International Islamic University, Chittagong



**Lab Report**

**Project Title:** **Real life problems management system**

Submitted By:

Syed Mohammad Afraim

Sajjad Hossain  
Nabajit Dey Utsab

I.D. No. : C-191077, C-191073, C-191075

Course Title: **Software Engineering Lab**

Course Code: CSE-3638

Session: Autumn 2021

Semester: 6th

Section: 6BM

Email: [c191077@ugrad.iiuc.ac.bd](mailto:c191077@ugrad.iiuc.ac.bd)

[c191073@ugrad.iiuc.ac.bd](mailto:c191073@ugrad.iiuc.ac.bd)

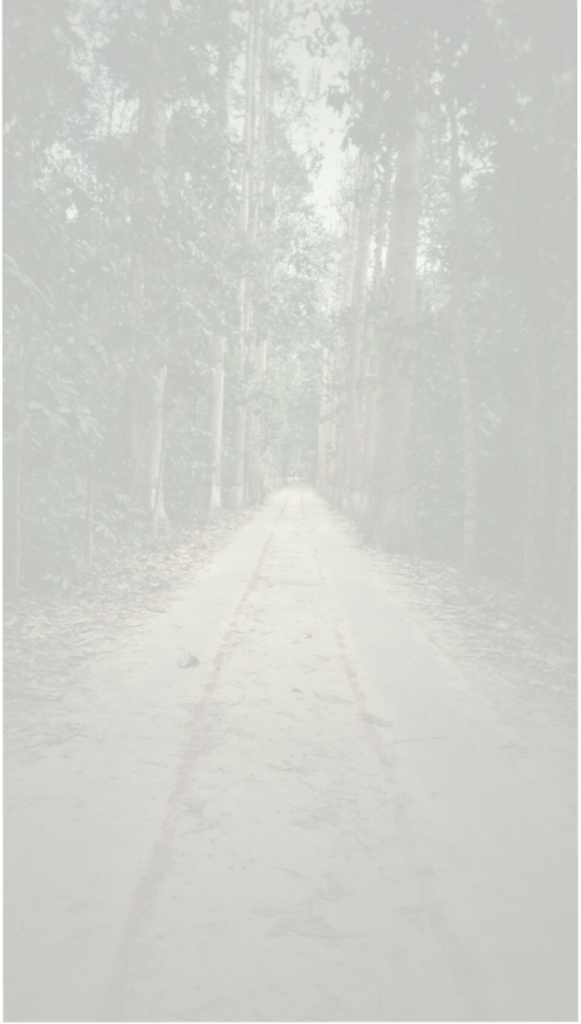
[c191075@ugrad.iiuc.ac.bd](mailto:c191075@ugrad.iiuc.ac.bd)

Submitted to:

Name of Instructor: Mr. Ahmed Shan-A-Alahi,

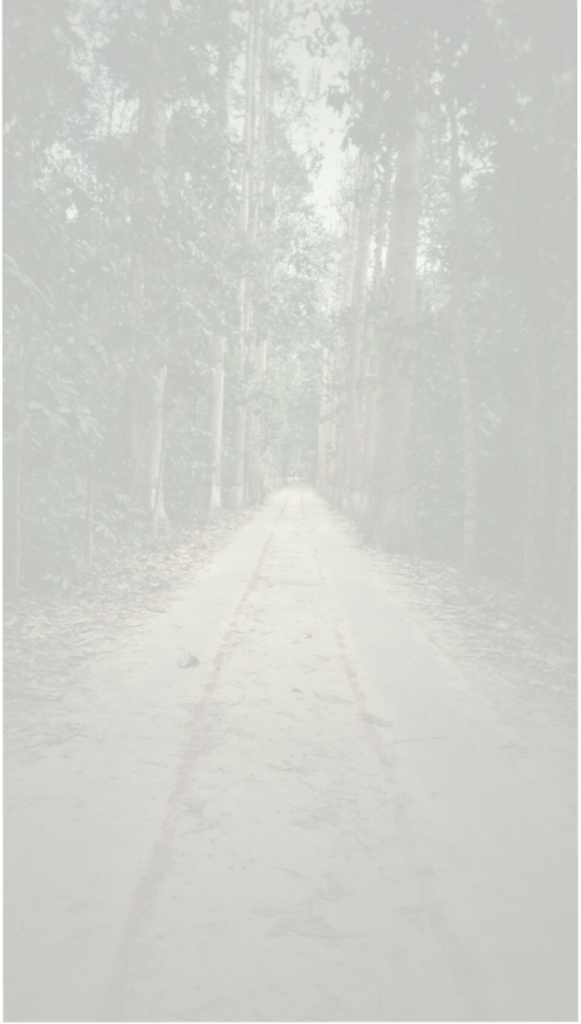
Asst. Prof., Dept. of CSE, IIUC.

Date of Submission:

28.06.2022

**INDEX**

1. Introduction
   * Purpose
   * Scope
   * Overview
   * Additional Information
2. General Description
3. Functional specifications
   * Login
   * Validation
   * Search of problems/issues
   * Payment of money
   * Report
4. Interface Requirements
   * GUI
   * Hardware Interface
   * Software Interface
5. Performance Requirements
6. Constraints
7. Performance
   * Security
   * Reliability
   * Availability
   * Maintainability
   * Reusability
8. References
9. Design and Implementation
   * Class Diagram
   * Data Flow Diagram
   * Use-Case Diagram
10. Conclusion



**Real Life Problems Management System**

**By**

**C191077, C191073, C191075**

**Introduction**

This document gives detailed functional and nonfunctional requirements for the real life problems management system. This product will support online real life problems solution. The purpose of this document is that the requirements mentioned in it should be utilized by software developer to implement the system.

**1.1 Purpose**

Real life problems system is specifically developed for real life problems for any domestic problems, mental health issues and outer problems like theft, robbery or hijacking. The Traditional way of maintaining details of a user in solving a problem was to enter the details or go through doctors, polices to issue complaints and take necessary arrangements. Every time when the user needed to perform some issues fixed, he has to go to hospital or police station for his own problems and perform the necessary actions, which may not be so feasible all the time. It may be a hard-hitting task for the users and the other employees too. The project gives real life understanding of real life problems and its solution and activities performed by various roles in the supply chain. Here, we provide an automation for real life problem solution system through Internet. Real life problem solution project captures activities performed by different roles in real life problem solving which provides enhanced techniques for maintaining the required in- formation up-to-date, which results in efficiency. The project gives real life understanding of Internet real life problem solution and activities performed by various roles in the supply chain.

**1.2 Scope**

This software will automate of issuing individual problems to any employees in the process. This Project investigates the entry threshold for providing a new real life service channel via the real options approach, where the entry threshold is established by using an Internet real life problems solution system designed for the use of normal users (individuals), Employees of different fields like doctors, lawyers, psychiatrists, Staffs like riders, managers under the whole supervision of an admin.

**1.3 Overview**

The system provides easy solution to real life problems.

**Overview:** The SRS will include two sections, namely:

Overall Description: This section will describe major components of the system, interconnections, and external interfaces.

Specific Requirements: This section will describe the functions of actors, their roles in the system and the constraints faced by system.

**2. General description**

**2.1 Product Perspective:**

The client will have client interface in which he can interact with the real life problems solution system. It is a web-based interface which will be the web page of the real life problems solution system application. Starting a page is displayed asking the type of customer he is whether ordinary customer or a member. Then the page is redirected to login page where the user can enter the login details. If the login particulars are valid then the user is taken to a home page where he has the entire transaction list that he can perform with the lists of problems he acquired help for. All the above activities come under the client interface.

The administrator will have an administrative interface which is a GUI so that he can view the entire system. He will also have a login page where he can enter the login particulars so that he can perform all his actions. This administrative interface provides different environment such that he can maintain database & provide backups for the information in the database. He can register the users by providing them with username, password & by creating account in the database. He can view the request or submission of the users’ individual problems & perform action to issue the necessary arrangement to the employees and staffs so that the users can get their required help in return.

**2.2 Software Interface**:

Front End Client:

The system is a web-based application clients are requiring using modern web browser such as Mozilla Firefox 1.5, PHP.

\* Web Server:

The web application will be hosted on one of the Apache server.

\* Back End:

We use backend as Django, SQLite

**3. Functional Specifications**

This section provides the functional overview of the product. The project will require the PHP as a front end and at the back end the database MYSQL will be running.

Various functional modules that can be implemented by the product will be

1. Login

2. Validation

3. Search problems

4. Payment

5. Submit a problem

6. Customer info.

**3.1 Login:**

Customer logins by entering customer name & a login pin.

**3.2 Validation:**

When a customer enters the password, its validity must be ensured. Then customer is allowed to enter into the software. The validation can be for following conditions.

Validation for membership card’s expiry date

If the membership inserted by the customer has crossed the expiry date then the system will prompt

“Expired Membership”.

Get balance information:

This system must be networked to the admin’s computer. The updated database of every customer is maintained with admin. Hence the information of every account is available in the database and can be displayed to the customer.

**3.3 Search of problems/issues**

The customer can search their own problems browsing the entire catalog and then submit problems and contact to the respective employees (doctors, lawyers etc) directly through a payment.

**3.4 Payment of Money:**

A customer is allowed to enter the amount which he/she wishes to withdraw. If he cancels his order replacement for the problem he searched for, there would be a fine related to that otherwise through proper reasoning his cancellation wouldn’t be a factor and there wouldn’t be any extra fees regarding that.

**3.5 Report:**

The online report card of their individual issues that contains information of corresponding account must be printed by the machine.

**3.6 Technical Issues**:

This product will work on client-server architecture. It will require an internet server and which will be able to run PHP applications. The product should support some commonly used browsers such as Internet Explorer, Mozilla Firefox.

**4. Interface Requirements**

**4.1 GUI**

This is interface must be highly intuitive or interactive because there will not be an assistance for the user who is operating the System. At most of the places help desk should be provided for users’ convenience. The screens appearing should be designed in such a manner that it can draw User attraction towards the new plans for the customers.

Also, the pin and password confidentiality should be maintained. This can be done by using asterisks at the password panel. Proper security messages should be displayed at most of the places.

**4.2 Hardware Interface**

Various interfaces for the product could be

1. Touch screen/Monitor

2. Keypad

3. Continuous battery backup

4. Printer which can produce the hard copy.

5. Interface that connects the device to bank’s computer.

6. An interface that can count currency notes.

**4.3 Software Interface**

1. Any windows operating system.

2. The python and django must be installed. For the database SQlite is there. These products are open source products.

3. The final application must be packaged in a set up program, so that the products can be easily installed on machines. This application must be networked to corresponding users/employees/staffs.

**5. Performance Requirements**

The system should be compatible enough to hold the general traffic. It should not get hang or show some other problems arising out due to large no of concurrent users. The system should be fast enough to meet the customer. The high and low temperature should not affect the performance of the device.

**6. Constraints**

\* The information of all the users must be stored in a database that is accessible by the online real life problem solution System.

\* The Online real life problem solution System is connected to the computer and is running all 24hours a day.

\* The users access the real life problem solution System from any computer that has Internet browsing capabilities and an Internet connection.

\*The users must have their correct usernames and passwords to enter into the Internet real life problem solution system.

**Design Constraints:**

\* Software Language Used

The languages that shall be used for coding Online Banking System are c , c++ ,Python, CSS, and HTML. For working on the coding phase of the Online job portal System Web Sphere Application Server/WebSphere Application Server CE Server needs to be installed.

\*Database design

In our database design, we give names to data flows, processes and data stores.

Although the names are descriptive of data, they do not give details. So, following DFD, our interest is to build some details of the contents of data flows, processes and data store. A data dictionary is a structured repository of data about data. It is a set of rigorous definitions of all DFD data elements and data-structures .

**7. Performance**

7.1 Security

The real life problem solution system must be fully accessible to only authentic user.

It should require pin for entry to a new environment.

7.2 Reliability

The application should be highly reliable and it should generate all the updated information in correct order.

7.3 Availability

Any information about the account should be quickly available from any computer to the authorized user. The previously visited customer’s data must not be cleared.

7.4 Maintainability

The application should be maintainable in such a manner that if any new requirement occurs then it should be easily incorporated in an individual module.

7.5 Portability

The application should be portable on any windows-based system. It should not be machine specific.

**8. References:**

* www.w3schools.com
* www.roseindia.net
* www.dbforums.com
* www.ibm.com
* http://tomcat.apache.org/

**9.1 Class DIAGRAM:**

Classes of real life problem solution system Class Diagram:

1. Customer Class: Manage all the operations of Customer.
2. Employees Class: Manage all the operations of Employees.
3. Accounts Class: Manage all the operations of Accounts.
4. Problems Class: Manage all the operations of Problems.
5. Staffs Class: Manage all the operations of staffs.

**Classes and their attributes of Real life Management System Class Diagram**:

• Customer Attributes: customer\_id, customer\_name, customer\_mobile,

customer\_email, customer\_username, customer\_password, customer\_address.

• Employees Attributes: employee\_id, employee\_name, employee \_mobile,

employee\_email, employee \_username, employee \_password, employee

\_address.

• Accounts Attributes: account\_id, account\_customer\_id, account\_number,

account\_type, account\_balance, account\_description.

• Problems Attributes: problems\_id, problems\_customer\_id, problems \_amount,

problems\_total, problem\_title, problems\_type, problems\_description.

• Staffs Attributes: staff\_id, staff \_name, staff \_mobile, staff \_email, staff \_username, staff \_password, staff

\_address.

**Classes and their methods of Internet Banking Management System Class Diagram:**

• Customer Methods: addCustomer(), editCustomer(), deleteCustomer(),

updateCustomer(), saveCustomer(), searchCustomer().

• Employees Methods: addEmployees(), editEmployees(), deleteEmployees(),

updateEmployees(), saveEmployees(), searchEmployees().

• Account Methods: addAccount(), editAccount(), deleteAccount(),

updateAccount(), saveAccount(), searchAccount().

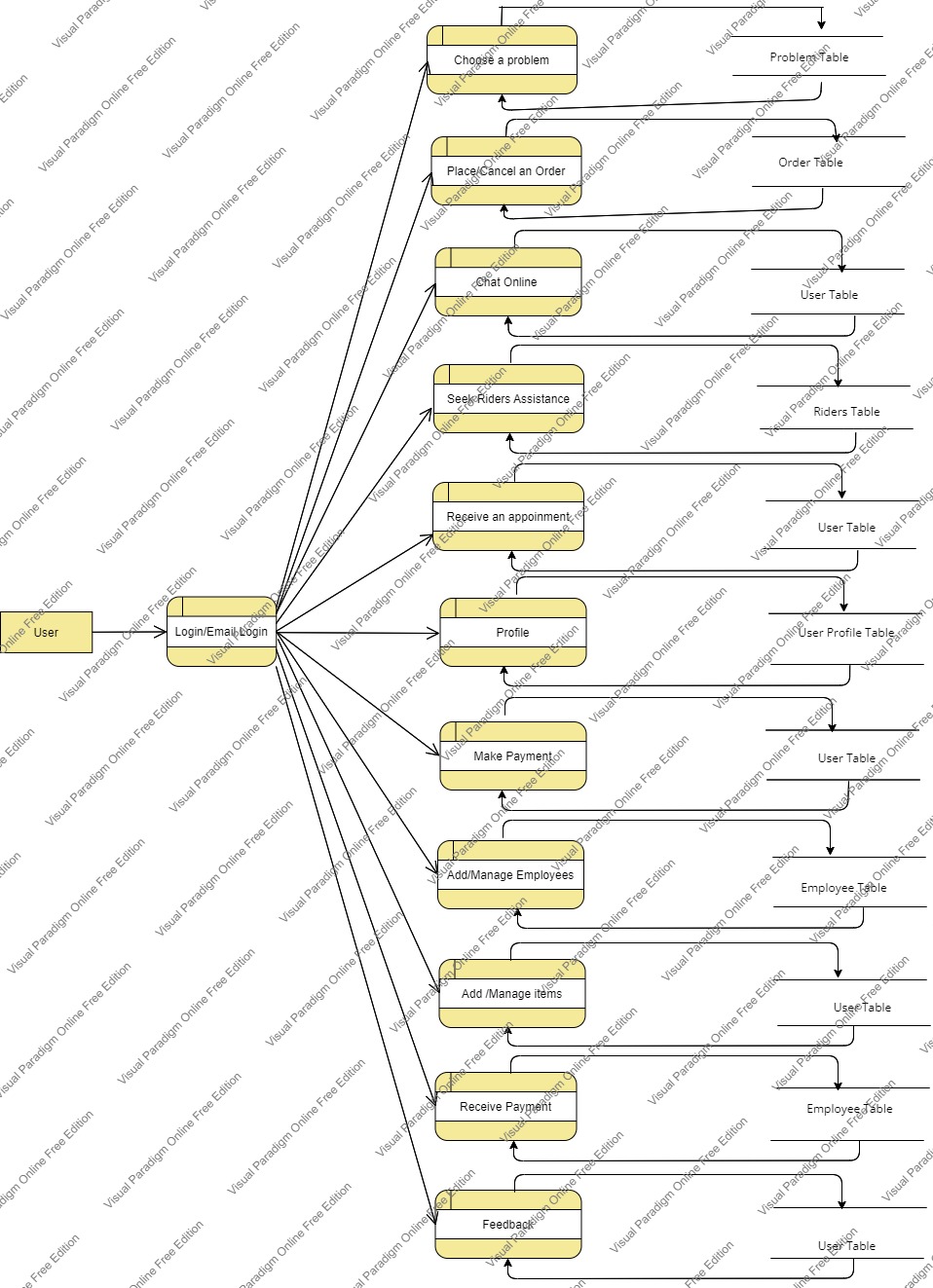
• Problem Methods: addproblems(), editproblems(),

deleteDeposit(), updateDeposit(), saveproblems(),

searchproblems().

• Staffs Methods: addStaffs(), editStaffs(), delete Staffs(), updateStaffs(), saveStaffs(), search Staffs().

DFD Diagram of **real life problem solution** Management System:



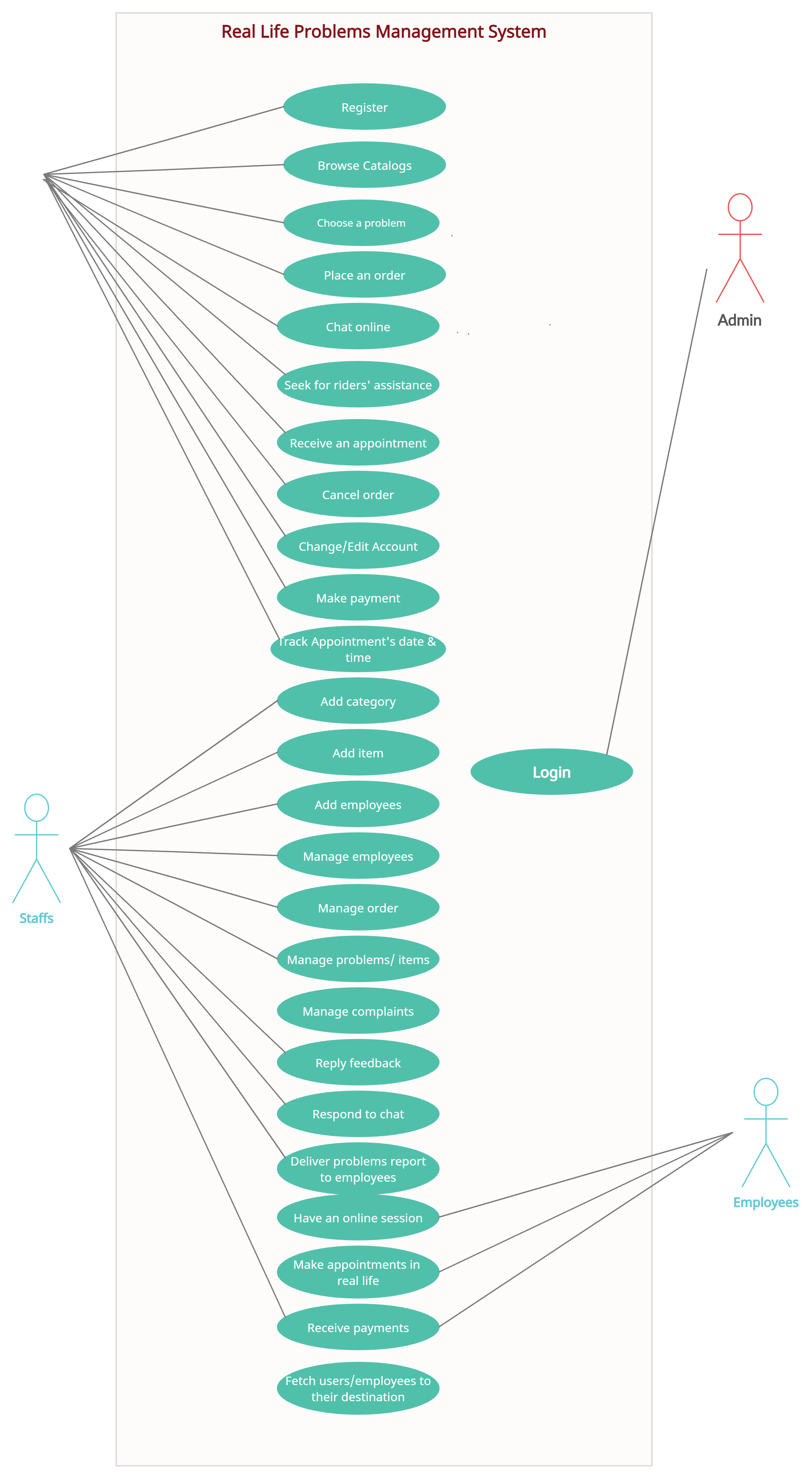
* **Use-Case Diagram**

Use-cases are a scenario-based requirements discovery technique which identifies the actors involved in an interaction and names the type of interaction. This is then supplemented by additional information describing the interaction with the system. The additional information may be a textual description or one or more graphical models such as sequence or state diagrams.

This Use Case Diagram is a graphic depiction of the interactions among the elements of Real life problems Management System. It represents the methodology used in system analysis to identify, clarify, and organize system requirements of Real life problems Management System. The main actors of Real life problems Management System in this Use Case Diagram are: Admin, Customer, Employees, Staffs/Riders who perform the different type of use cases such as Search a problem, Place an order, Manage Employees, Manage order, Chat online, Reply to feedbacks, make appointments and Full Real life problems Management System Operations. Major elements of the UML use case diagram of Real life problems Management System are shown on the picture below.

<<include>>

User



The relationships between and among the actors and the use cases of Real life problems Management System:

* **Admin Entity:** Use cases of Admin are Manage Customer, Manage Employees, Manage Accounts, Manage Balance, Manage Users and Full Real life problems Management System Operation.
* **Employee Entity:** Use cases of Employee are Manage Customer, Manage appointments, Manage report, Manage Customer submission, Receive payments.
* **Customers/Users Entity:** Use cases of Customers are Login, Submit a problem, make payment, search problem, chat online, place order, track their appointment’s date and time.
* **Staffs Entity:** Use cases of Staffs are manage customers, manage reports, manage problems, add new problems, reply feedbacks, reply chats, deliver problem reports to employees.

There is include relation with customer because customer must be login first to issue a problems and gain help and this login must be verified by admin.

There is exclude relation between placing an order and getting discounts based on it because discounts’ request of customer wouldn’t be activated without an order placement.

**10. Conclusion:**

Real life problems management system is a virtualization of problems solution in every way of his/her own mental health, safety and other related purposes. The solving system are infact done manually in real life but when it comes to the utilization of the online problems managing system it becomes ease for many of us in solving basic real life problems. If user makes a complaint/issue or wants help in the real life problems management system it would be overcome with a real time solution virtually and as easy as possible. Real life problems management system is saving the time with accuracy than going to any places to solving that problem. Infact, it’s time, labor & cost efficient.