com/data/ commodity_h2?commodity=06,TOTAL&indicator=TV,tv_wrld_share&trade_flow=Export,Import&time_period=2022

Appendix

Project Proposal

PROJECT PROPOSAL

Date of proposal: 2023.10.2
Project Title: Indoor Green Guide
Group ID (As Enrolled in Canvas Class Groups): Group 3. **Group Members (name, Student ID):** CAI Jingjing A0285758H GAO Zihui A0285749H LI Dongshuo A0285829J LI Jixuan A0285733W WANG Jiangqianyi A0285765L
Sponsor/Client: *(Company Name, Address and Contact Name, Email, if any)* Institute of Systems Science (ISS) at 25 Heng Mui Keng Terrace, Singapore NATIONAL UNIVERSITY OF SINGAPORE (NUS) Contact: Mr. GU ZHAN / Lecturer & Consultant Telephone No.: 65-6516 8021 Email: zhan.gu@nus.edu.sg
Background/Aims/Objectives: Whether in the workplace or at home, indoor air pollution can contribute to a range of health issues, including asthma, wheezing, conjunctivitis, and eczema. Additionally, elevated levels of carbon dioxide within enclosed spaces can lead to concentration problems and hinder decision-making. Indoor plants offer an effective and natural means to improve air quality. Studies have indicated that indoor plants can have several positive effects, such as boosting moods, mitigate noise levels, enhancing productivity, increasing concentration and creativity, reducing stress and fatigue, purifying indoor air by absorbing toxins, elevating humidity, and generating oxygen. The market for indoor plants has expanded quickly after the COVID-19 epidemic. Due to people spending more time indoors and the growing demand for better indoor air, there has been a large increase in the number of web searches for indoor plants after 2020 compared to the past. When people want to buy indoor plants, most of them will check the recommended plants on the browser, and then buy online, or directly go to the offline purchase and ask the merchant to help recommend. However, the lack of knowledge about plants and understanding of plant habits makes it difficult for consumers to choose when making decisions, and the advice provided by the Internet or merchants is often influenced by marketing factors.

Therefore, we hope to design a system that can recommend suitable plant species through consumer demand description.

The system will provide plant pictures, plant growth needs (light, water and fertilizer, etc.), plant size and other characteristics to help consumers screen suitable plants, allowing consumers to find the best houseplant for them.

Project Descriptions:

Project Scope

This project aims to develop an intelligent indoor plant recommendation system, named 'Indoor Green Guide,' to assist users in effortlessly making a plant purchase decision. The scope of this project includes:

- User Needs Analysis
  In the early stages of the project, we need to gather user requirements and identify the specific constraints or criteria that users wish to apply to obtain plant recommendations through the system. These may include lighting conditions, temperature, space size, plant types, and other user preferences.

- Data Collection and Preparation
  To build an intelligent recommendation system, we have to collect data related to plant growth, like the growing requirements of different plants. And we should pre-process the dataset to prepare it for information extraction and retrieval.

- Modeling and Algorithm Selection
  Select appropriate intelligent reasoning algorithms and techniques to build the recommendation system. This may involve the application of machine learning, deep learning, expert systems, etc. it is also needed to determine how to match user inputs with plant requirements and make recommendations.

- User Interface Design
  Develop a user-friendly interface that allows users to input their conditions and requirements while displaying a list of recommended plants. Interface design needs to consider user experience and usability.

- System Testing and Evaluation
  After the project is completed, conduct testing and evaluation of the system to ensure that it can provide accurate plant recommendations based on user conditions and requirements. Evaluation may include accuracy, efficiency, and user satisfaction metrics.

System Design (Main Functionalities)

Our system will offer a range of powerful and intuitive features designed to meet users' needs. The following are the main functionalities of the system:

- **Collect User Preferences**: Users will be presented with a series of questions to determine their plant preferences. These questions cover aspects such as light condition, plant height, effort to care, plant species, and so on. The system is expected to record the user’s answers to later recommendation.

- **Recommendation Generation**: Intelligent reasoning algorithms and techniques will be used to retrieve information from the dataset and decide which plants can fit in the user’s preference.

- **Display Recommendations**: The recommendation results should be displayed in a user-friendly way, including the plant's name, picture, and other necessary information.

Mapped System Functionalities Against Courses Knowledge

Course Name	System Functionalities Against Knowledge/Techniques Applied
Machine Reasoning (MR)	**Knowledge Elicitation and extraction**: Web crawling from websites & repositories, manual extraction from websites/internet **Text Pre-Processing**: Used for data pre-processing, involving cleaning, tokenization, removing stop words, stemming (reducing words to their base form), etc **Knowledge Representation**: Plant data representation for Neo4j database (section 3.4), graphical representation for system architecture (Figure 10), process flowchart for each module (Figure 11) **Rule Based System**: The plant recommendation system incorporates rules. Upon receiving user preferences, the backend initially filters based on keywords related to light conditions, plant height, and care time required. This process is guided by predefined rules.
Reasoning Systems (RS)	**Cosine Similarity**: Measuring the similarity between the vectorized user input (keywords) and the vectorized plant descriptions
Cognitive Systems (CGS)	**Cognitive Knowledge Representation and Reasoning**: Understanding user preferences and plant characteristics, and reasoning to make personalized recommendations. **Natural Language Processing**: Interpreting user input, which includes preferences for light conditions, plant height, and more

Installation and User Guide

Installation

To open the project, you need to complete the following operations:

1) Having an appropriate web browser, recommend Google Chrome
2) Clone or download project source code from GitHub:
   https://github.com/AiyanaCAI/IRS-PM-2023-10-03-IS05FT-GRP8-IndoorGreenGuide
3) Environment: Python or Anaconda (recommend), make sure you have installed all required libraries before running the project.
   - ✓ streamlit
   - ✓ numpy
   - ✓ pandas
   - ✓ scikit-learn
   - ✓ matpltlib
   - ✓ nltk
   - ✓ json
   - ✓ pillow

   If you don’t have any of the above libraries, pip install it in your project environment.
4) Install Streamlit. For Streamlit install details, please refer to the link:
   https://docs.streamlit.io/library/get-started/installation
5) After completing all install steps, open a terminal in your environment, input
   streamlit run homepage.py to run our website.

Be sure to change the address commented in the code to the address of your project, especially don't forget to change the local URL.

```
<h2 class="titles">
    <a href='http://localhost:8501'>
```

Then run the page1.html, you can enter our website link.

User Guide

1. Go to the home page of our website, you can see the page shown below.

Figure 16. First page of the website

Click "Choose your own plants!" to navigate to the options page.

2. When you enter the options page, you will see the following screen.

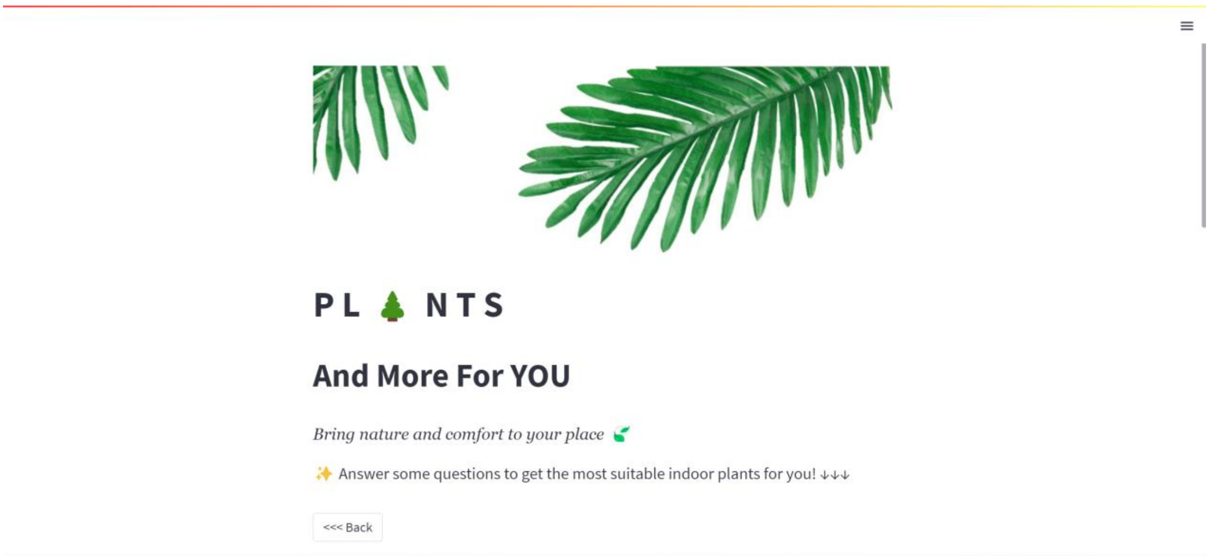

Figure 17. Questionnaire page

Scroll down and you'll see five questions, the first three are required and the last two are optional, as shown below.

The first optional question is the species of indoor plant, and the second one is your specific requirements for the plant. You can either drop down the form to select it or leave it unselected; the default value is none.

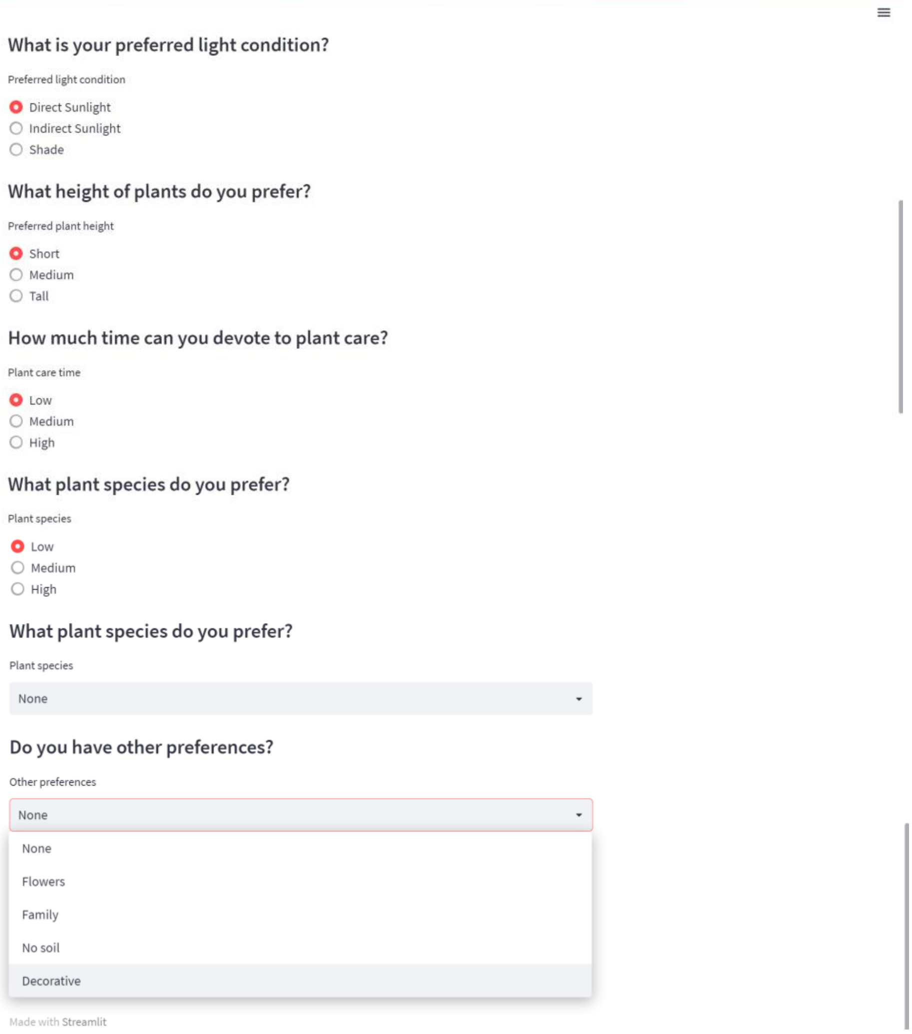

Figure 18. Five questions

After selected the answers to the five questions according to your own indoor plant needs, click the "Get Recommendations" button below to submit your preferences and get your personalized recommendation results.

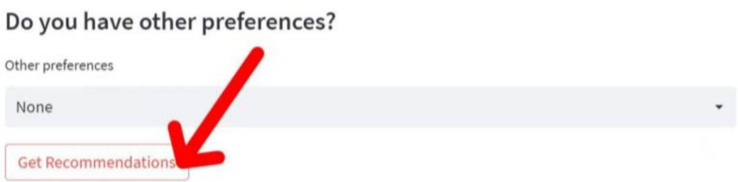

Figure 19. Submit your preferences

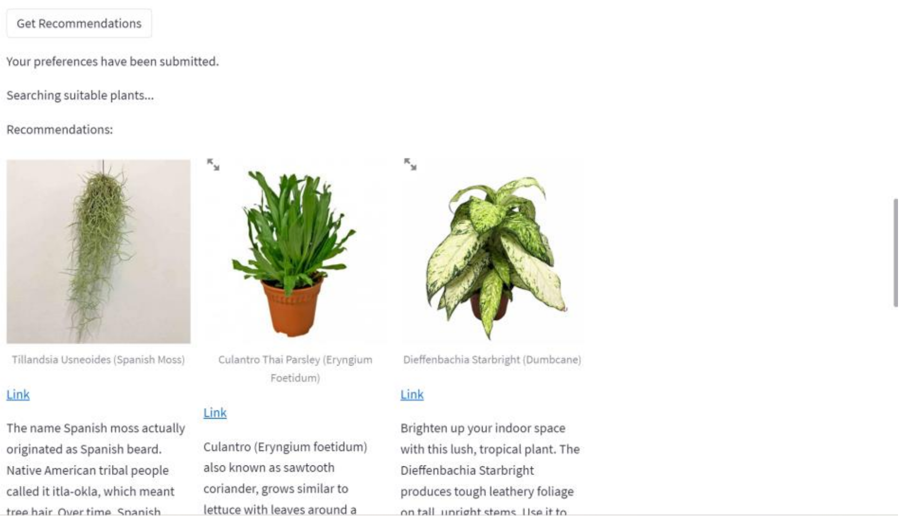

Figure 20. Recommendation result

After getting the recommendation results, you can see the picture and basic information of the plant, click the blue “Link” to jump to the purchase page of the plant, which has more detailed information and price reference.

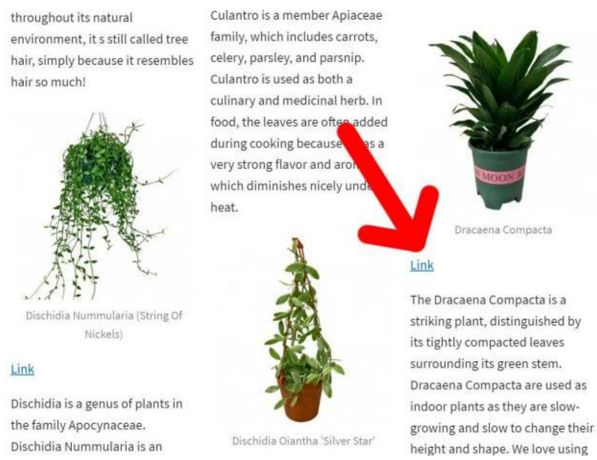

Figure 21. Jump to plant details and purchase page

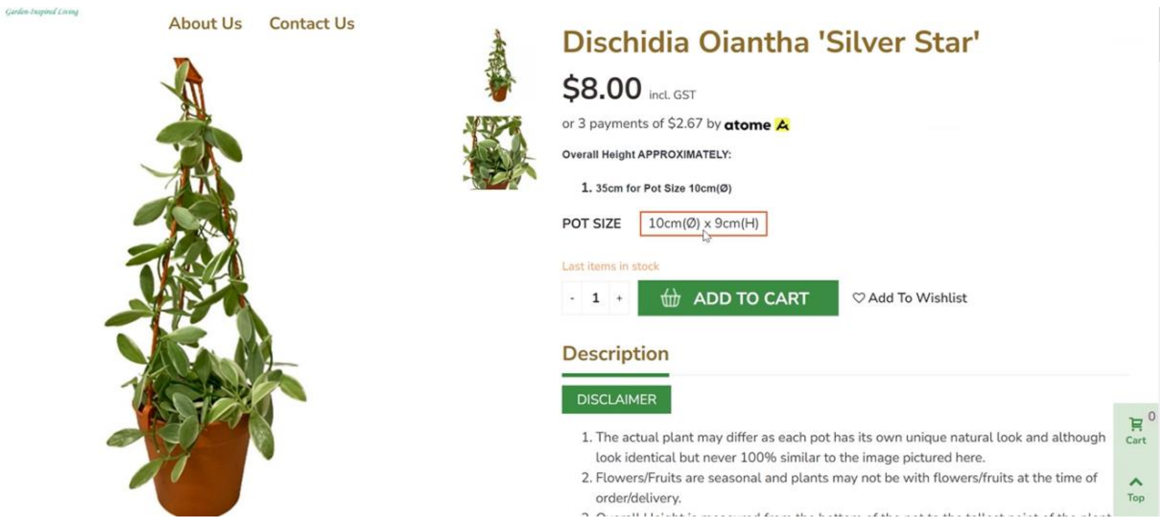

Figure 22. Plant detail and purchase page sample

Individual Project Report

Cai Jingjing's Individual Project Report

Individual Project Report

Your Name:	CAI JINGJING
Certificate:	Graduate Certificate in Intelligent Reasoning Systems

1. Your personal contribution to the project.

Organize group meetings and discussions as group leader.

Propose and determine topic selection, first presentation.

Data process and correct, files edit and clear up.

Write report, write script and make videos.

2. What you have learnt from the project.

I learned a lot in learning and problem solving. The group discussion allowed me to learn a lot from the other group members.

At the same time, it was my first contact with front-end knowledge. I also learned a lot of new techniques about data processing. Building the system from scratch gave me a clear understanding of how what I learned in the classroom combined with practical projects.

And this project experience made me light up the new skill of video editing. It has been both enjoyable and promises to be a valuable asset for my future work.

3. How you can apply this in future work-related projects.

This experience has sharpened my critical thinking and problem-solving skills, which I can apply to tackle challenges and devise innovative solutions in future work projects. Additionally, group discussions have enhanced my collaboration and communication skills, enabling me to work effectively in teams and promote a more productive work environment.

The knowledge, skills, and experiences I've gained will form a solid foundation for my future work. They will empower me to navigate challenges, collaborate effectively, leverage technical expertise, analyze data, bridge theory and practice, and communicate ideas more effectively, ultimately enhancing the quality and success of my future endeavors.

Wang Jiangqianyi's Individual Project Report

Individual Project Report

Your Name:	WANG Jiangqianyi
Certificate:	Graduate Certificate in Intelligent Reasoning Systems – IRS project

1. Your personal contribution to the project.

1) Project Ideation: I actively engaged in the project starting phase, contributing to discussions and the project topic selection. This entailed determining the requirement for a user-friendly intelligent plant suggestion system.
2) Data Preprocessing: To transform and prepare the data to create a dataset suitable for the system, I played an essential role in data preprocessing, which involved cleaning and structuring data to make it usable for the recommendation engine.
3) Model and algorithm selection: I actively participated in the process of evaluating various algorithms and models suitable for our Intelligent Plant Recommendation System. Different recommendation algorithms and techniques were researched and assessed to determine the best fit for our project's objectives during the process.
4) Frontend Development with Streamlit: I was responsible for designing and developing the frontend of the system using Streamlit, an open-source Python library. This involved creating a user-friendly interface for users to input their plant preferences and trigger the interaction with backend.
5) Report Writing: I actively participated in the documentation of the project. I wrote the project report's System Design and Implementation, Testing and Evaluation, Conclusion and Further Work. Besides, I organised the entire report to ensure that it provided a clear and comprehensive overview of the system.

2. What you have learnt from the project.

This project provided me with invaluable learning experiences. To start with, I gained hands-on experience in data preprocessing, including data cleaning and structuring. This ability is essential for processing unprocessed data for recommendation and machine learning systems. In addition, as I evaluated and contrasted several algorithms to see which ones would work best for our system, I gained a better grasp of the models and their advantages and disadvantages. The project also improved my knowledge of frontend programming using Streamlit and taught me how to efficiently build interactive web applications for data science projects. This experience will come in handy for projects in the future where user interfaces are crucial.

Together with my expertise being reinforced, I also gained insight into the significance of project management and productive cooperation. Collaborating with team members and coordinating tasks helped me understand how to work efficiently on a complex project.

3. How you can apply this in future work-related projects.

The knowledge and skills gained from this project will be invaluable in future work-related projects. Expertise in data preprocessing, creating user-friendly interfaces, and developing content-based filtering and recommendation systems is essential for projects that need accurate analysis, machine learning, giving user-specific suggestions and so on. Furthermore, the ability to assess algorithm performance is essential for achieving goals and adding value for project participants.

Plus, project efficiency may be ensured by using good work delegation and cooperation abilities to every project requiring numerous people.

Li Dongshuo's Individual Project Report

Individual Project Report

Your Name:	Li Dongshuo
Certificate:	Intelligent Reasoning Systems

1. Your personal contribution to the project.

1. Contributed to the initial ideation and selection of the idea
2. Database screening and sorting
3. Front-end page design and production
4. Participated in the production of the project video introduction

2. What you have learnt from the project.

1. Knowledge and techniques

Participated in the entire project production process, I have a deep understanding about the knowledge we learnt at the classes and during the actual operation, I learnt more details about building a system.

During the implementation of the system, I learnt how to process the original data set to a useful one. Also learnt how to deal with the recommendation system based on similarity.

2. Tools

It deepened my understanding and application of python.

Also the first contact with the front-end page design. Have a basic learning and understanding about the html and CSS.

3. How you can apply this in future work-related projects.

It was my first time to do such relatively complete system and I was really confused at first. But after we truly worked on it step by step, it's not a formerly-tricky task.

In the future work-related project, I will be more familiar and confidence to the work. And having those experience, I can save a lot of unnecessary time wasted. And I also contact the front-end, I think I can use it to better exhibition our project.

Li Jixuan's Individual Project Report

Individual Project Report

Your Name:	LI JIXUAN
Certificate:	Graduate Certificate in Intelligent Reasoning Systems

1. Your personal contribution to the project.

(1) Idea Discussion: As an indoor plants lover, understanding the potential benefits of such a system, I proposed its development as a solution to enhance the user experience and provide user-need indoor plants recommendations even when they do not really know about specific type of indoor plants.

(2) Data crawling: We group search on the internet for indoor plants website and filter them which is suitable for Singapore and data is abundant for us to crawl. We just find one website called https://www.candy.com which is specific for Singapore and it has many types of indoor plants and specific description. We use python to After crawling the content from the website. We just get a csv file.

(3) Data preprocessing: We extract the specific data column from the described content of each indoor plants type. And we clean the incomplete data from the whole data. Separated them into soil, water, measurement columns which is easier to analysis.

(4) Training model discuss: By jointly exploring and experimenting with different approaches, such as Bert pre-trained model and cosine-similarity methods. We finally decided to take cosine-similarity method to compute the similarity.

2. What you have learnt from the project.

(1) Crawling data

Learn the approach to crawl data from website. In this project, we use scrapy to crawl data. We define crawling model, and then, we set the rule of crawling. After requesting, we parse the response from F12 and get the crawling data by xml data response.

(2) Text preprocessing

After crawling the data, I learned how to split the whole description text into several specific data column using the text preprocessing methods such as stopword removal, stemming etc. I learned how to separate data into specific column.

(3) The approach to Compare training model

I learned the approach to compare training models. We should select a similarity measurement method, such as cosine similarity, Euclidean distance, edit distance, Jaccard coefficient, etc. Different methods are suitable for different types of data, so the selection should be based on the specific circumstances.

3. How you can apply this in future work-related projects.

I learned about the data crawling method so I can use the almost general code to crawl data from the website next time in work-related projects. And I learned many text preprocessing methods which can be used in other data preprocessing model. And after the discuss with my team member, I learned the pros and cons of the similarity training models and know how to select them in different environments. And I learned about the training process of this recommended systems.

Gao Zihui's Individual Project Report

Individual Project Report

Your Name:	Gao Zihui
Certificate:	Graduate Certificate in Intelligent Reasoning Systems

1. Your personal contribution to the project.

In this project, my team members and I discussed the selection and preprocessing of the dataset. We ultimately chose three mandatory criteria for recommending potted plants, which include lighting conditions, cultivation difficulty, and the size of the potted plants. Additionally, we included multiple optional criteria to refine user preferences. I then handled the backend processing, categorizing and processing the data transmitted to the frontend. Finally, I used cosine similarity for text matching to provide personalized recommendations for users.

2. What you have learnt from the project.

In this project, I learned a lot about machine learning-related knowledge. First and foremost, data preprocessing was a crucial part of model building, and I believe it is one of the most important aspects. In our project, the dataset we obtained from web scraping was quite messy, so it was essential to devise a strategy and methods for extracting key information. Only after properly handling the data could we proceed with further model construction.

Since our goal was to build a recommendation system, text matching played a pivotal role. Here, I delved deeper into and applied NLP similarity matching techniques and cosine similarity matching for personalized recommendations. While writing backend code, I also acquired some knowledge about frontend development and data transmission between the frontend and backend, which has been immensely beneficial to me.

3. How you can apply this in future work-related projects.

As an intern aspiring to work in the field of artificial intelligence, the experience gained from this project will be highly beneficial for my future internships. Firstly, data processing, which plays a significant role in model development, is a key takeaway. Secondly, the ability to select and construct models based on real-world problems is another crucial skill I've acquired. Lastly, the project has equipped me with the knowledge to perform final processing and optimization in line with project objectives.

Equipped with the knowledge that I have acquired through the lessons and the group project, I will be able to better scope out potential intelligent system projects using the framework taught in the earlier classes, create a more robust and scalable system architecture and implement good coding practices.