North South University
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

PROJECT PROPOSAL

Neighbor - Network

We are for US

CSE 299: JUNIOR DESIGN

SECTION: 1

Fall 2022

Students Name	**ID**
Iftekher Mahbub Rafi	2021463642
Mariam Binte Bashir	2021874042
Tahmidul Alam	2021819642

Course Faculty Dr. Sifat Momen

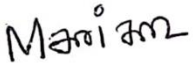
Date of Submission 18 December 2022

Table of Contents

Declaration

We, hereby, declare that the work presented in this report is the outcome of the 6 weeks work performed by us with the feedback of Dr. Sifat Momen, Associate Professor, Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh. The work was spread over a period of a course CSE299 (Junior Design Project) in accordance with the course curriculum of the department for the Bachelor of Science in Computer Science and Engineering program.

Students name and Signature:



Mariam Binte Bashir
ID:202 1874 042
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Iftekhar Mahbub Rafi
ID:202 1463 642
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Tahmidul Alam
ID: 202 1819 642
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Approval

The junior project report on “Neighbor-Network” has been submitted by Mariam Binte Bashir (ID: 202 1874 042), Iftekhar Mahbub Rafi (ID: 202 1463 642) and Tahmidul Alam (ID: 202 1819 642), students of Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh. This report partially fulfills the requirement for the degree of Bachelor of Science in Computer Science and Engineering in Summer 2022 and has been accepted as satisfactory.

Supervisor’s Signature:

Dr. Sifat Momen
Associate Professor
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Department Chairman’s Signature:

Dr. Rajesh Palit
Professor and Chair
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

Acknowledgement

All praises are due to Almighty Allah, Most Gracious, Most Merciful who blessed us to be here at North South University for pursuing the bachelor's degree.

We are grateful and wish our profound indebtedness to Dr. Sifat Momen, Associate Professor, Department of Electrical and Computer Engineering (ECE), North South University, Dhaka. We are blessed with the deep knowledge and keen interest of our supervisor in the field of design to carry this project. His guidance, constant supervision, enthusiastic encouragement, sagacious advice and an effective surveillance throughout the entire period of the project have made it possible to complete this project.

We would like to express deepest appreciation towards Dr. Rajesh Palit, Professor and Chair of Electrical and Computer Engineering Department, North South University, Dhaka, Bangladesh.

We would also like to thank each and everyone who were involved in the project from our friends, faculty members, fellow classmates, juniors and seniors who took the time to help and guide us during the span of the project.

We would also like to thank our family members, our parents, brothers and sisters who are the the very reason of our existence. Without their unconditional love and support, this project could not be completed.

Iftekhar Mahbub Rafi
Mariam Binte Bashir
Tahmidul Alam

Abstract

One who lives close by is referred to as a neighbor (or neighbor in American English), typically in a home or apartment that is next door or, in the case of homes, across the street [1]. A neighbor is always the first to come to our rescue in times of need. Our neighbors will surely be the first to arrive if we ask for help in an emergency situation while we are at home. For instance, one or two neighbors may be doctors or nurses and might be of great help in giving the sick person urgent treatment. The same is true for community-based technicians, locksmiths, and police officers. They could also lend us supplies or equipment in an emergency. But it's not always possible.

People usually struggle while trying to get help in an emergency. It is quite challenging to get any emergency assistance, such as for health-related difficulties, door locks or other home problems, mechanical issues, etc., due to the present traffic scenario. We created a web application to help people get emergency assistance within a 20-minute walking radius in order to solve the issue of urgent requirements. Depending on their expertise, people will be able to use this online application to ask their neighbors for main support. People who live in connected communities support one another, provide reliable information that is simple to implement, and frequently encourage one another to change their lifestyle choices. Keeping in touch with our neighbors on a social level may be beneficial in many ways.

To connect customers with service providers across different industries and make our service accessible to the largest possible audience, we want an intuitive and straightforward online application. To address this dynamic issue, full-stack web application development will be required, as well as the use of modern languages and scripts.

Introduction

1.1 Identification of the problem

A Neighbor (or neighbor in American English) is a person who lives nearby, normally in a house or apartment that is next door or, in the case of houses, across the street [1]. In any crisis, a neighbor is the first to rush to our aid. If we are in an emergency scenario in our homes and call for assistance, our neighbors will undoubtedly be the first to respond. For example, one or two neighbors may be physicians or nurses and so be of tremendous assistance in providing emergency care to that sick individual. The same is true for technicians, locksmiths, and police officers who live in communities. In an emergency, they may also lend us equipment or supplies. However, it is not always feasible to determine who among you has any medical and/or technological abilities that may be useful in an emergency. So, we will design a solution that will keep them linked and assist individuals in emergency circumstances from their neighbors.

1.2 Importance

People always have difficulty seeking any aid in an emergency. Because of the current traffic situation, it is extremely difficult to obtain any emergency assistance, such as health-related concerns, door lock or other domestic issues, mechanical issues, and so on. To address the issue of emergency needs, we developed a web application that will assist in obtaining emergency assistance within a 20-minute walking distance. People will be able to seek primary assistance from their neighbors through this web application, depending on their experience. People who live in linked communities help one another, give trustworthy information that is easy to adopt, and regularly motivate one another to modify their living habits. Maintaining social contact with our neighbors can help us in a number of ways [2]. One of the many unexpected benefits of building a strong community is that it makes finding the appropriate individuals for the job simpler. Building a strong community offers a plethora of unanticipated good consequences [3]. As a result, recruiters should encourage and foster strong ties with community professionals on their teams.

1.3 Our Approach

We need an easy-to-use and simple web application to link clients with service providers of various vocations and make our service available to the biggest number of people. We will need full-stack web application development to handle this dynamic challenge, which will need the usage of contemporary languages and scripts.
The graphic below shows how this method works:

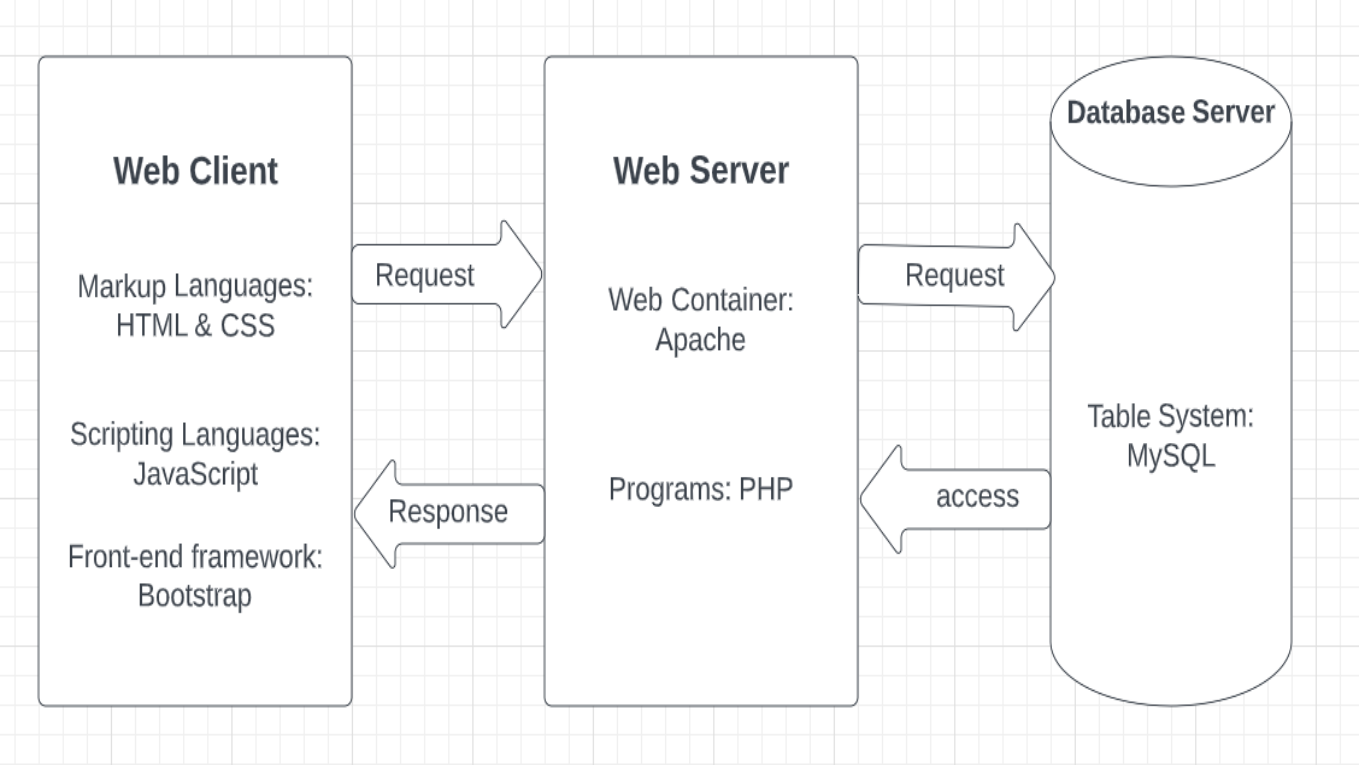


Figure1.1: A typical structure

Project Specifications:

Users and Their Roles:

User:

The user will sign up for the system by entering pertinent information such as their name, email, phone number, location, and date of birth. When they join up, they will be assigned a unique user id (UID). Signing can also be done with a Google account. They will then be required to authenticate their identity before delivering or obtaining services.
To prove their identity, each user must go through a verification procedure that includes submitting their NID, worker permit/license, testimonial, and referral system.
Verified users will be assigned two roles: one as a customer and one as a service provider. They will be able to seek assistance or services as a customer. In the same way, a service provider will be allowed to accept the request.

Admin:

Admin will have full access to administer the system, including the ability to change a user's information, verify documents, suspend an account, and contact local police authorities based on an alert system.

Use Cases of the System:

The different use cases are grouped according to their actors below:

User:

1. **Sign up/Register:** Users may sign up for the system by entering information such as their name, email address, phone number, location, gender, date of birth, and password. Each user will be assigned a unique id (UID) when they register, which will be used to access their profile and edit other information. They may also join up with their Gmail id.

2. **Login:** After registering, the user may log in to the system with their UID and password.

3. **Logout:** A user can log out from the system anytime.

4. **Verification:** To protect the safety of the consumer, each user will go through a thorough verification procedure. Each user must submit their NID in order to utilize or supply services. If a person chooses to become a service provider once their NID has been verified, they must indicate their expertise/occupation, give their license/worker permit, and offer a testimonial from their current employer. There will also be a recommendation system via which existing service providers can suggest other users who want to become service providers.

5. **Profile:** Each user will have a profile page with information such as their name, phone number, and email address. In the case of a service provider, it will display a history of all services performed up to that point, complete with evaluations and comments.

6. **Update Profile:** A user's profile may be updated to modify information such as their name, email address, phone number, current location, and password.

7. **Request Help (User as a Client):** When a user requires a service, he or she must specify the service by selecting the appropriate profession. The system will then provide a list of all nearby service providers depending on their distance from the user.

8. **Notification:** When a user requests a certain service, all neighboring service providers in that field are notified. If a service provider accepts that service, the user will receive a message with a list of all providers ready to deliver it. When a customer selects a certain service provider via confirmation, he/she (the service provider) will receive a communication.

9. **Review & Comment (User as a client):** Once a service has been completed, the customer will be able to provide feedback in the form of Ratings and Comments.

10. **Safety Button:** Clients will have a safety button that, if a service provider perceives a threat while delivering service, they will be able to click, which will summon local law enforcement and also inform the admin.

11. **Refer (User as a service provider):** A user must be suggested by an existing service provider in order to be confirmed as a service provider. As a result, an existing service provider will be able to suggest a new user.

12. **Expertise (Only for Service Provider):** A user who wishes to become a service provider must declare their area of expertise/occupation.

13. **Accept Request (Only for Service Provider):** When a service provider receives a notification for a nearby service, he or she might choose to accept the request.

14. **Nearby Service Provider Location in Map (user as a client):** When a user requests a certain service, he or she will be presented with a list of all locally specified service providers.

15. **Confirmation (User as a client):** When a user receives a list of all local service providers, he or she will have the opportunity to confirm which service provider to select.

16. **Service Received (User as a client):** When a client receives their requested service, they must acknowledge via a service received button, which will activate the review and remark area.

17. **Report (User as a client):** If a user encounters any inappropriate activity, he or she has the option of reporting it to the service provider.

18. **Delete:** A user will be able to delete their account.

Admin:

1. **User information update:** An administrator will be able to update a user's information.
2. **Account verification:** Once a user's document has been validated, an administrator will enter that confirmation into the system.
3. **Account suspension:** Based on customer reports, an administrator will have the authority to suspend an account.

Planning:

Gantt Chart:

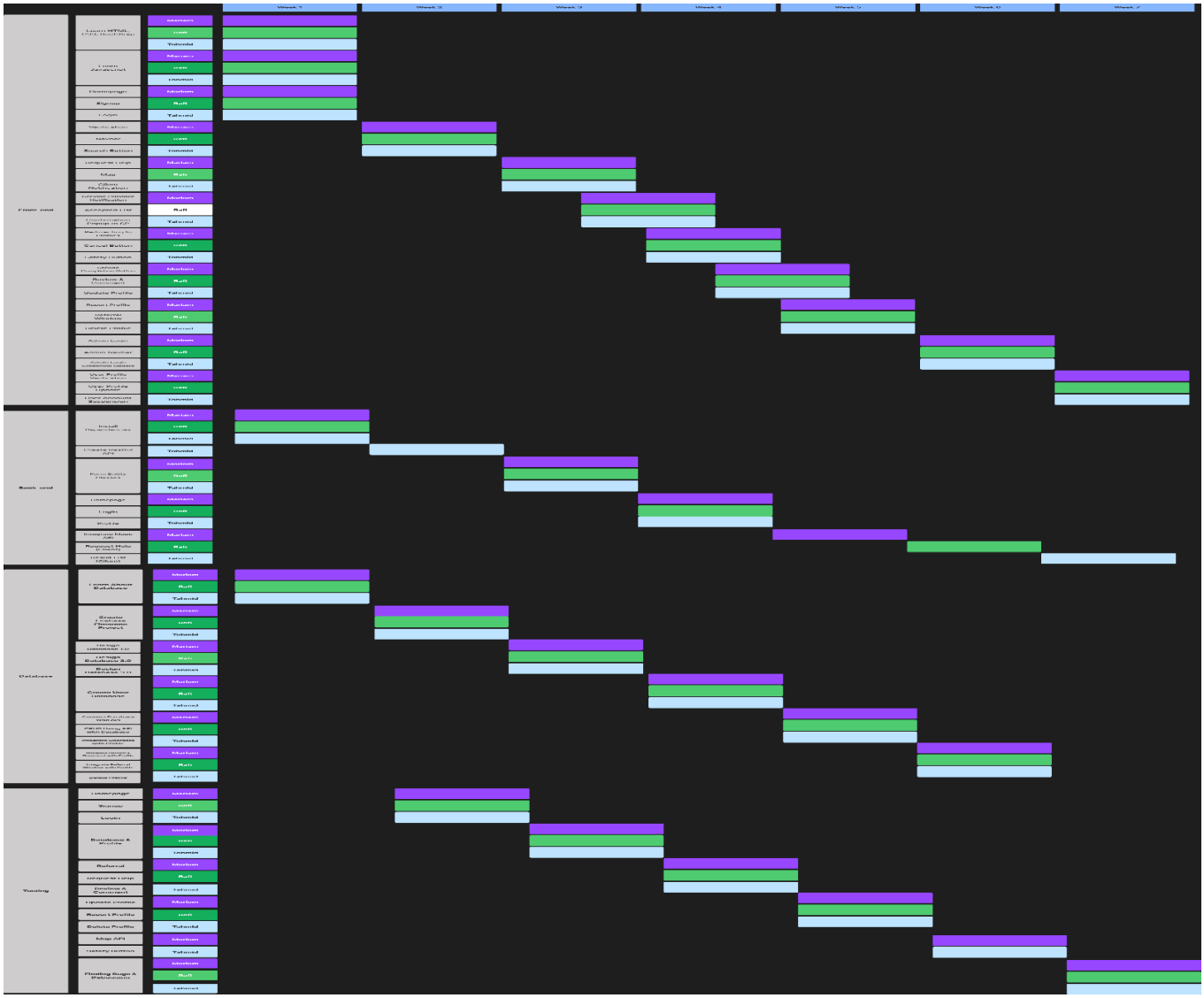


Figure 2.1: Gantt Chart

Design

Front-end

The front-end is designed for the users to easy interact with the application. Here is the estimated design for the user interface.

Home

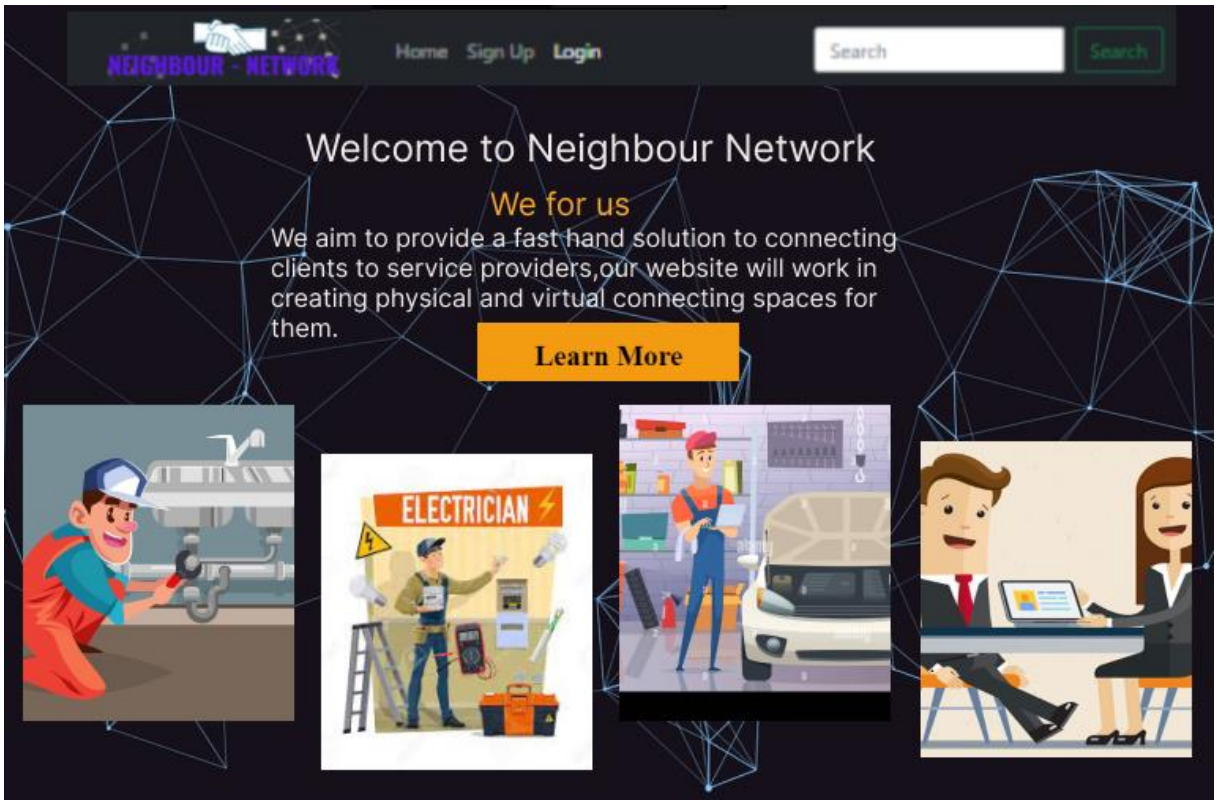


Figure 3.1: Landing Page

Signup

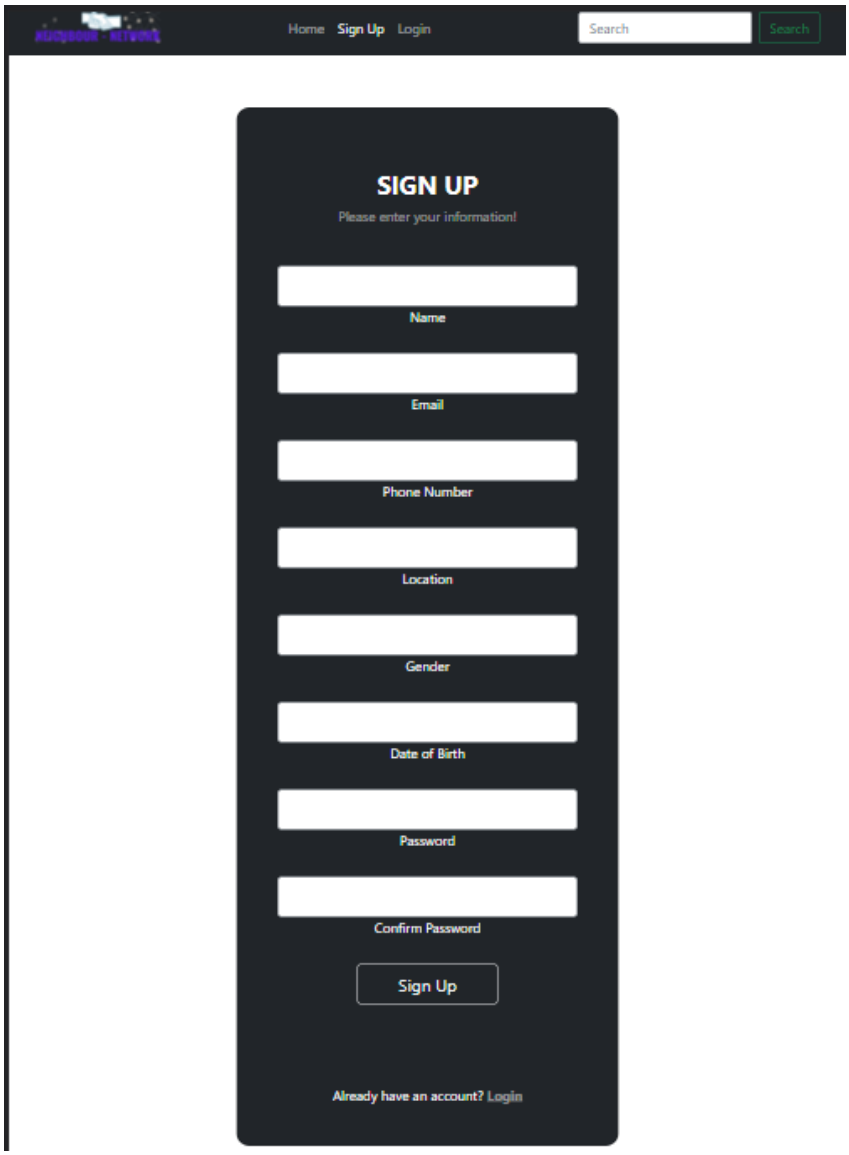


Figure 3.2: Signup Page

Login

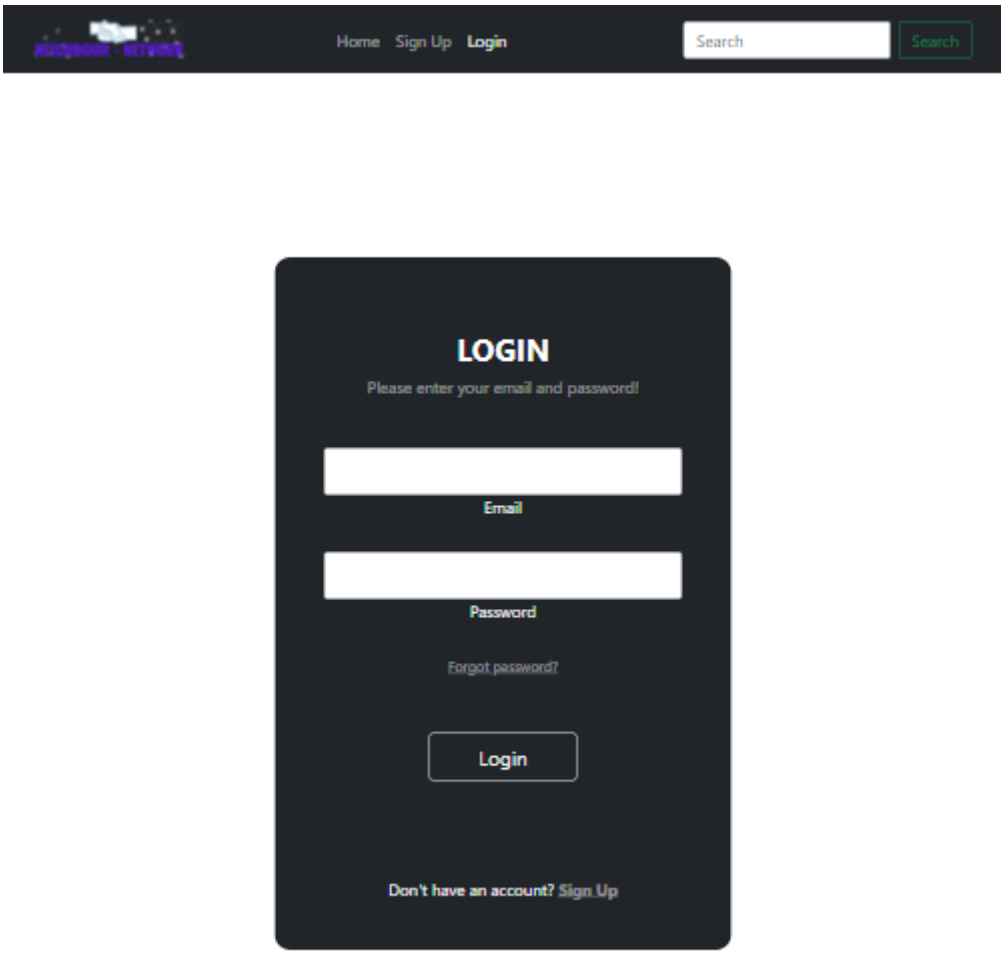


Figure 3.3: Login Page

Profile

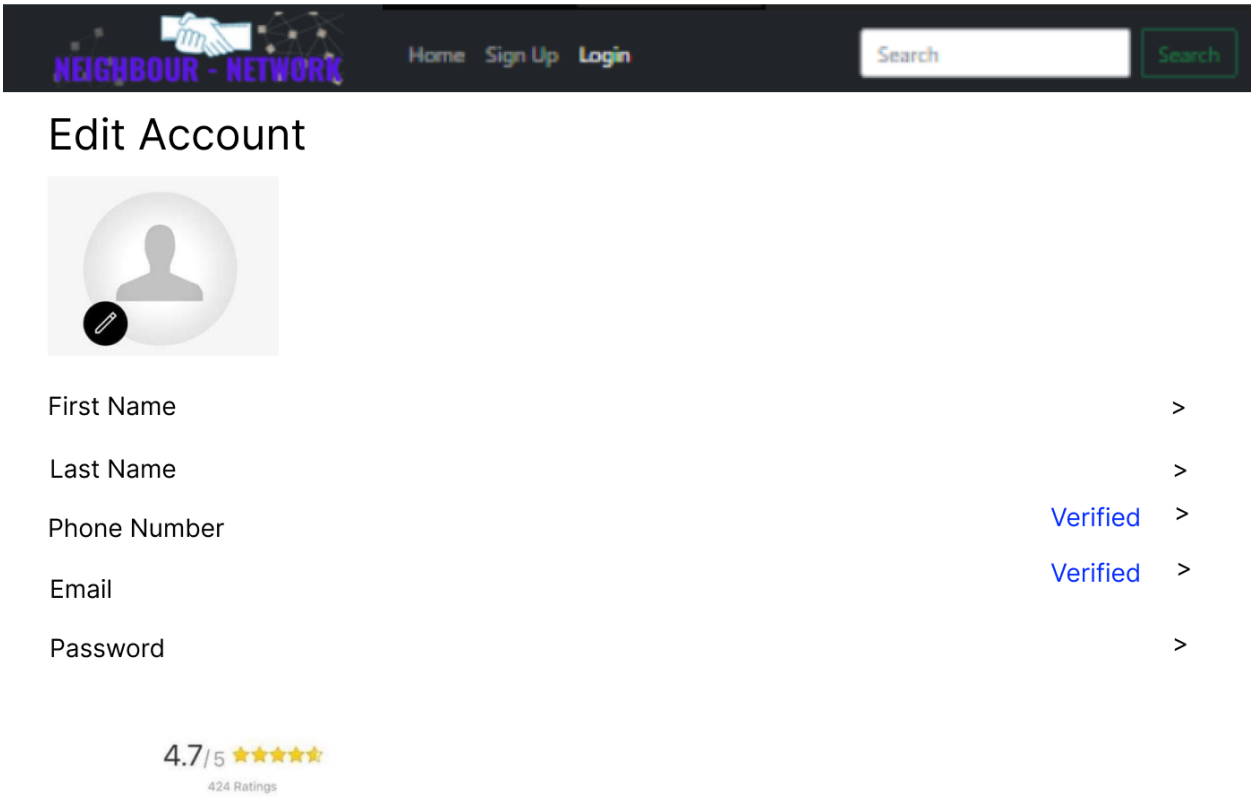


Figure 3.4: Users Profile

Verification

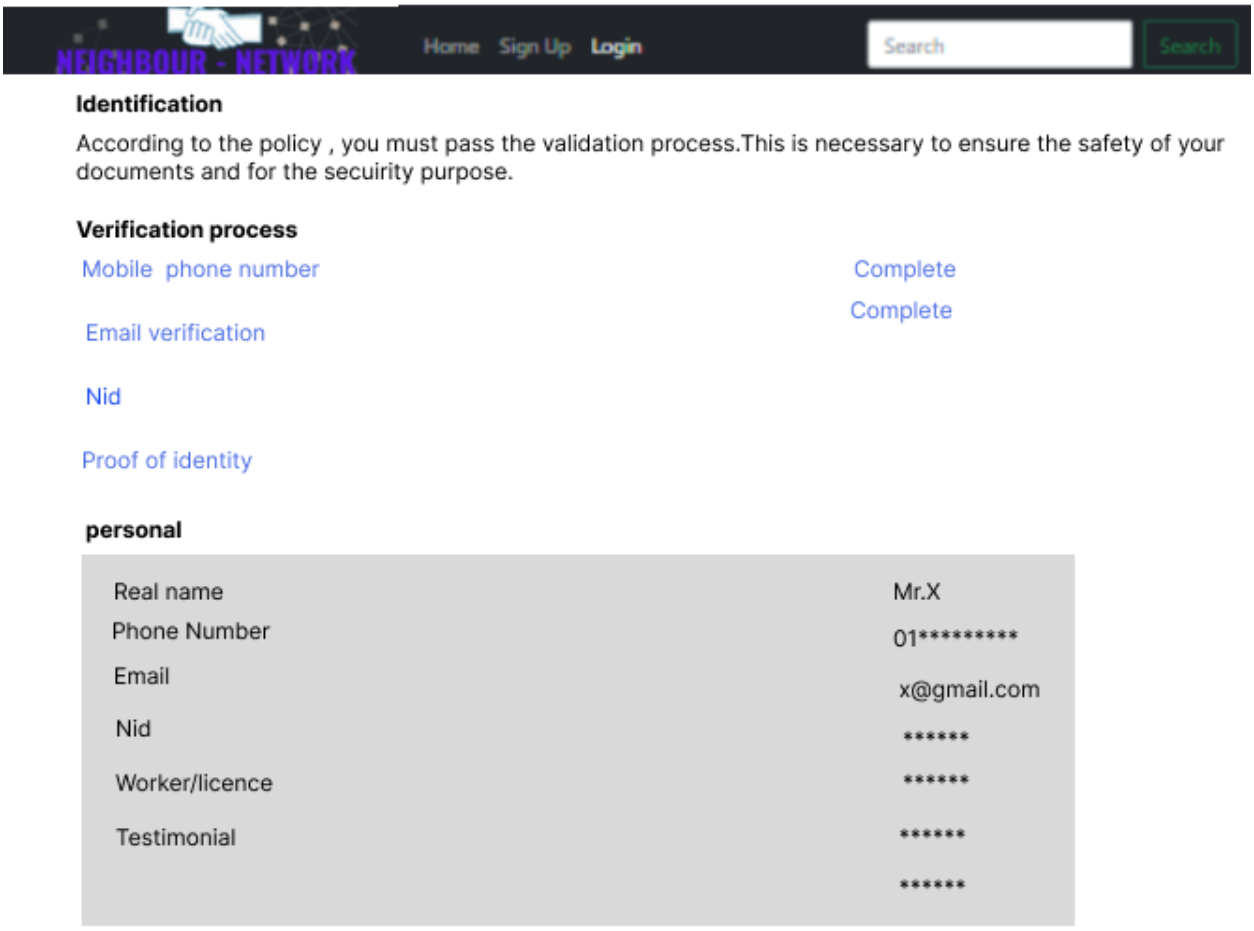


Figure 3.5: Users Verification Page

Map

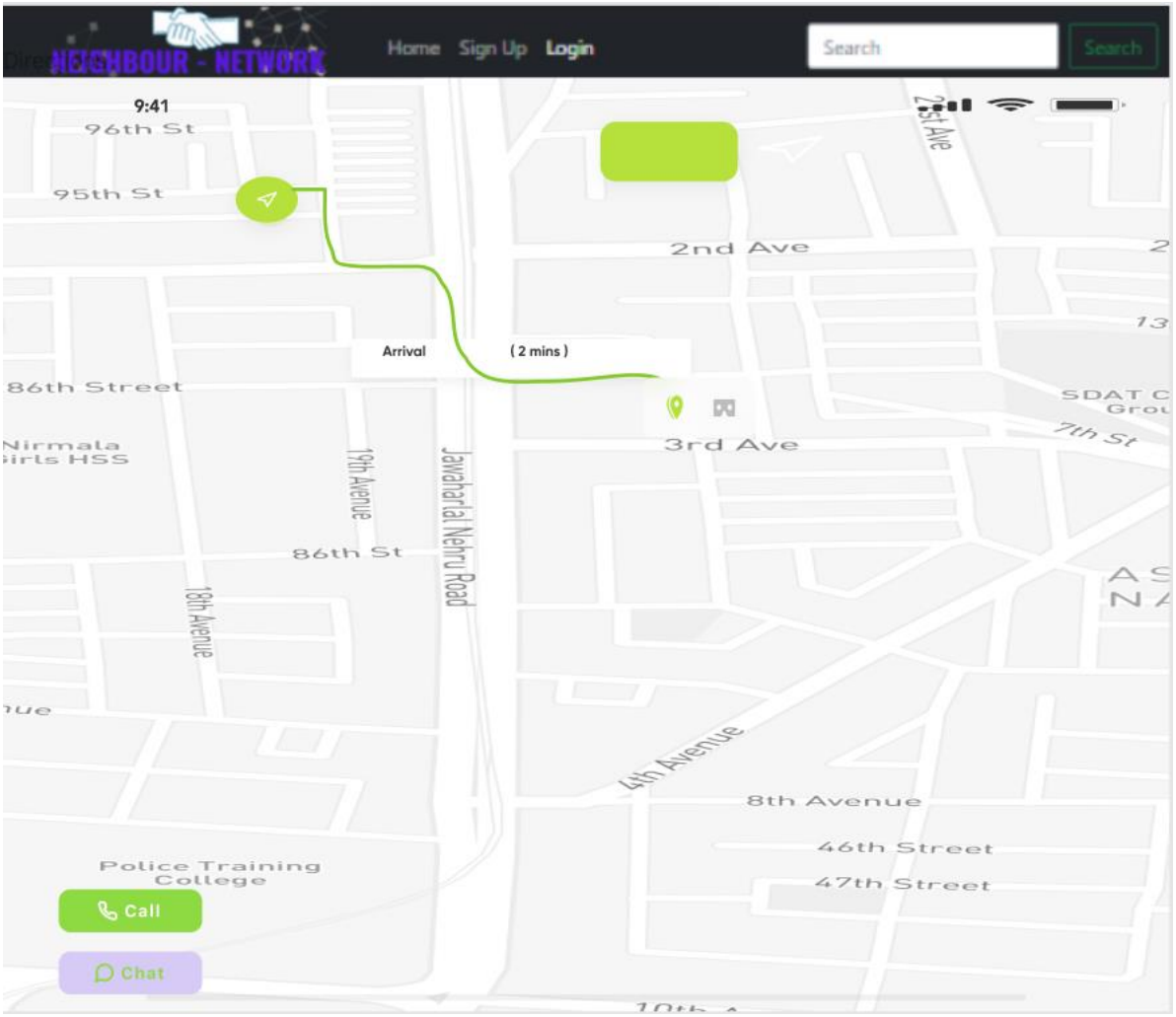


Figure 3.6: Map View for users when taking services.

Backend

The backend design is done as following manner:

Database: ER Diagram

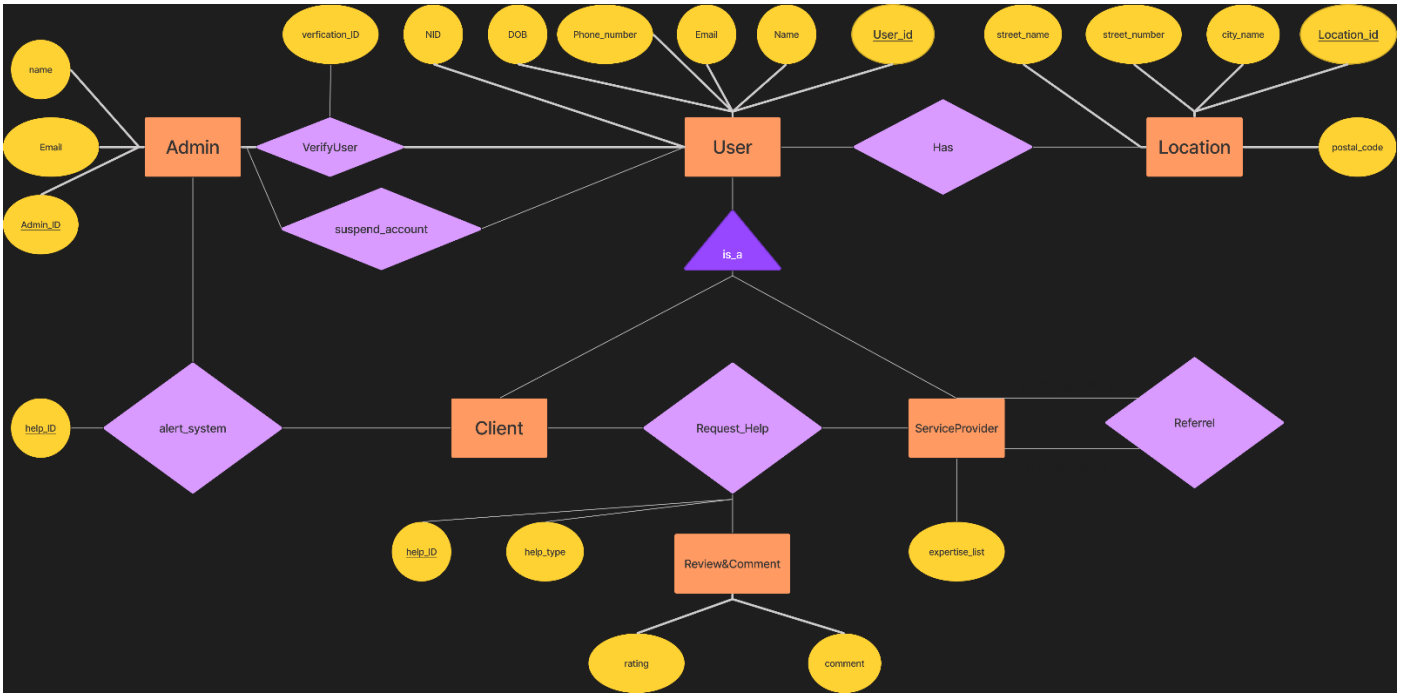


Figure 3.7: ER Diagram

Relational Table: (Normalized Form)

Figure 3.8: Relational Table

Implementation

Front-end:

Markup Language: The markup language for the front end was Hypertext Markup Language (HTML). HTML is the industry standard for document presentation in web browsers. It was rendered inactive with the aid of other technologies like JavaScript (JS) and Cascading Style Sheets (CSS).

Styling: Raw CSS (Cascading Style Sheets) and Bootstrap 5 were used to style the markup language that was rendered. Along with HTML and JavaScript, CSS is a fundamental component of the World Wide Web.

A free and open-source CSS framework called Bootstrap is designed for front-end web development that prioritizes mobile compatibility and responsiveness. It includes design templates for typography, forms, buttons, navigation, and other interface elements that are based on HTML, CSS, and (optionally) JavaScript. To give our front end a traditional appearance, the markup language was stylized and wrapped with CSS and JavaScript.

Backend

Content Management System: WordPress was utilized as the content management system and for the backend. Free open-source WordPress is used in conjunction with a MYSQL database that supports HTTPS.

Additionally, WordPress provides a REST API that enables us to interact with the program without actually needing to open it in a web browser.

Figure: Backend Implementation.

Database

The MySQL database management system is used by WordPress. Because it integrates with other open-source programs, this relational database management system (RDBMS) is the most often used choice for building databases. All of this data is kept in the database in tables, rows, and columns.

The free WordPress plugin WP-DB Manager keeps track of all of your data and records for each individual under one roof. It enables you to repair, backup, restore, and optimize databases before executing the chosen website queries.

Connection

Numerous PHP, JavaScript, and other frameworks are supported by WordPress for both front-end and back-end applications. The front-end development frameworks for creating APIs, plugins, and themes include AngularJS, Vue JS, Bootstrap, and React JS, among others.

Endpoints are operations accessible via the API. Such actions include getting the API index, editing a post, and removing comments. Endpoints carry out a specified task while requiring a set of parameters and sending data back to the client.

Deployment

The web application was hosted and deployed online on a web server. The registered url for our website is: https://nsuteam3.com/

Testing:

Software testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. The benefits of testing include preventing bugs, reducing development costs and improving performance. A good testing approach encompasses the application programming interface (API), user interface and system levels. As well, the more tests that are automated, and run early, the better. Some teams build in-house test automation tools. However, vendor solutions offer features that can streamline key test management tasks such as, continuous testing Project teams test each build as it becomes available. This type of software testing relies on test automation that is integrated with the deployment process. It enables software to be validated in realistic test environments earlier in the process – improving design and reducing risks. By Defect Monitoring defects is important to both testing and development teams for measuring and improving quality. Automated tools allowed us to track defects, measure their scope and impact, and uncover related issues. Reporting and analytics

enable team members to share status, goals and test results. Also, we used selenium for testing and verifying. It supports Functional Testing at the System Testing Level and the Acceptance Testing Level of the Software Development Life Cycle".

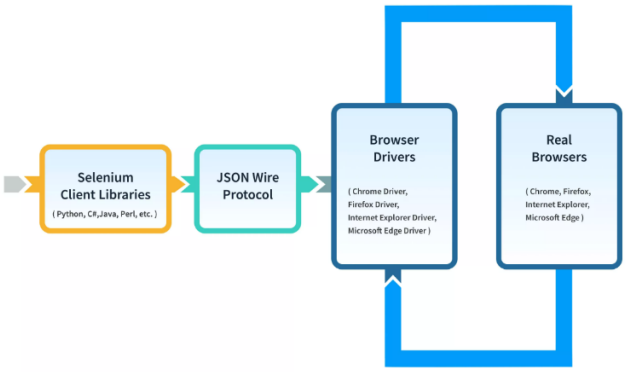


Figure: Selenium Testing

Maintenance plan:

Software maintenance is the process of changing, modifying, and updating software to keep up with customer needs. Software maintenance is done after the product has launched; for several reasons including improving the software overall, correcting issues or bugs, to boost performance, and more. The four types of maintenance techniques typically followed are: Corrective Software Maintenance, Preventative Software Maintenance, Perfective Software Maintenance and Adaptive Software Maintenance.

The combination of these four maintenance techniques gives us seven critical steps in maintenance which are targeted to be followed in the future:

Corrective Maintenance: In this task are including updating, modification, fixing the errors time to time while it's by the user or concluded by user report.

Adaptive Maintenance: In this, we modify the software product after the delivery for the better performance and maintainability.

Perfective Maintenance: We did this for adding new features, user new requirements and it also for the improving the performance.

Preventive Maintenance: In this, we prevent our software from errors or in future problems of software. This has main aim to prevent software time to time.

Figure: Maintenance Techniques

The combination of these four maintenance techniques gives us seven critical steps in maintenance which are targeted to be followed in the future:

- **Identification and Tracing:** The process of determining what part of the software needs to be modified (or maintained). This can be user-generated or identified by the software developer itself depending on the situation and specific fault.

- **Analysis:** The process of analyzing the suggested modification including understanding the potential effects of such a change. This step typically includes cost analysis to understand if the change is financially worthwhile.

- **Design:** Designing the new changes using requirement specifications

- **Implementation:** The process of implementing the new modules by programmers.

- **System Testing:** Before being launched, the software and system must be tested. This includes the module itself, the system and the module, and the whole system at once.

- **Acceptance Testing:** Users test the modification for acceptance. This is an important step as users can identify ongoing issues and generate recommendations for more effective implementation and changes.

- **Delivery:** Software updates or in some cases new installation of the software. This is when the changes arrive at the customers.