*Project Title: Service Management System*

*Group Member Information:*

*Name: Kashfia Saif*

*ID: 2018-2-60-001*

*Name: Sayed Monshurul Alam*

*ID:2018-2-60-123*

*Submitted to:*

***Md Mohsin Uddin***

*Senior Lecturer*

*Department of Computer Science and Engineering*

*East West University*

***Date of Submission****:*

*03.01.2021*

**TABLE OF CONTENTS**

**Topic Page**

**Introduction & Motivations** ………………………………………………… 3

**Challenges** …………………………………………………………………3 - 4

**Requirements** ……………………………………………………………….5 - 7

Non-functional Requirements ……………………………………………5

Functional Requirements …………………………………………….. 5 - 7

**Technologies** ……………………………………………………………………..7

**UML Use Case Diagram**………………………………………………………8-9

**UML Class Diagram**……………………………………………………………10

**Sequence Diagram** ……………………………………………………………..11

**Activity Diagram** ………………………………………………………………12

**ER Diagram** ……………………………………………………………………13

**GUI Screenshots** ……………………………………………………………13-20

**Introduction**

Service management system is a service providing platform that makes life easy and saves time for those who can’t go and take service directly. Moreover in case of emergency clients can call any type of service provider such as- electrician, barber,maid,plumber etc. With this web application users will be able to call service providers and get the service at home. This document contains the software requirements specification(SRS) for said app.

**Motivations**

* The main goal of our app is to give people the opportunity to get services by booking through an app. For example-if one needs an electrician then they’ll have to go to the shop and ask them if anybody is free to come and fix any electronics stuff at home. But in this web application client can book an electrician sitting at home and get the service.
* Our app is targeted to make life easier for people by providing service. For example-In this pandemic,people are at home so if they need barber they can’t go to salon because of safety issues. So in this situation they can call barber through app and get the service.
* Through this app clients can give review after taking service and recommend for future. All of that information available will be constantly available through the app’s intuitive UI.

**Challenges**

* Implementing certain features such as location sharing in a web-application.
* Since this project is entirely team driven, we lack any client or upper-management feedback. Lacking well-defined client-requirements will make the development process a little more difficult.
* Adding to the previous point, since we lack any client-defined deadlines, proper time management will also be an issue. To remedy this, the team as decided to use the Agile Scrum development method.
* Integrating several APIs into the application and ensuring they work properly together.
* Getting the team familiar with technologies they haven’t worked with before, namely, Django.
* Making sure the UI isn’t too technical or cluttered and follows the latest design trends.
* Not being able to test the responsiveness of the app in certain devices/platforms the team doesn’t have access to, such as Safari browser (iOS), smart-televisions, etc.
* Internet Explorer probably won’t be able to support many of the apps features. However since IE isn’t very popular with the targeted audience, this should be considered a minor hindrance at best.
* Ensuring strict client security and protecting the client’s data.

**Requirements:**

**Non-Functional Requirements**

* We have to thoroughly ensure the user’s data and privacy is protected. Since our users will entrust us with their password and security questions, highest priority should be given to protect such sensitive information.
* Making an user friendly UI.

**Functional Requirements**

* **User authentication:** Since our app deal with sensitive information about the client’s information, a thorough authentication procedure should be followed.

**Scenario:** When a user tries to access their account or home/profile page while they are not logged in, they would be redirected to the login page. Once the user enters their username and password and authentication is complete, they will be redirected to their desired page. Pages with sensitive/personal information can only be accessed through logging in to the system.

**There will be 3 types of users: i. Admin, ii. Customer iii. Employee**  
**Admin:** Administrators of the system.

**Input:** username, password **Output**: user authentication,admin panel, access to administrative functions, logout.

**Customer:** General end-users of our system; the targeted audience.

**Input:** username, password.

**Output:** user authentication, access to user homepage & profile, logout,all services,orders.

**Employee:** Employees who are going to provide the service to the clients.

**Input:** username, password.

**Output:** user authentication, access to user homepage & profile, logout,order request.

* **Customer profile:** User profiles should contain first name,last name,user name, email,password,user image,dob.gender,phone. no,address etc.

**Scenarios:**

**Make appointments :** Customers have the ability to make appointments. Appointment should have time and address and what service is needed.

**Finding Employees:** Customers can see our database for employees who can provide that particular service.

**Category-wise employee list:** Customer can browse the list of category in which they want the service.

**Sending Requests:** Customer can send requests to check if any employee is available for any particular service.

**Customizing Customer Settings:** Customers can customize their accounts through the profile option. There should be options to edit personal and account information (such as name, password, username, avatar, contact information, etc.) and manage their privacy and security settings.

* **Employee profile:** Worker profiles should contain employee name, employee avatar,service that he/she wants to provide,working experience,packages etc.

**Scenarios:**

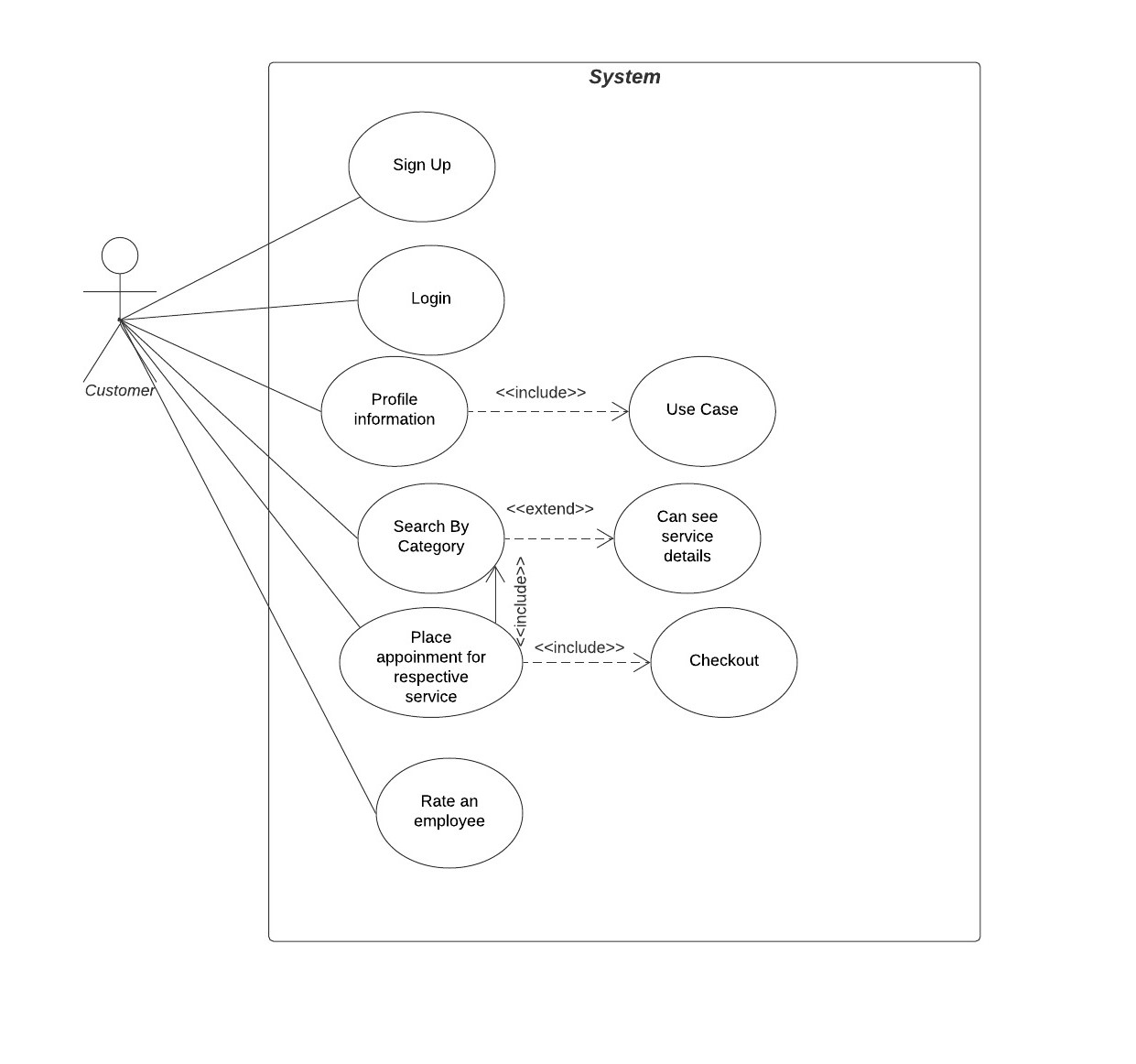
**Appointment Notification:** If any customer sends request for a particular service and any employee is available at that time will get notification.

**Receiving Requests:** Employee can accept or decline any request sent by customer.

**Customizing Employee Settings:** Employees can customize their accounts through the profile option. There should be options to edit personal and account information (such as name, password, username, avatar, contact information, etc.) and manage their privacy and security settings.

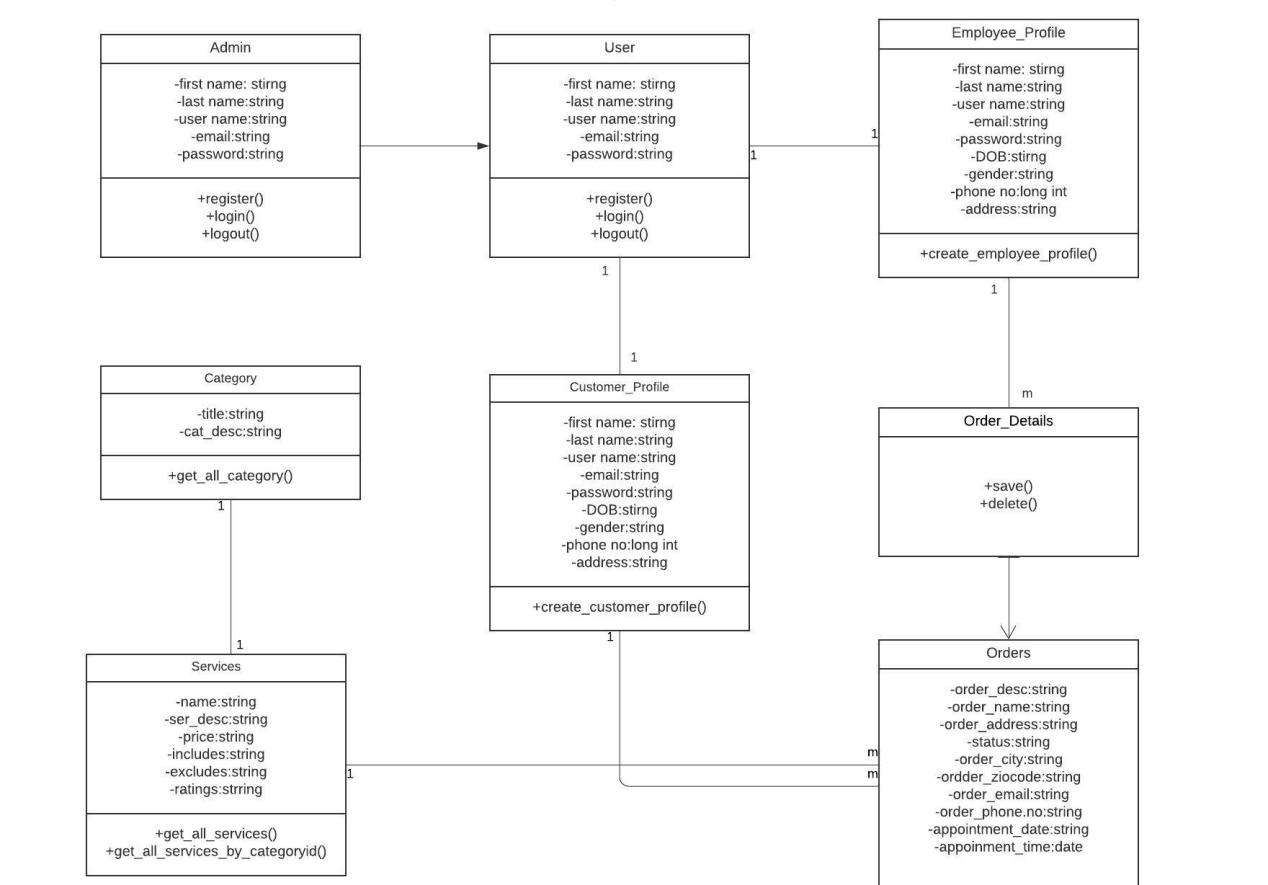
**Technologies:** HTML,CSS,Python,Django.

UML Use Case Diagram:

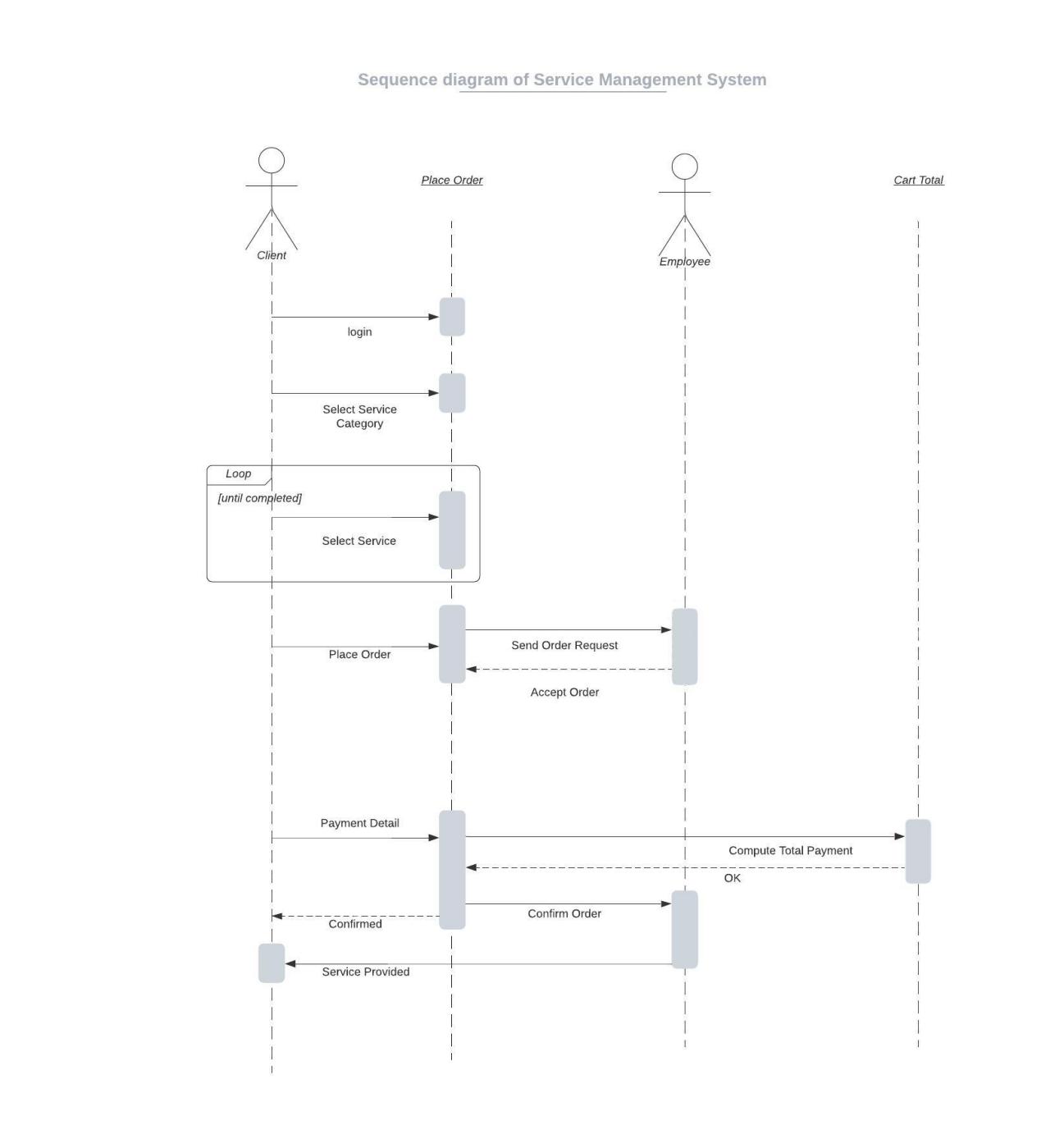




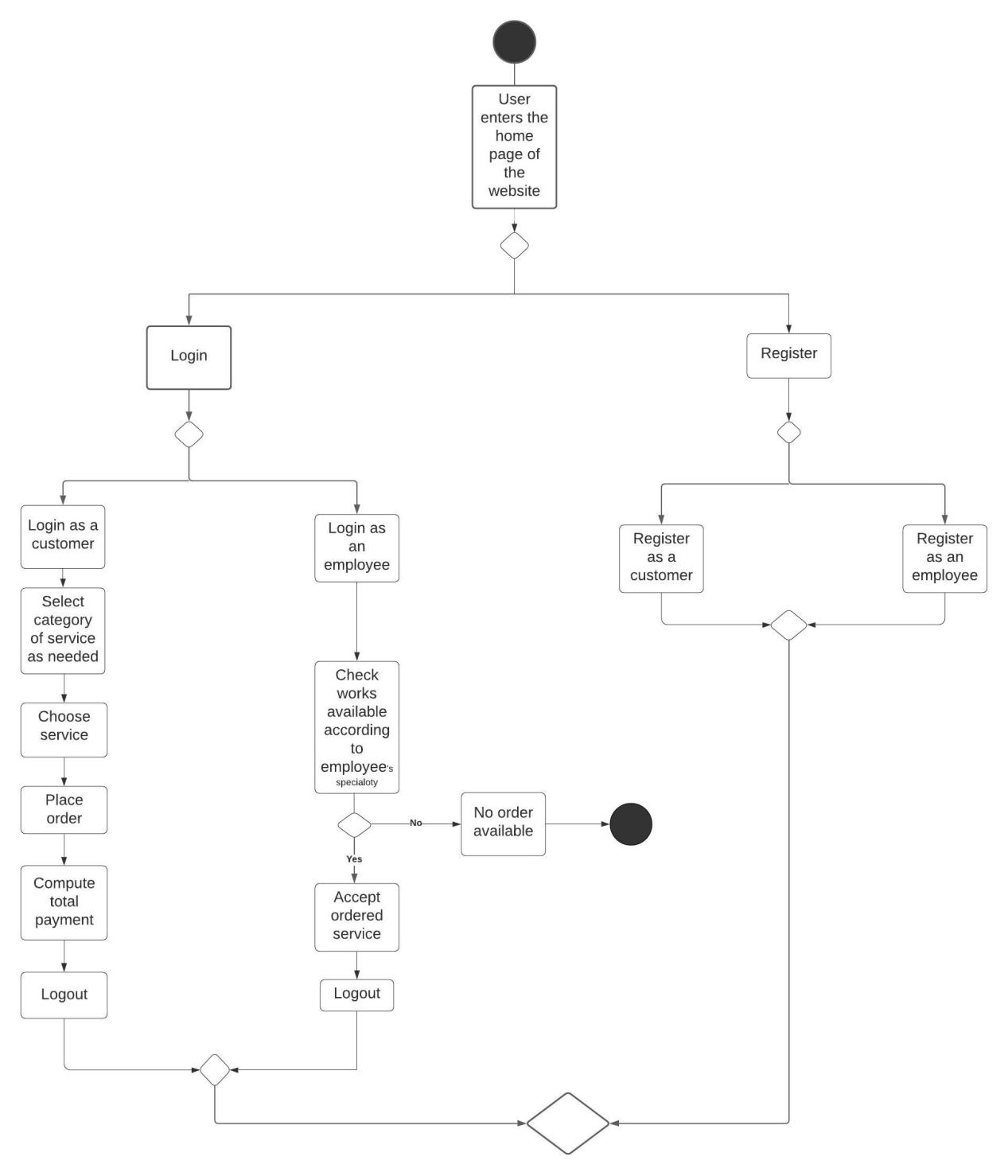
UML Class Diagram:



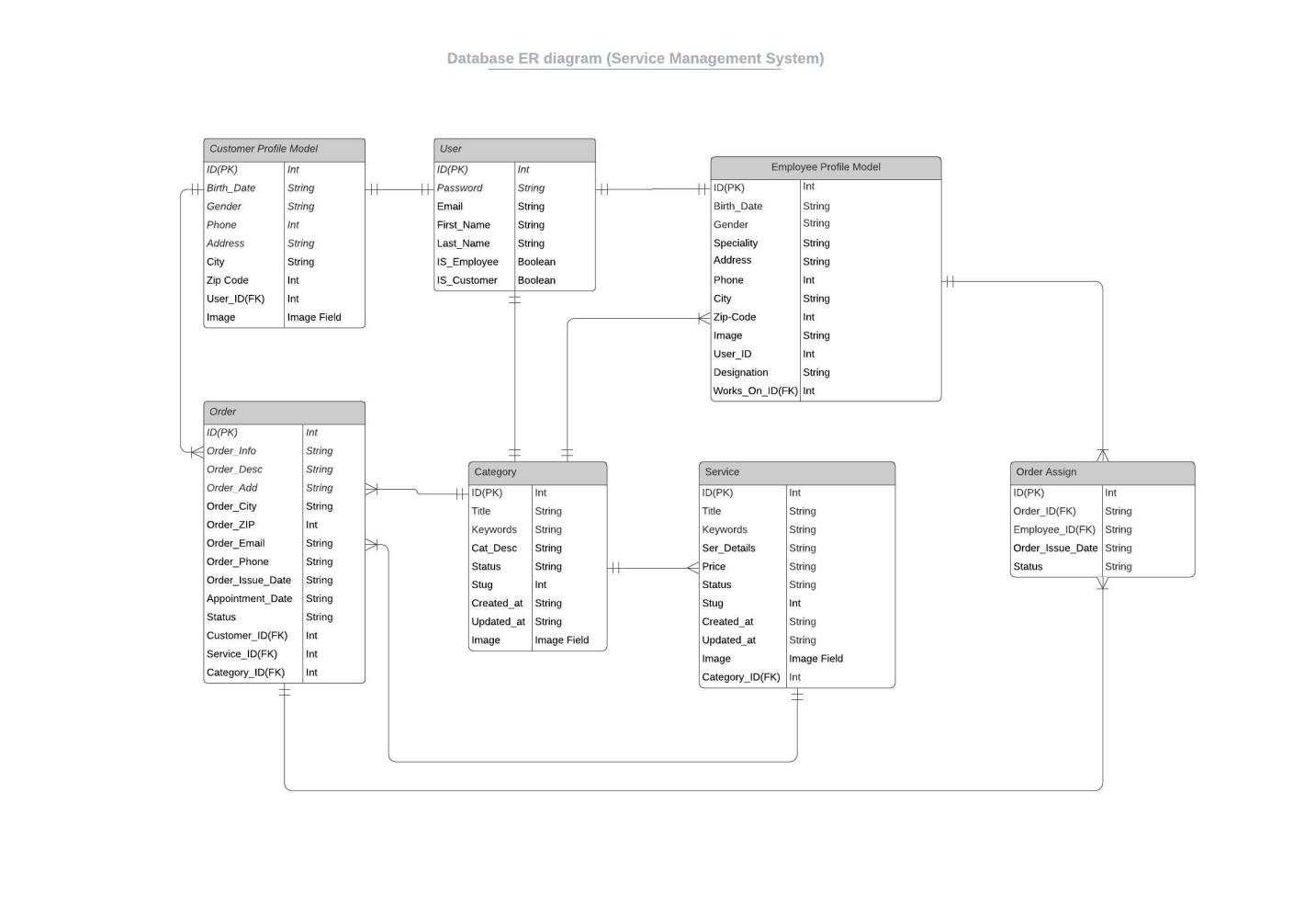
Sequence Diagram:



Activity Diagram:



ER Diagram:



GUI Screenshots:

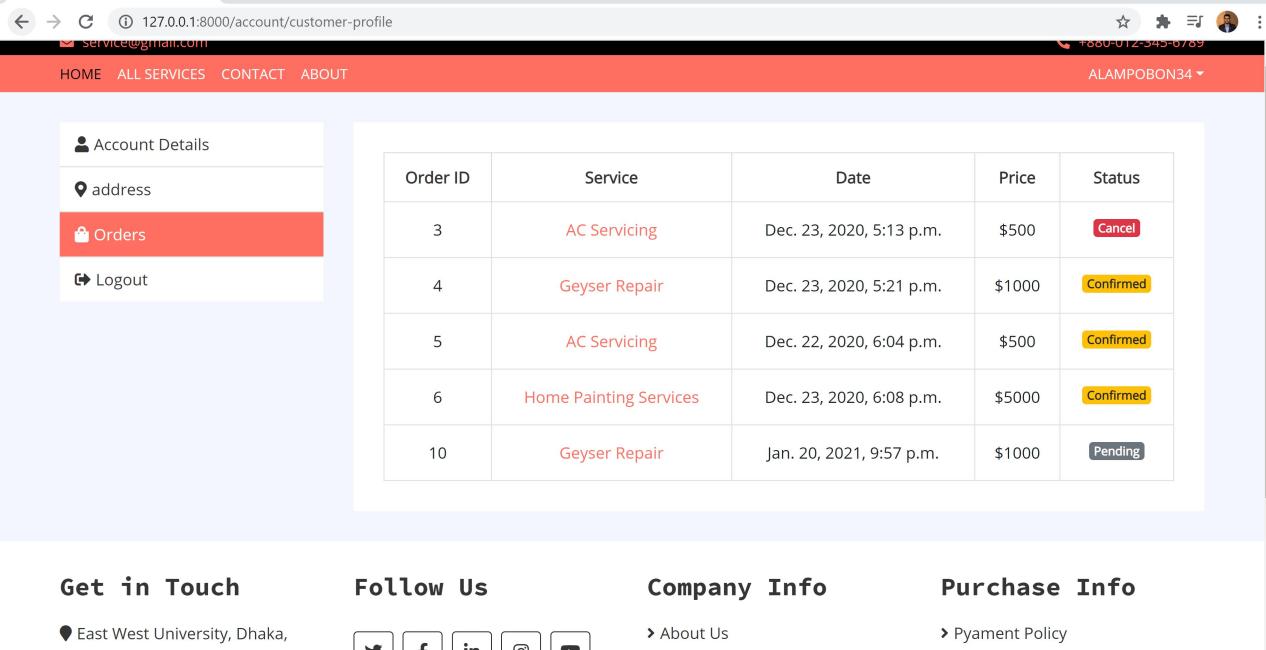


Fig: All Order

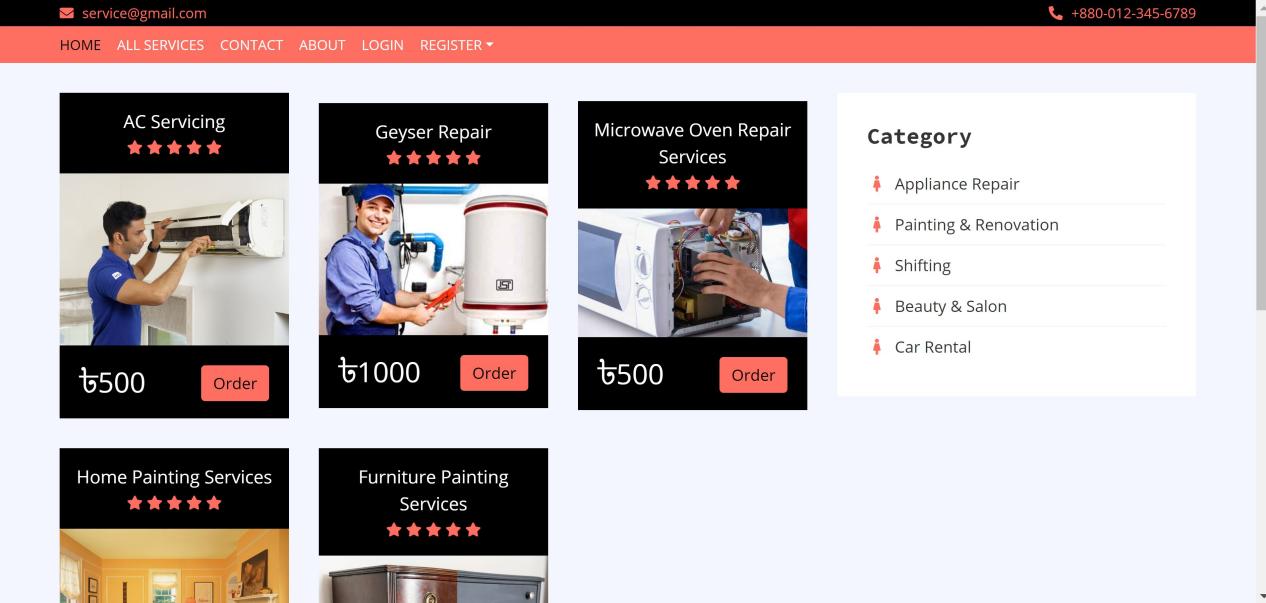


Fig: All Services

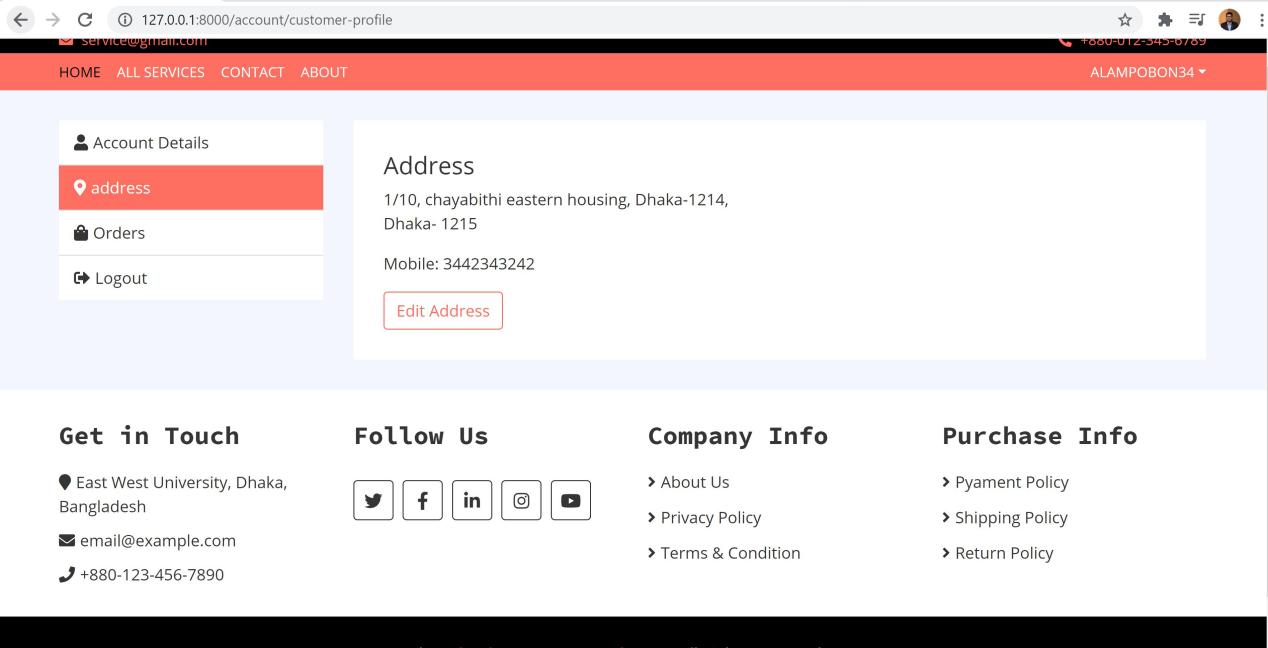


Fig: Customer Address

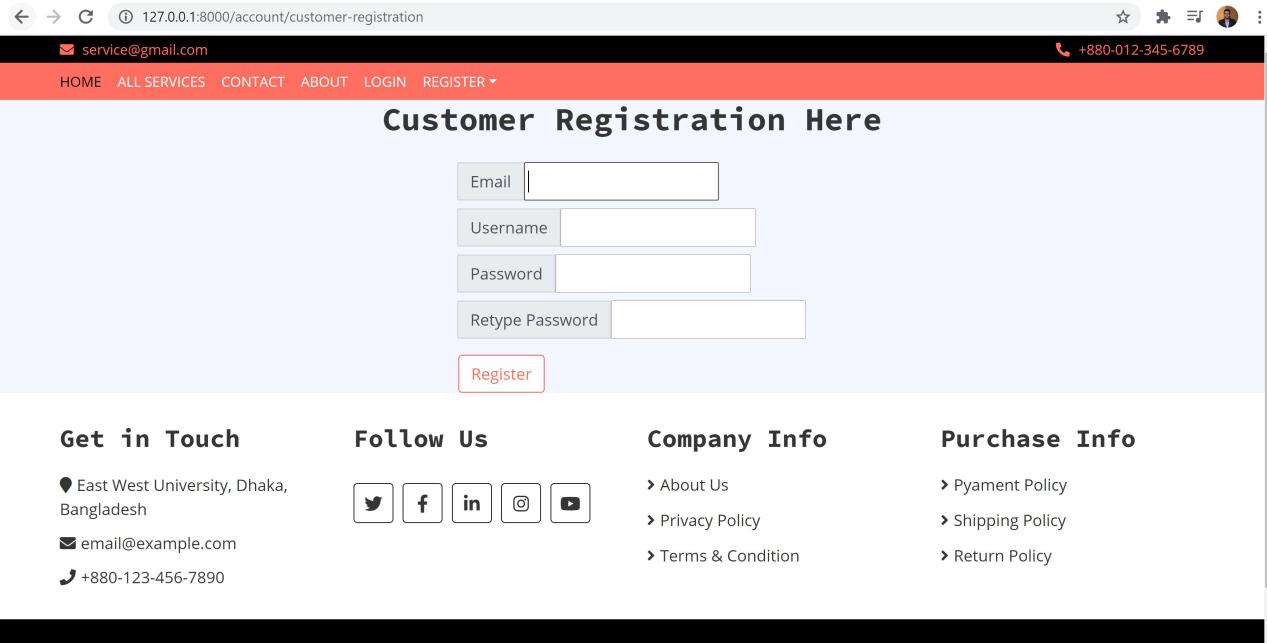


Fig:Customer Registration Form

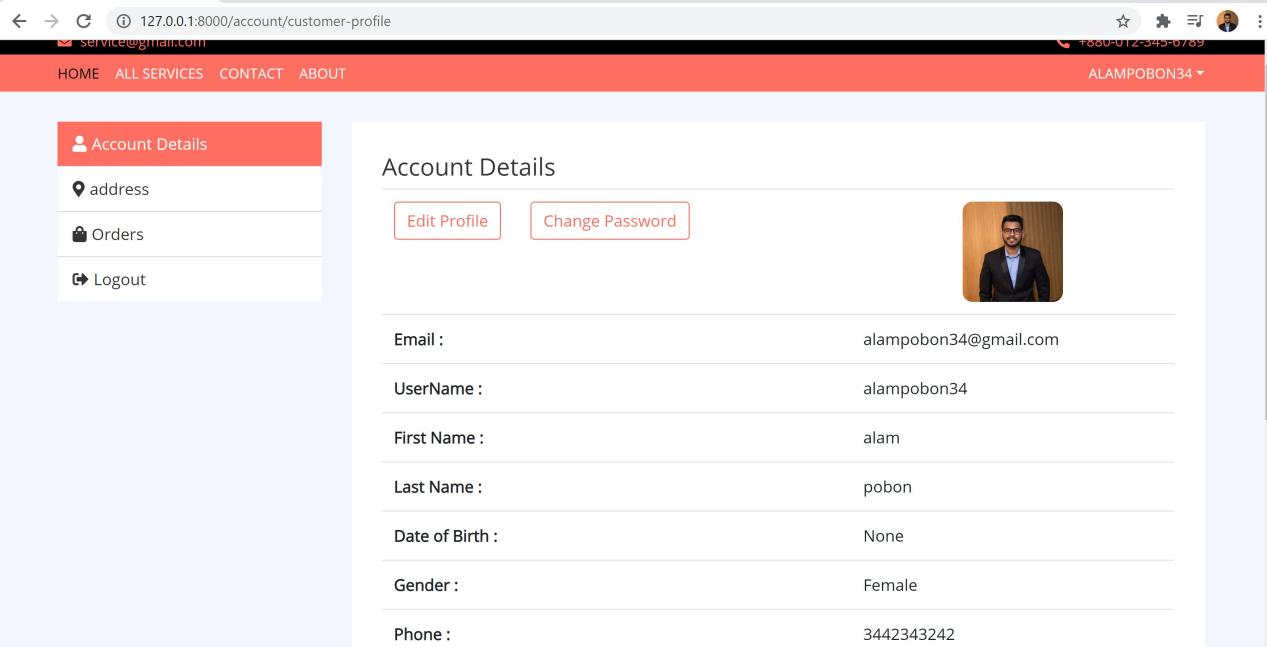


Fig: Customer Profile

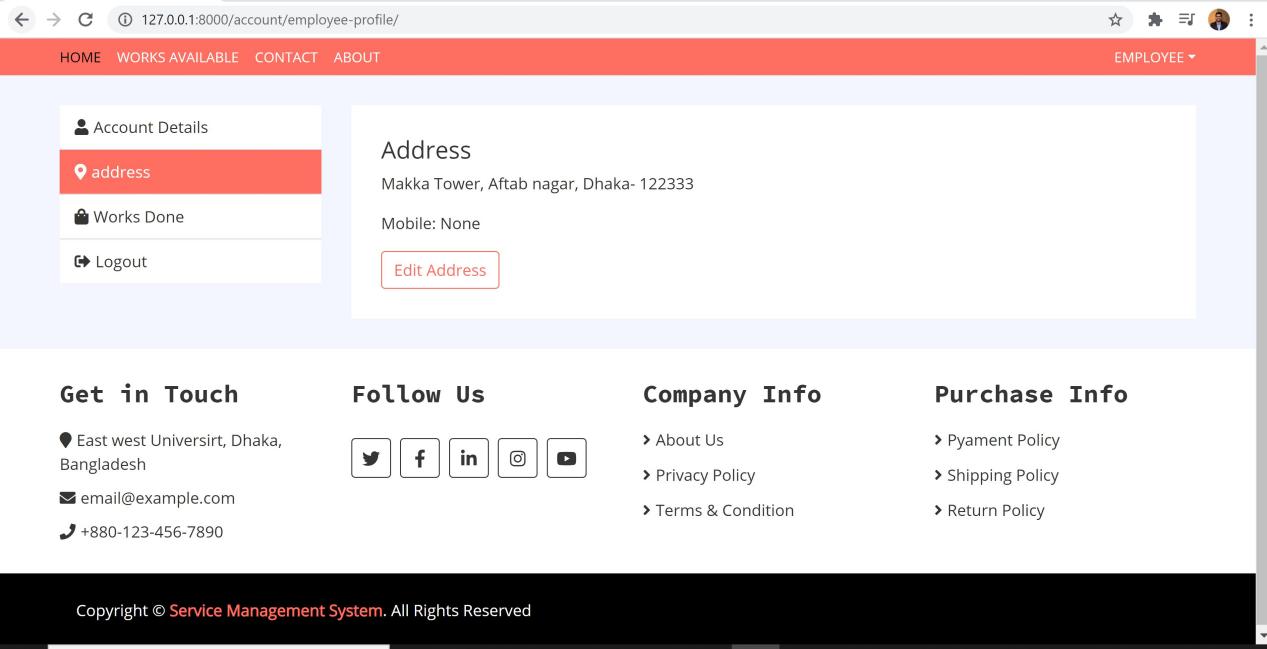


Fig:Employee Address

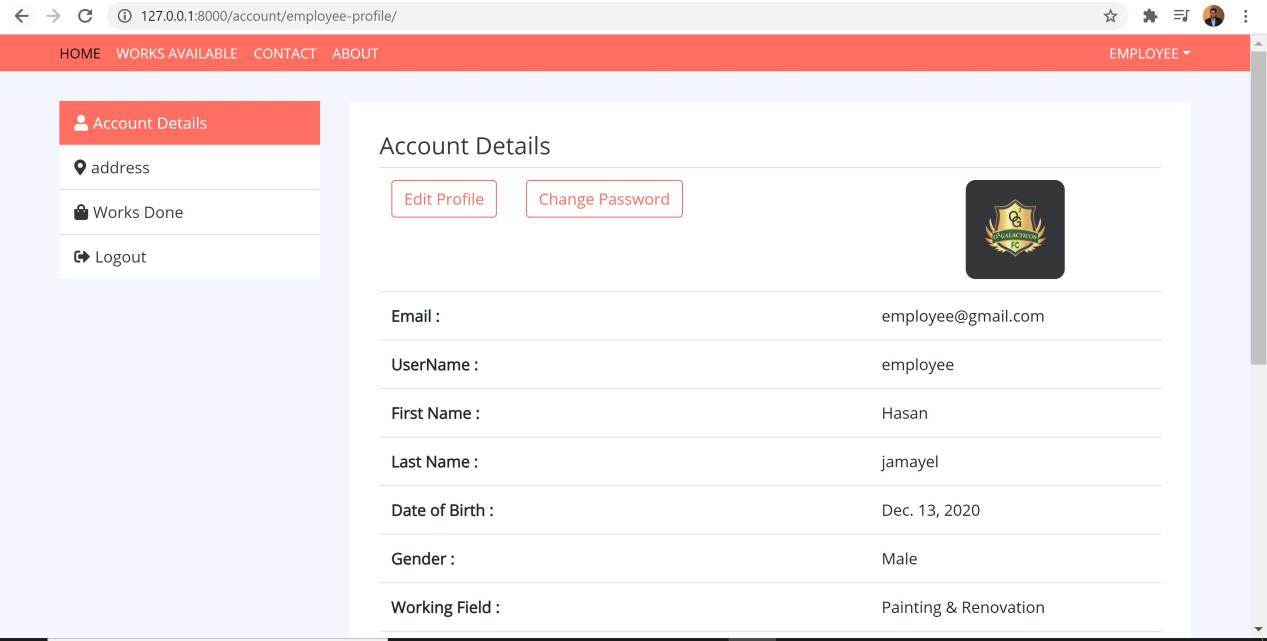


Fig:Employee Profile



Fig:Employee Home

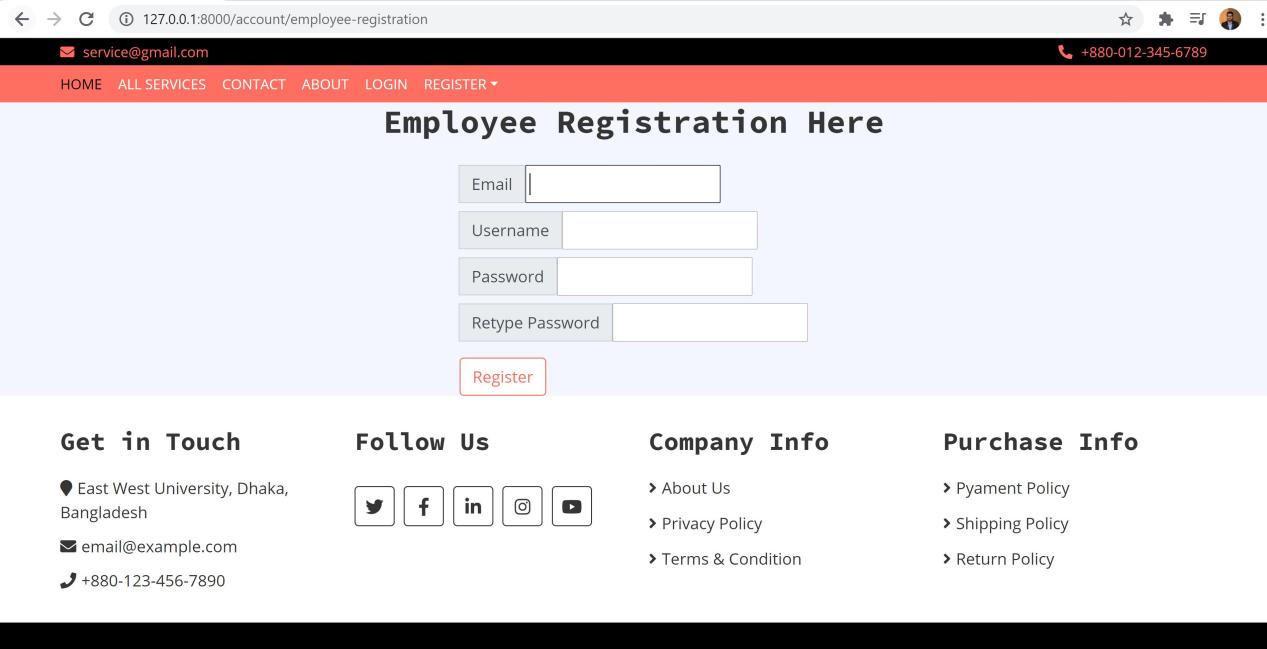


Fig: Employee Registration

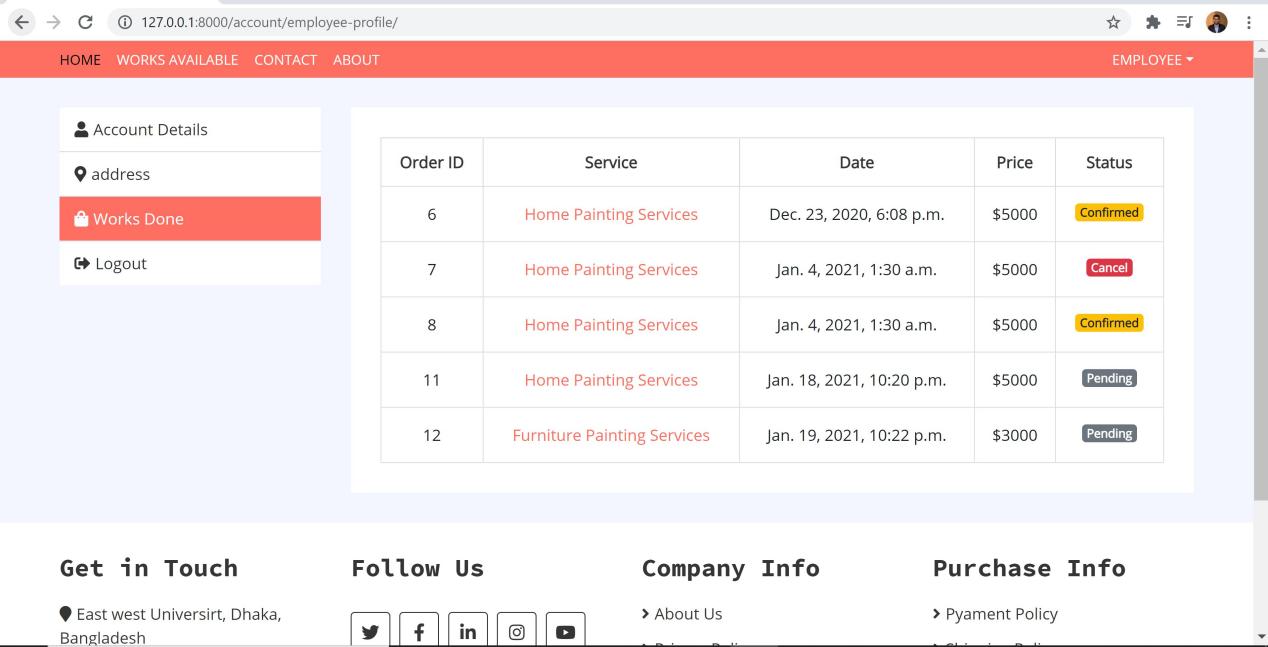


Fig: Employee Work Done

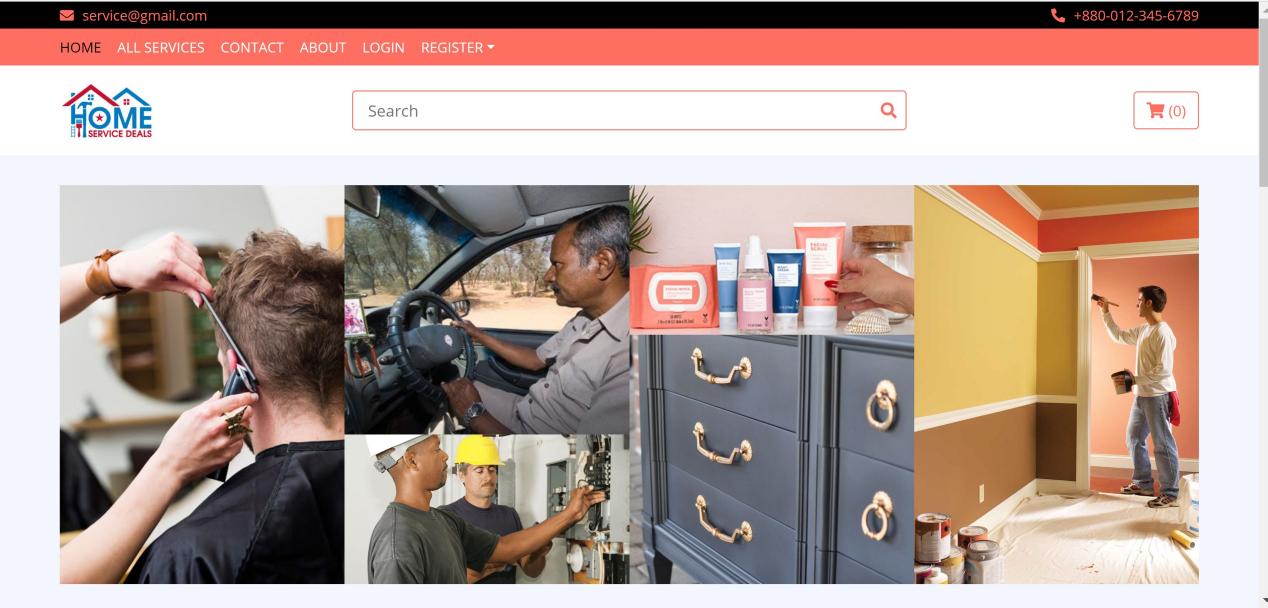


Fig: Home

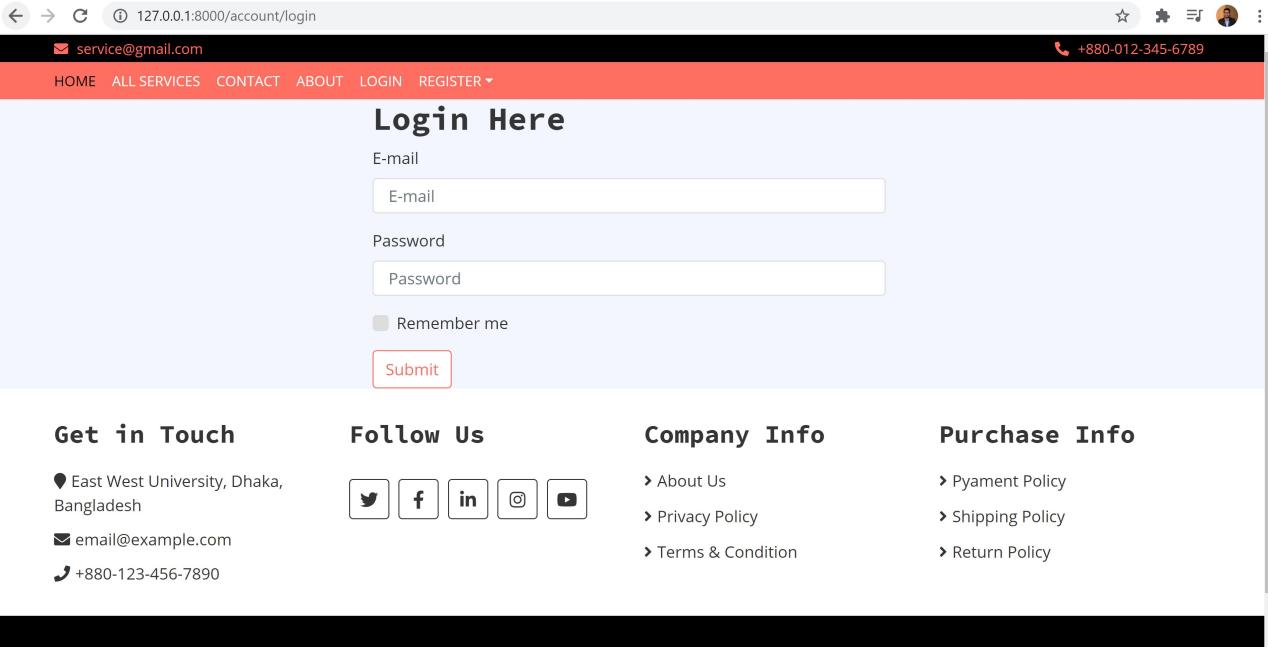


Fig:Login

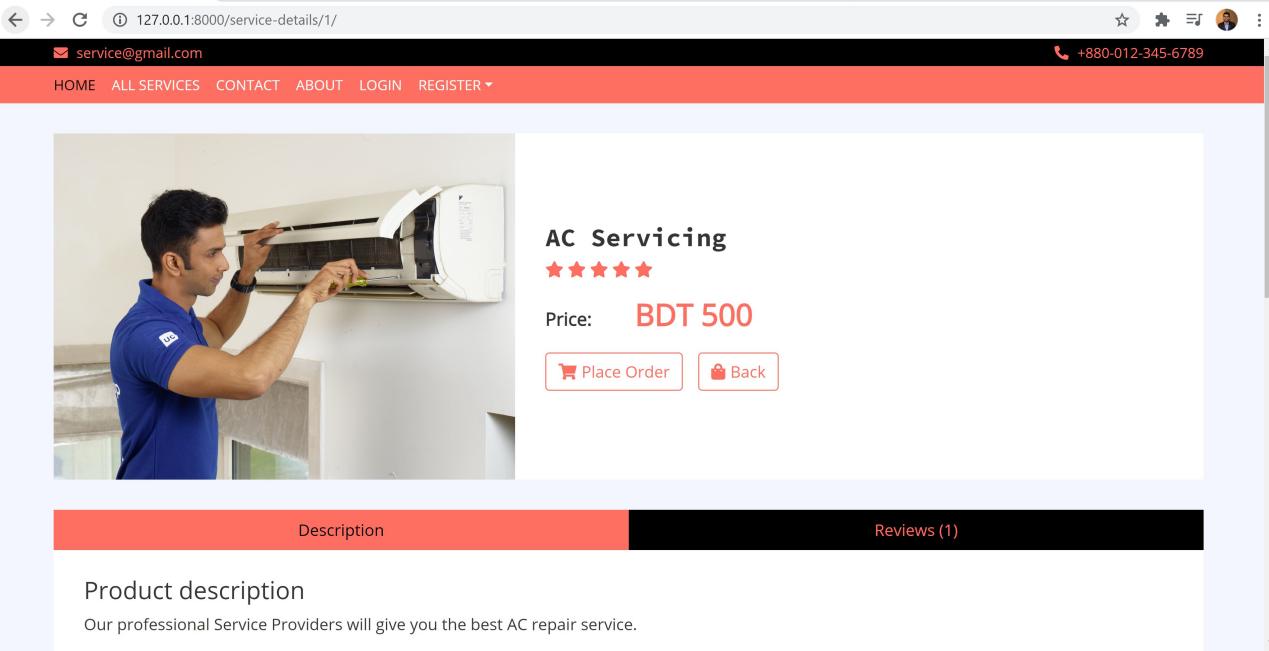


Fig:Service Details

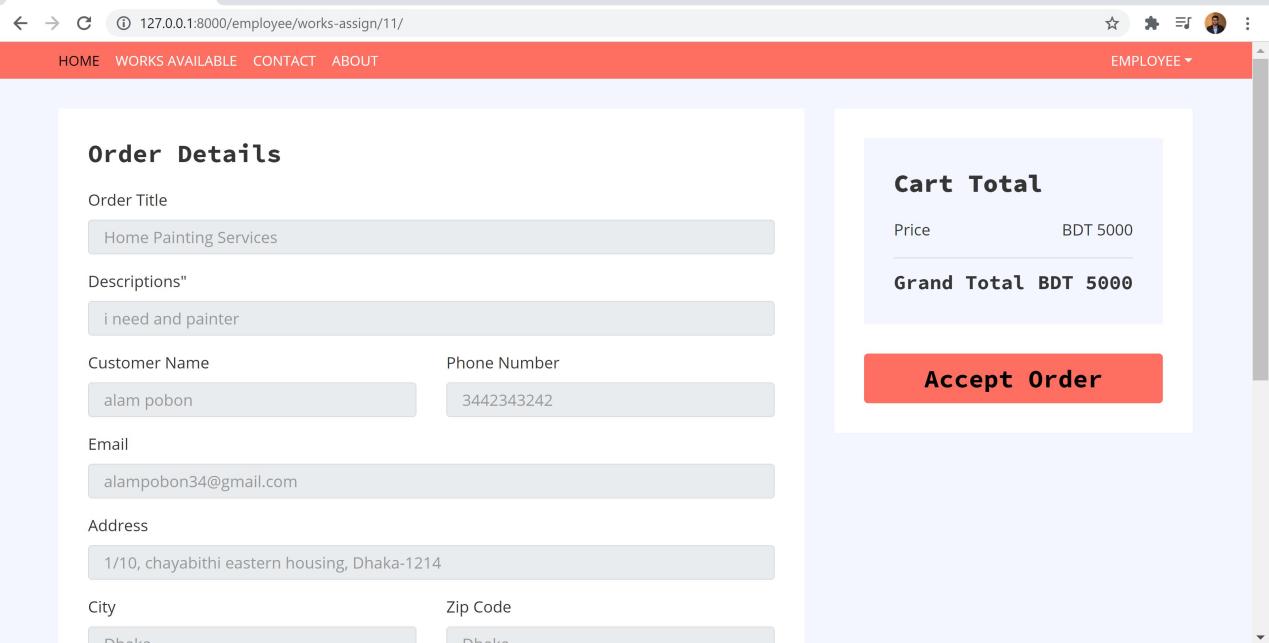


Fig: Work Assign

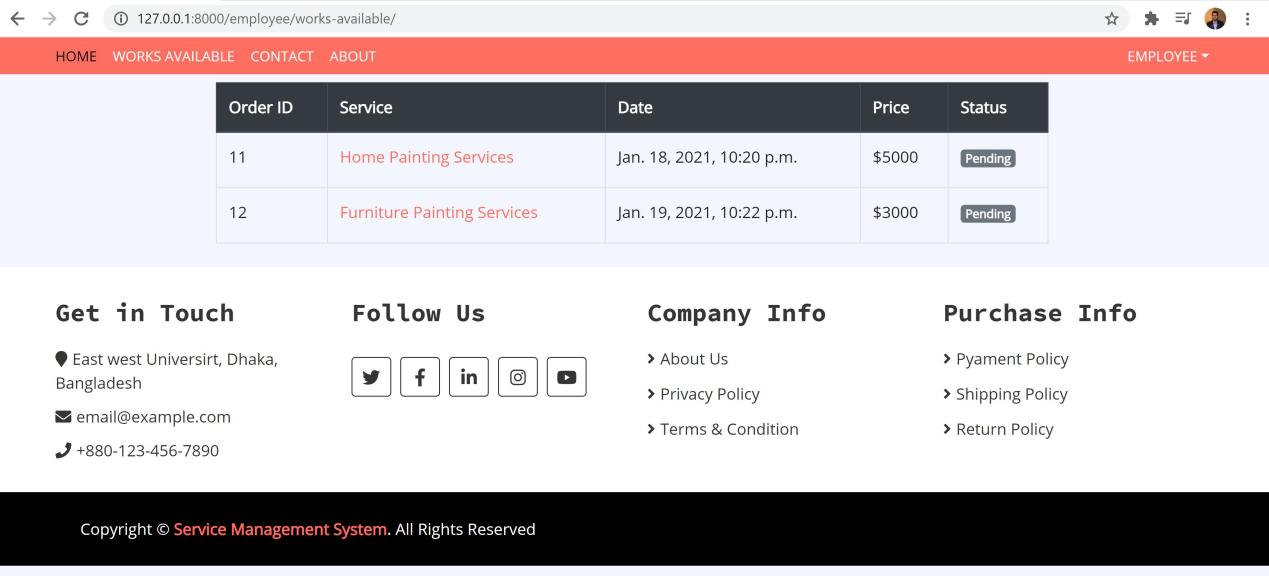


Fig: Works Available