**BSc (Hons) in Computing**

**Level – 6**

Text, application

Description automatically generated[](http://apiit.edu.my/)

**Mid-Point Report**

**Module Title: Final Year Project**

**Prepared by:**

**Shifny Ahmed CB007753**

**[HF2161SEENG]**

**Date of Submission: 29th October 2021**

**Instructor: Prof. Priyantha Kumarawadu**

**Word Count: 4747**

**Acknowledgement**

I would like to thank Professor Priyantha for providing all the advice and teaching required to do my FYP and I would like to thank my colleagues for helping me on any mistakes I make regarding my FYP. And finally, I would like to thank Staffordshire University for providing access for most of the online resources required for my research need.

Table of Contents

[**01.** Introduction 4](#_Toc86957096)

[**01.1.** Project Aim, Objective, and Scope 5](#_Toc86957097)

[**01.2.** Project Plan 6](#_Toc86957098)

[**02.** Literature Review 7](#_Toc86957099)

[**03.** Requirement Analysis 11](#_Toc86957100)

[**03.1.** User Requirements 11](#_Toc86957101)

[**03.2.** Functionality 11](#_Toc86957102)

[**03.3.** Functional Requirements 11](#_Toc86957103)

[**03.4.** Non-Functional Requirements 12](#_Toc86957104)

[**04.** Methodology 13](#_Toc86957105)

[**04.1.** Research and Application of Methods 13](#_Toc86957106)

[**04.2.** Technology Selection 13](#_Toc86957107)

[**04.3.** Development Platform 14](#_Toc86957108)

[**05.** Solution Concept 15](#_Toc86957109)

[**06.** System Design 16](#_Toc86957110)

[**06.1.** High-level use case 16](#_Toc86957111)

[**06.2.** Activity Diagram 24](#_Toc86957112)

[**06.3.** Sequence Diagram 30](#_Toc86957113)

[**07.** Reference 36](#_Toc86957114)

# Introduction

In this modern world, people often get used to being served at their desired place, which is mostly at home. These people expect service providers such as doctors, nurses, barbers, plumbers, electricians, carpenters, car washers, cleaners, etc. to visit their home and offer their services, and most of those people have already been hiring these types of service providers to come to their home. This hiring is being done in a manual way by asking for suggestions of well-known service providers to people for the work they want done, and not every individual service provider in the above-mentioned category will be available all the time and the manual process of hiring will cost more time and effort.

In order to automate this process, we will be building a system that will allow people to register through the application as customers and hire their desired service provider for the service they require. This system will also recommend service providers to the customer based on the city customer lives and based on the work of any selected service provider, the first release of the system will be on website which will have several functionalities for the customers in a user-friendly manner.

## Project Aim, Objective, and Scope

The main goal of this project is to connect customers with local professional service providers. This can be done by creating a recommendation system that will recommend the best service providers in their city as well as service providers based on their preferences.

We will implement a web application that will allow customers all over Sri Lanka to register and hire the best service provider in their city. Service providers in each category will be recommended to the customer based on the city and the customer's interests. The project's final product will be the first version of a web application that allows customers to register and hire only plumbers in their city, and several plumbers will be recommended based on their selection.

In order to start the project few areas have to be researched to understand the basics such as:

* **User Profiling** - Analyzing the characteristics of users’ interest is a crucial part in developing a recommendation system for home services*.*
* **Recommendation system using machine learning -** As machine learning has shown great success in predicting information based on the past information provided, this can be used to recommend a service provider to the customer based on service provider rating.
* **Collaborative Filtering** - This is a filtering strategy used to recommend a user’s preference to another similar user.
* **Content-Based Filtering** - This is the normal strategy used in most of the recommendation system which recommends the user based on the past interaction and preference of the user.
* **Python and Machine Learning** - Python is among the first programming languages to gain machine learning support in the form of libraries and tools. This can be researched for further details to learn the language.
* **Similar Mobile based Service providing System** - This can be researched to learn what functionalities are most popular among them and how much time they take to get implemented.

After the research and efficient information has been collected, we will be moving onto the implementation where we will be applying the dos and don’ts which we had identified. Dos and don’ts are listed below:

* The payment gate way won’t be implemented in the first version of this application.
* The system builder will need to implement the application in a way where only electrician needs to be available in the service providing section in the first version of the application.
* The system builder will need to implement the application in a way where customer will be able to see the personal information such as name, age, address, and image, of the hired service provider.
* The system will not have any implementation where the customer can call the hired service provider through app and vice versa.
* The system will not have any implementation where the customer can track the live location of the hired service provider and vice versa.

## Project Plan

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Tasks | Duration | Aug | | | | Sep | | | | Oct | | | | Nov | | | | Dec | | | | Jan and Feb | | | | |
| 1 | PPF | 5days |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | PSR | 3weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Project Research | 5Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | MPR | 2Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Additional Research | 5Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Developing Frontend Prototype | 3Week |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Prototype Testing | 5days |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 | Developing Frontend | 5Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 9 | Developing Backend | 8Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 10 | Connect Frontend and Backend | 3Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 11 | Complete Documentation | 2Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 12 | System Testing | 2Week |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 13 | Complete the project | 3Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 14 | Final Submission | 3Weeks |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# Literature Review

**Introduction**

People in today's environment are accustomed to being served at their preferred location, which is usually their house. These people expect service providers such as doctors, nurses, barbers, plumbers, electricians, carpenters, car washers, cleaners, and other professionals to come to their homes and offer their services, and the majority of them have already done so. This hiring is done manually by asking known people for service providers to be suggested for the work they need done, and not every service provider in the above-mentioned category will be available at all times. In this instance, the customer may have to wait for their turn to be served. To solve this, we came up with a system which is built using machine learning to offer a Service Provider Recommendation System which will list down the available service providers and will recommend the best suiting service providers according to each user’s interest.

To improve the efficiency of this home service recommendation system, it is critical to manage data generated on a daily basis in the home service application correctly, and to do so, systems must execute data collection, processing, analysis, and distribution(Garcia, et al., 2020)**.** Proper data management can boost productivity by allowing you to hire more service providers and earn more trust in the system. Over the year the amount of data generated by global sources of data such as Web sites, social media, smartphone apps, news networks, weather, political institutes, society, and the economy has been steadily increasing. Without appropriate preparation and processing, no matter how large the data is, it may be useless(Ribeiro, et al., 2015). In the home service application, best service providers over the years can be recommended to all the customers whenever the application is accessed, by collecting various customer rating on each service provider to arrange each service provider by ascending order in terms of overall rating and recommend them to the customer or the service providers may be recommended based on individual customer preference which will be collected based on the customer’s past activity. To make such recommendation, Machine learning is often used.

**Machine Learning and Recommendation Techniques**

Machine Learning is a set of statistical methods for creating mathematical models that can draw conclusions from data samples (known as a training set) (Ribeiro, et al., 2015). It is among the most rapidly developing areas of computer science (Alpaydin, 2020). For scientific modeling, consumer behavior, energy consumption predictions, related article recommendation, and user trends, a variety of machine learning algorithms have been applied to extract useful knowledge from data(Ribeiro, et al., 2015).

There are many machine learning algorithms such as Association Rules, Unsupervised learning, Supervised Learning, and Reinforcement learning. Association rules is often used in supermarket basket analysis, this is the process of defining associations between things purchased by customers. For example: Consumers who buy milk are likely to buy Choco powder as well, and buyers who buy milk, but not Choco powder are potential Choco powder customers(Alpaydin, 2020). Unsupervised learning is a basic idea in which we don't have a supervisor to offer accurate values and only have input data to detect regularities in(Alpaydin, 2020), this is mostly used for unlabeled datasets. Supervised learning is vice versa explanation of unsupervised learning, we will be doing the machine learning with datasets which are labeled, few examples of the algorithms used are: Classification, Linear Regression, Naïve bayes and etc. Reinforcement learning is where the output of the system will be sequence of action which will be used to predict the future actions. One place where this can be used is gaming. Chess, for example, has a modest number of rules but is extremely complex due to the enormous number of possible moves at each phase and the overall number of moves throughout the game and good algorithm can be used which can learn to play games well by learning all the possible sequence of action(Alpaydin, 2020).

Techniques Used to build a recommendation system are many, 3 of those will be discussed below:

* **Content based Filtering**

Based on the user's prior activities or input, content-based filtering recommends things that are similar to what they like. Some user-related characteristics may be offered expressly by the user. Other aspects, such as the programs they've already installed, may be implied.

This is one of the most effective recommendation techniques for new projects with little or no user data.

* **Implementation Process**

There are two methods that may be applied in this situation. To begin, customers can be provided a list of attributes from which they can select the ones that they most identify with. Second, the algorithm may remember which goods the user has previously selected and add those characteristics to the user's data(Shoval, et al., 2008).

* **Advantage**

Due to the little quantity of data, this approach is easily scalable. Furthermore, as it’s not required to compare data with other users, unlike other models, it may provide specialized findings tailored to the present user(Shoval, et al., 2008).

* **Disadvantage**

This model demands a significant level of domain expertise on the part of those tasked with attributing features to products. As a result, its accuracy is highly dependent on the accuracy of that knowledge. Furthermore, content-based filtering is heavily reliant on previously identified user preferences. As a result, it is restricted in that it cannot expand on previously identified user interests(Shoval, et al., 2008).

* **Collaborative Filtering**

Collaborative filtering is a method to extract items that a user might like based on the actions of other users. It works by scanning through a huge group of users to discover a smaller group of users with similar preferences to a specific user. It analyzes their favorite products and creates a ranked list of recommendations(Ekstrand, et al., 2010)**.**

* **Implementation Process**

The initial step is to look for users or objects that are comparable. The second phase is to forecast user ratings for things that have not yet been rated.

* **Demographic Filtering**

Users are classified into demographic groups based on their personal characteristics. The input data for the recommendation process comes from these classes. The goal of this technique is to identify groups of people who enjoy a particular product rather than a single brand or service(Ryngksai & Chameikho, 2014).

* **Implementation Process**

If people in class B enjoy product A and there is a person b in class B who has not yet seen product A, this product can be recommended to person b(Ryngksai & Chameikho, 2014).

Any recommendation system's main idea is to recommend items to the potential customer by predicting their utility based on previous user actions or similar user actions. It's possible that the user profile will be utilized to filter the user's preferred content. Basically, Content-Based-Filtering predicts and recommends contents, including unseen ones, to new incoming users based on the preferences of the current user. These items may be comparable to previous rated items or contain keywords that are similar(Paireekreng, 2013). In the Collaborative Filtering technique, on the other hand, the preferences of other users must be considered for the user who is now using the recommendation system (Xu, et al., 2010). Similar users can be identified in collaborative filtering using their frequent preferences and in future, preference of one user can be recommended to other similar users. But Meteren and Someren suggested to construct a recommendation system based on content-based-filtering and collaborative filtering jointly to make the system more effective. They also discovered that the precision ratios for prediction for different topics in the same document could differ(Meteren & Someren, 2000).

**User Profile in Recommendation System**

User profile is an important part of the recommendation system since it allows users to get personalized content and be recommended items that are relevant to them (Paireekreng, 2013). Wagner was the first to develop the concept of a user profile (Wagner, et al., 2002). The study proposed a framework for sophisticated mobile service personalization based on a profiling strategy that makes use of the semantic enrichment service. User modeling can assist in the creation of a user profile based on demographic factors in order to determine a user's characteristics. This can give a group of people with content that is relevant for their requirements(Paireekreng, 2013). The early-stage difficulty in the recommendation system can be addressed by the user profile, which tackles the issue of a user's lack of information.

**Text Vectorization**

Text Vectorization is a process that involves converting text into numerical form. There are several ways to perform text vectorization.

* **TF-IDF**

The TF-IDF algorithm compares the relative frequency of words in a single document to the inverse proportion of that word across the entire document corpus. This determines the importance of a provided word in a given document (Salton & Buckley, 1988).

The TF-IDF is calculated as follows: Given a document collection *D*, a word *w*, and a single document *d є D*, we calculate the TF-IDF values.

Logo

Description automatically generated

Figure 1: TF-IDF-Formulae

where *f* *w, d* is the number of times w appears in *d*, |*D*| is the corpus size, and *f**w*, *D* is the number of documents where w appears in *D* (Salton & Buckley, 1988).

* **Count Vectorizer**

The scikit-learn Python library provides Count Vectorizer. It is used to convert a text into a vector based on the frequency of each word that appears throughout the text. The performance of Count Vectorizer increases when used on short words which is mostly the case in our project (Geeksforgeeks, 2020).

We're looking for similarities between service providers based on the "work" and "city" columns, which contain sentences of no more than 2-4 words. The TF-IDF algorithm tends to downplay the significance of words that are more common throughout the corpus. Because our work and city columns have a limited number of words, each word is equally important. As a result, using the "Count Vectorizer" for this project would be appropriate.

**Similarity Measuring**

We may need to determine how similar a product is to one that the customer likes or is interested in before we can recommend it to them. This is where similarity metrics come into play. Anything can be used to determine how similar two products are. For example: the color, taste, or size of a fruit, can be used to determine its similarity.

In machine learning, similarity scores range from 0 to 1, with 1 indicating the most similar product (100 percent similar) and 0 indicating the least similar product (0 percent similar).

* **Cosine Similarity**

Cosine similarity is a metric for determining how similar products are regardless of size. It calculates the cosine of the angle formed by two vectors projected in three dimensions. This is also one of the most widely used approaches in machine learning for determining similarity. (Ye, 2011).

**Conclusion**

As machine learning has been found to be the only approach to building a recommendation system, different machine learning algorithms and filtering techniques have been found.

In this review, a home service recommendation system was proposed with all the relevant information required to implement and test it. The problem of the home service recommendation system in the early stages of recommendation can be considered based on content-based-filtering. And when the time processes, both collaborative and content-based filtering can be used jointly to make the system more effective, as suggested by Meteren and Someren (Meteren & Someren, 2000).

Content-based filtering can be accomplished in the first version by carrying out a few steps of the machine learning approach to any recommendation system. The training dataset needs to be preprocessed to remove any non-ASCII words and stop words, convert the words into lower case and attach them to the dataset as a new column. The words in the cleaned column can be converted into a vector based on the frequency of each word that appears throughout the column using the Count Vectorizer. Finally, to measure the similarity between the words, cosine similarity, which is one of the most widely used similarity measures in machine learning, can be used to find the similarity scores of each word in the column to each other. And to recommend any service provider to the customer, we can implement a function that takes into account the factor on which the recommendation needs to be made and recommend similar service providers to the customers.

The recommendation system can also utilize the benefit of user profiling, which can provide the user with personalized content and recommend relevant items to them. There is indeed a shift away from recommendations based on other users' ratings and toward personalized recommendations based on user preferences when the user shares more information related to the home services. Users can also manage their own preferences and profiles in order to receive customized content based on their user profile and how they manage their content choices.

# Requirement Analysis

## User Requirements

* **Customers**: These are public who will be able to register through our application to hire home service providers whenever required after the subscription has been made.

## Functionality

* The system should be able to keep track of all the service hired by the customer in a database with all the relevant data including id of hired service provider.
* Users must be able to sign in and sign up

## Functional Requirements

Functional requirement is a software requirement which describes a functionality that a software component or system must be capable of accomplishing(Chung, et al., 2000). This section will be addressing all the functions that the system can perform.

* User must be able to register themselves in the application.
* A user authorization method must be included in the system, in which users must identify themselves using a username and password. Only users with this level of authorization have access to the system's data.
* The user must be able to log out of the application whenever wanted.
* The system should be able to keep track of all the service hired by the customer in a database with all the relevant data including the id of hired service provider.
* The system should be able to keep track of user interest based on the service providers he/she has hired.
* System must be able to provide recommendation based on the city user lives.
* System must be able to provide recommendation based on most frequently hired service in the user’s hiring history.
* System must recommend plumbers based on any plumber the user selects.
* User must be able to view their profile details
* User must be able to edit their profile details.
* User must be able to view currently hired plumber
* User must be able to view currently hired plumber details
* User must be able to update currently hired plumber as arrived
* User must be able to update hiring as work completed
* User must be able to view hired history of user
* User must be able to view details of individual hired history of user
* User must be able to add review on each hiring
* User must be able to view their review on each hiring
* User must be able to view all the customer review on all plumbers
* User should be able to hire a service provider by clicking the “Hire Now” button.
* System must be able to send notification on important information of the user.

## Non-Functional Requirements

In software system engineering, non-functional requirements are software requirements that specify how the software will accomplish something rather than explaining what it will do(Chung, et al., 2000).

* **Performance**
* The application will need to load the login page to the user within 3-5 sec of opening
* The application will need to load the home page to the user within 1-3 sec of logging in
* Each request made should be processed within 2-4 seconds.
* **Usability**
* Customer must be able to view the general details of the service providers.
* The Application UI needs to be engaging.
* The system home page needs to be manipulative to attract new customers
* Functionalities of the user must be handled and processed in a user-friendly manner.
* **Security**
* User password must be encrypted using strong hashing.

# Methodology

## Research and Application of Methods

As our system is about recommending a service provider to the customers, this application can be named as a recommendation system which basically is built using machine learning algorithms.

Here, we have decided to use a manually made plumber dataset which consist details of various Sri Lankan plumbers. We have decided to apply content-based filtering technique to overcome the starter problem.

## Technology Selection

* Python and flask will be used to build this Home Service Recommender System using machine learning algorithms such as content-based filtering.
* As this is a web development project, no mobile development frameworks or libraries will be used.
* Content based Filtering will be used in this system to recommend service providers to the customers.

## Development Platform

The final outcome of this project will be on a web application, so a development language, framework, and IDE will be needed to develop this application.

**Development Language**

Python is used as the development language as it plays important role in almost all the development such as gaming application, desktop applications, web applications, mobile applications and etc. It is also considered one of the easiest and high demand development language by most of the software developers.

Version: Python 3.9

For the front end of the application, HTML5 with Bootstrap has been used. To make the front end presentable few CSS and java scripts has also been used. To run python code within html, Jinja2 has been used.

**Framework**

Python is one of the easiest and most powerful development languages, and there are a variety of frameworks available for it.

Django and Flask are the two most popular web frameworks in Python, and we chose Flask as the web framework for this application because it is more understandable and easier to use.

**IDE**

JetBrains' PyCharm is one of the most widely used Python IDEs. PyCharm is the tool that Python developers need to get the job done quickly. PyCharm allows developers to write clean codes.

# Solution Concept

The fundamental idea behind this application is for the system to recommend a service provider to customers depending on their interests. This might be accomplished by enabling the system to have functionalities for user: customers, user can register/login and use the functionalities made accessible to them. The application's solution idea is illustrated in the High-Level Component Diagram below, which shows the system's six primary components: database, customer, service provider, security, service, and recommender. To begin, the customer will require access control, which will be given by the security component. The customer components will rely on the Home Service Database component to complete this procedure. In order for the customer component to look for and engage with the service component, the service component will require the service provider component to provide their service. Finally, the recommender component will handle the fundamental concept of recommending a service provider, and it will rely on the home service database component to give the necessary data. The recommender component will process this data and provide recommendations to the customer component.

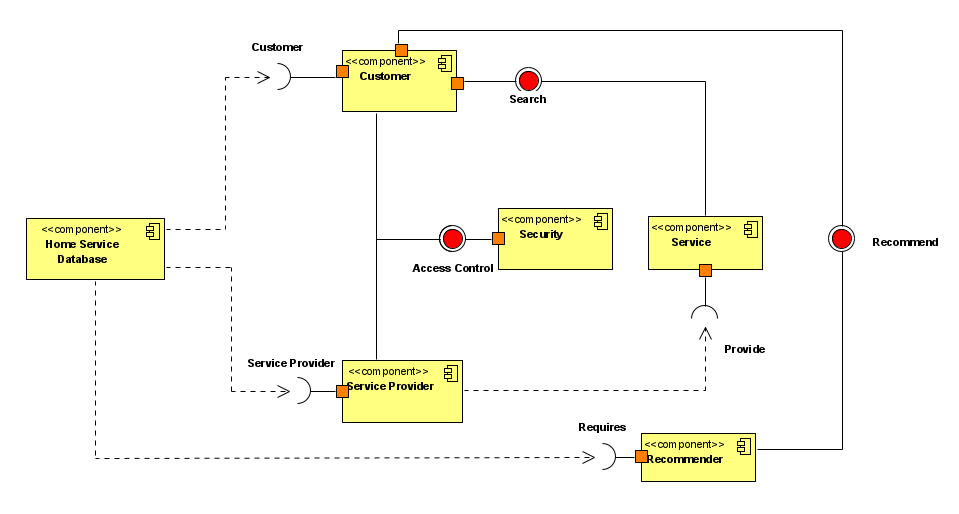


Figure 2: High Level Component Diagram

# System Design

## High-level use case

Diagram

Description automatically generated

Figure 3: High-Level Use Case

|  |  |
| --- | --- |
| Use Case Name | Register |
| Description | Allows the user to register into the system |
| Primary Actor | Customer and Service Provider |
| Pre-Conditions | * System should be up and running |
| Basic Flow | 1. System displays the home page 2. User selects their role 3. System redirects user to their login page 4. User chooses the “sign up” option 5. System redirects user to their register page 6. User enters the required information 7. User clicks the “register” button. 8. System validates the information and saves in database 9. System redirects the user to the login page |
| Extensions | 8A) If any of the required information is empty   1. System prompts the user to enter the required information.   6A) If the entered password and confirm password is not match   1. The system displays error and prompts the user to enter the matching password |
| Post Condition | The user can login to the system. |
| Optional Information | The password can be in between 8 to 25 characters. |

|  |  |
| --- | --- |
| Use Case Name | Login |
| Description | Allows the user to login to the system. |
| Primary Actor | Customer and Service Provider |
| Pre-Conditions | * System must be up and running * User must be registered |
| Basic Flow | 1. System displays the homepage  2. User selects their role  3. System displays the Login page  4. User enters the username and password  5. User clicks “Login” button.  6. The system validates the credentials  7. System direct the user to their home page. |
| Extensions | 5A) If the Username or Password is Empty   1. System prompts the user to enter the username or password   6A) If the user entered incorrect credentials   1. System prompts the user to enter the correct username and password. |
| Post Condition | User can make full use of the available functions. |

|  |  |
| --- | --- |
| Use Case Name | Recommend Service Provider |
| Description | Allows the user to get recommendation from the system. |
| Primary Actor | Customer |
| Pre-Conditions | * System must be up and running * Customer must be logged in |
| Basic Flow | 1. System displays customer homepage  2. Customer clicks “Services”  3. Customer clicks\search “Electricians”  4. System recommends few electricians hired by similar customers |
| Extensions | 3A) If the customer searches unavailable services  1.System displays an error |
| Post Condition | Customer can hire any electricians |

|  |  |
| --- | --- |
| Use Case Name | Search Services |
| Description | Allows the user to search services |
| Primary Actor | Customer and service provider |
| Pre-Conditions | * System must be up and running * User must be logged in |
| Basic Flow | 1. User enters the home page  2. User clicks “Services”  3. System directs the user to service page  4. User searches a home service |
| Extensions | 4A) If the user searches unavailable services  1.System displays an error |
| Post Condition | System recommends few service providers hired by similar customers to the customer  Service provider will be able to see the available service providers |

|  |  |
| --- | --- |
| Use Case Name | Hire Services |
| Description | Allows the user to hire a service provider |
| Primary Actor | Customer |
| Pre-Conditions | * System must be up and running * Customer must be logged in and searched a service provider |
| Basic Flow | 1. System displays service provider page  2. System recommends few service providers hired by similar customers to the customer  3. Customer selects one and clicks “Hire”  4. System processes the function and hires the selected service provider |
| Extensions | - |
| Post Condition | Cancel the service/ hire another service |

|  |  |
| --- | --- |
| Use Case Name | Cancel Hiring |
| Description | Allows the user to cancel the hired service provider |
| Primary Actor | Customer |
| Pre-Conditions | * System must be up and running * Customer must be logged in and hired a service provider |
| Basic Flow | 1. System displays hiring page  2. Customer clicks “Cancel”  3. System processes the function  4. System cancels the hiring |
| Extensions | 3A) If the customer clicks the cancel button after 5 minutes  1.System rejects the cancel request and displays an error |
| Post Condition | hire another service |

|  |  |
| --- | --- |
| Use Case Name | Manage Profile |
| Description | Allows the user to edit their profile |
| Primary Actor | Customer and Service Provider |
| Pre-Conditions | * System must be up and running * Customer must be logged in |
| Basic Flow | 1. User clicks his/her profile  2. System directs the user to profile page  3. User makes changes to his/her profile details  4. User clicks “save”  5. System validates updated details  6. System saves the updated information in database |
| Extensions | 5A) If the entered information is not valid  1.System displays an error and prompts user to re-edit profile |
| Post Condition | Explore other functions |

## Activity Diagram

**Register**

Diagram

Description automatically generated**Login**

Diagram

Description automatically generated

**Recommend Service Providers**

Diagram

Description automatically generated

**Search Services**

Diagram

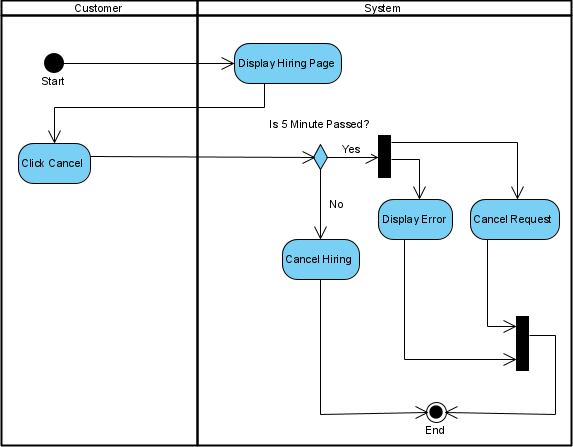
Description automatically generated

**Hire Service**

Diagram

Description automatically generated

**Cancel Hiring**



**Manage Profile**

Diagram, schematic

Description automatically generated

## Sequence Diagram

**Register**

Table

Description automatically generated

**Login**

Table

Description automatically generated with medium confidence

**Recommend Service Provider**

Diagram

Description automatically generated with low confidence

**Hire Service Provider**

Diagram, box and whisker chart

Description automatically generated

**Cancel Hiring**

Diagram

Description automatically generated

**Search Service**

Diagram

Description automatically generated

**Manage Profile**

Diagram

Description automatically generated with medium confidence

# Implementation

Loading dataset

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generatedDropping unwanted columns for training

Checking for null values and dropping it

Graphical user interface, application

Description automatically generated

Visual representation of “City\_of\_work” column with count

Chart, bar chart

Description automatically generated

Chart

Description automatically generatedVisual representation of “work” column with count

Text preprocessing to remove non-ASCII characters and stop words from the work column and convert it into lower case to avoid duplication while finding similarities

Graphical user interface, text, application

Description automatically generated

TF-IDF vectorizer is used to convert words in the “work” column into vectors

Below code will display the data frame of each word used in the “work” column and their tf-idf values will be displayed

Table

Description automatically generated

We are trying to find similarities amongst service providers based on “work” column. We’ll need a basic frequency counter for each word in my “cleaned\_work” column (text preprocessed column) to accomplish this. TF-IDF tends to downplay the importance of words that are more prevalent across the corpus. Because we have a limited number of words in our work column, each word is equally essential. As a result, we choose to employ the "Count Vectorizer."

Graphical user interface, text, application, email

Description automatically generated

We will be generating the similarity of the “cleaned\_work” column across the dataset. For this, we will be using Cosine Similarity algorithm.

Graphical user interface, application

Description automatically generated

Now we will be implementing a function to recommend top 5 service providers by taking work as input and name, city, work, and experience as output.

Text

Description automatically generated

Now for testing purpose we will be giving an input manually

Graphical user interface, text

Description automatically generated

# Reference

Alpaydin, E., 2020. *Introduction to Machine Learning.* 4th ed. Cambridge: Massachisetts Institute of Technology.

Amaratunga, D., Fernando, N., Haigh, R. & Jayasinghe, N., 2020. Progress in Disaster. *The COVID-19 outbreak in Sri Lanka: A synoptic analysis focusing on trends, impacts, risks and science-policy interaction processes,* 8(100133).

Amaratunga, D., Fernando, N., Haigh, R. & Jayasinghe, N., 2020. Progress in Disaster Science. *The COVID-19 outbreak in Sri Lanka: A synoptic analysis focusing on trends, impacts, risks and science-policy interaction processes,* 8(100133).

Chung, L., Brian, N. A., Yu, E. & Mylopoulos, J., 2000. *Non-Functional Requirenments in Software Engineering.* 1st ed. New York: Kluwer Academic Publishers.

De Silva, N. & Ranasinghe, M., 2010. Maintainability risks of condominiums in Sri Lanka. *Journal of Financial Management of Property and Construction,* 15(1), pp. 41-60.

Ekstrand, D. M., Riedl, T. J. & Konstan, A. J., 2010. *Collaborative filtering recommender systems.* 2nd ed. Netherland: Foundation and Trends.

Garcia, R. et al., 2020. A systematic literature review on the use of machine learning in precision livestock farming. *Computers and Electronics in Agriculture,* 179(105829).

Geeksforgeeks, 2020. *Using CountVectorizer to Extracting Features from Text.* [Online]   
Available at: https://www.geeksforgeeks.org/using-countvectorizer-to-extracting-features-from-text/  
[Accessed 23 12 2021].

International Labour Organization, 2020. *Working from Home: Estimating the worldwide potential,* Switzerland: International Labour Organization.

Meteren, R. v. & Someren, M. v., 2000. *Using Content-Based Filtering for Recommendation,* Netherlands: NetlinQ Group.

Paireekreng, W., 2013. *Mobile content recommendation system for re-visiting user using content-based filtering and client-side user profile,* Tianjin: IEEE.

Ribeiro, M., Grolinger, K. & Capretz, A. M., 2015. *MLaaS: Machine Learning as a Service,* Miami: IEEE.

Ryngksai, I. & Chameikho, L., 2014. ecommender Systems: Types of Filtering. *International Journal of Engineering Research & Technology,* 3(11), pp. 251-254.

Salton, G. & Buckley, C., 1988. Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing & Management,* 5(24), pp. 513-523.

Shoval, P., Maidel, V. & Shapira, B., 2008. An Ontology- Content-based Filtering Method. *Information Theories & Applications,* 15(4), pp. 303-314.

Wagner, M., Balke, W. T., Hirschfeld, R. & Kellerer, W., 2002. *A Roadmap to Advanced Personalization of Mobile Services,* Augsberg: ODBASE.

Xu, G., Zhang, Y. & Li, L., 2010. *Web Mining and Social Networking.* 1st ed. New York: Springer.

Ye, J., 2011. Cosine similarity measures for intuitionistic fuzzy sets and their applications. *Mathematical and Computer Modelling,* 53(2), pp. 91-97.