

# **MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR**

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)



**Final Year Internship Report**

**On**

**RESIDENTIAL SOCIETY MAINTENANCE PORTAL**

**Submitted By:**

**Krishnakant Doneriya**

**0901CS181056**

**Faculty Mentor:**

**Dr. Ranjeet Kumar Singh**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE**

**GWALIOR - 474005 (MP) est. 1957**

**MAY-JUNE 2022**



# **MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR**

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)



## **RESIDENTIAL SOCIETY MAINTENANCE PORTAL**

A final year internship report submitted in partial fulfilment of the requirement for the

degree of

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE AND ENGINEERING**

Submitted by:

**Krishnakant Doneriya**

**0901CS181056**

Internship Faculty Mentor:

**Dr. Ranjeet Kumar Singh**

Submitted to:

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

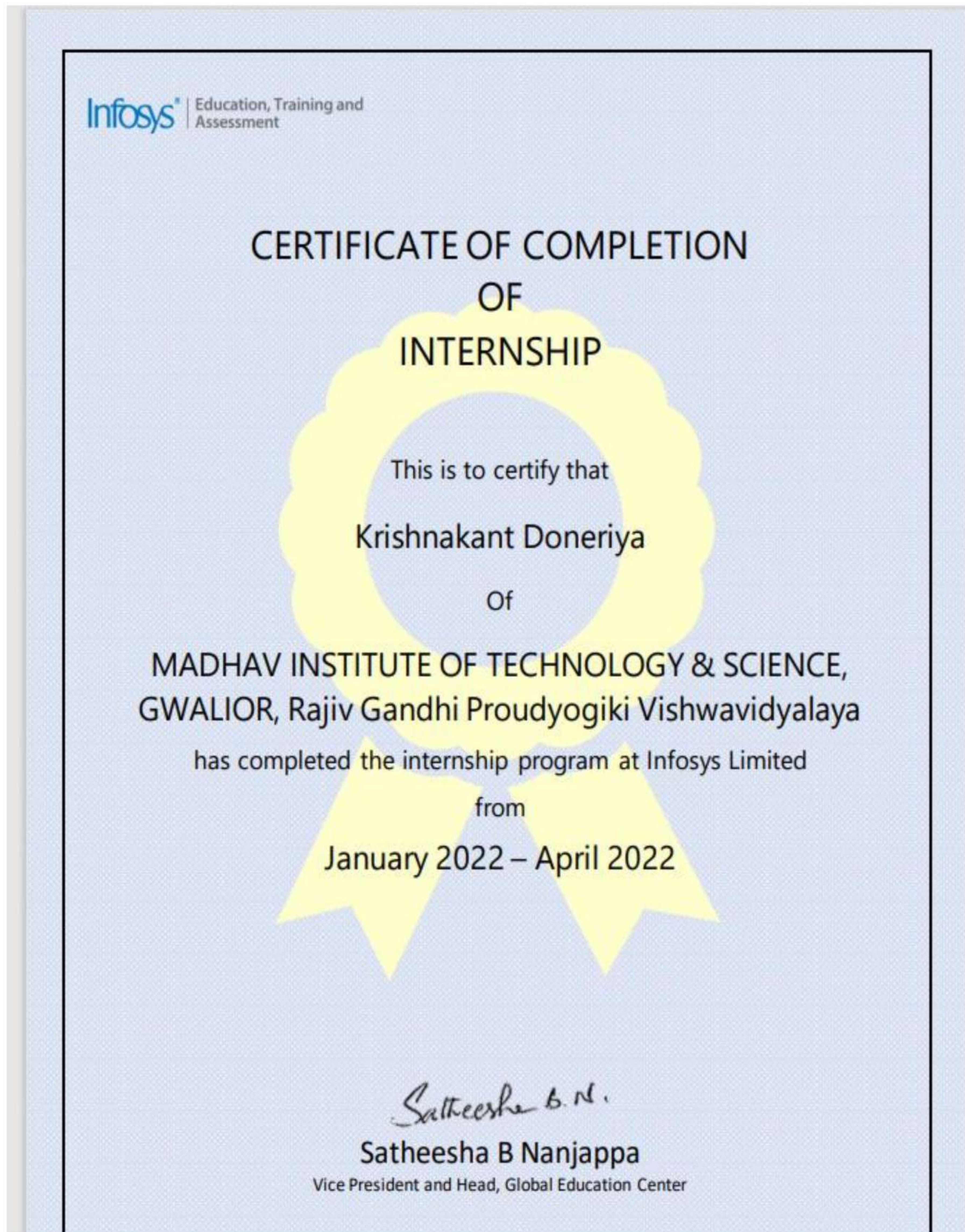
**MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE**

**GWALIOR - 474005 (MP) est. 1957**

**MAY-JUNE 2022**



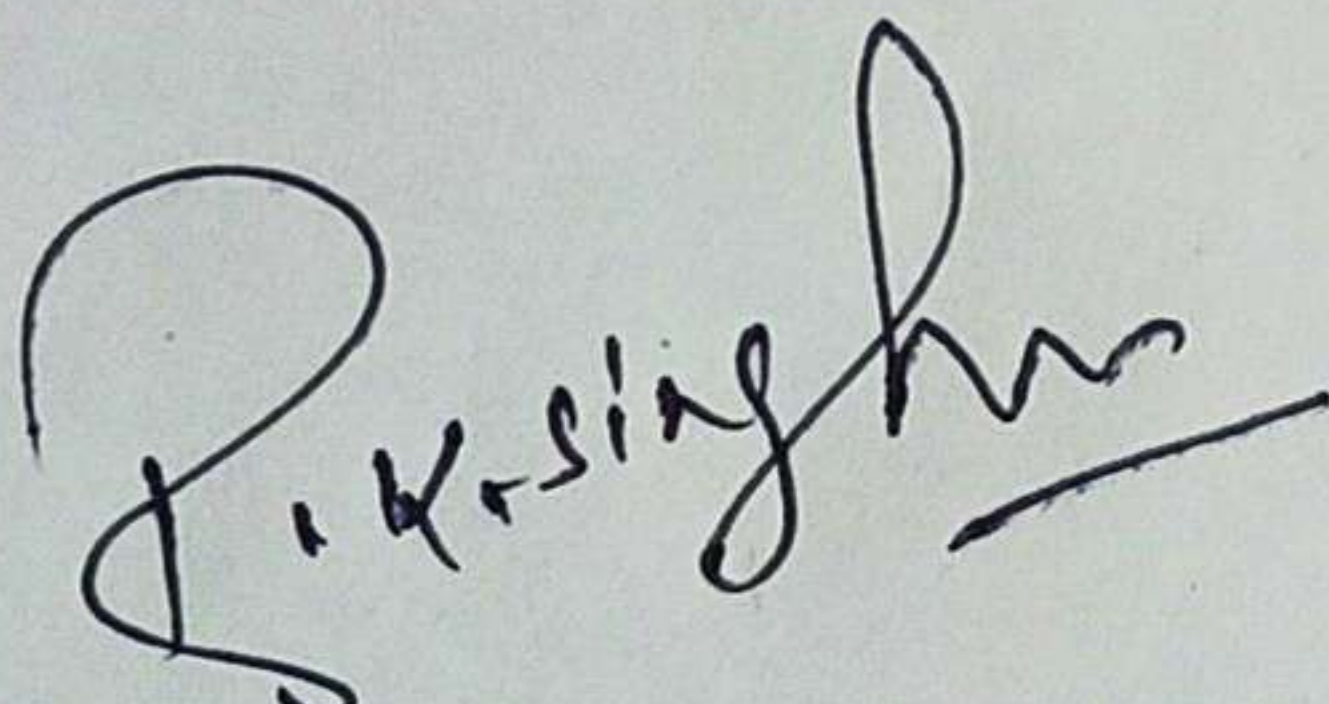
Internship Certificate Received  
from  
**Infosys Private Limited, Mysore**



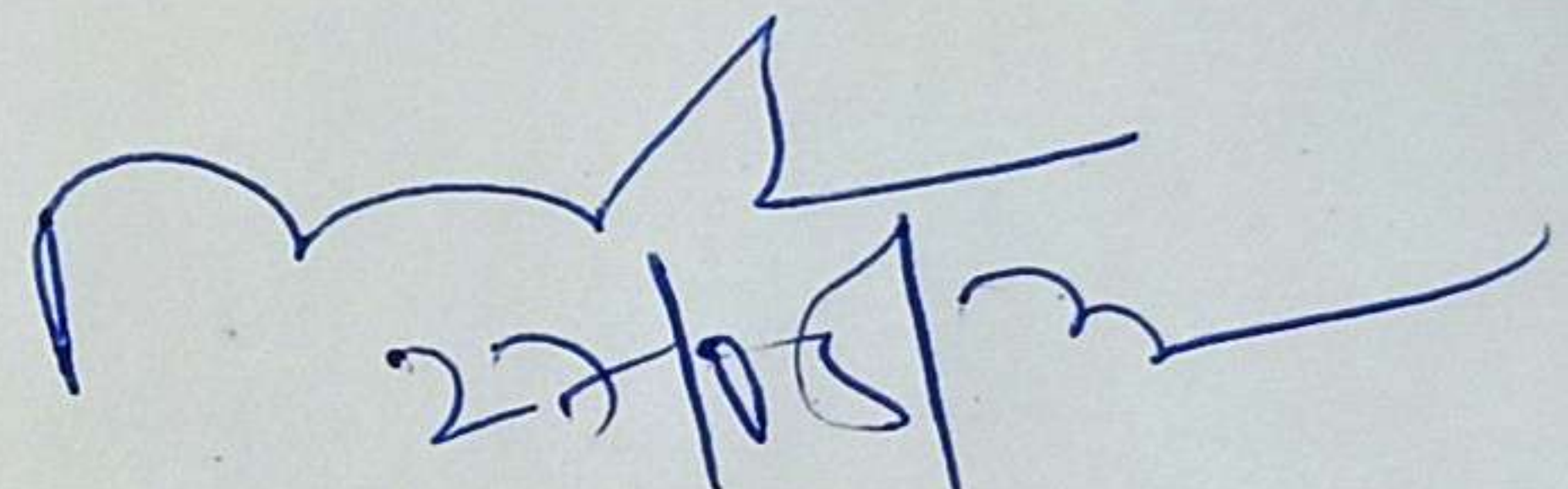


## CERTIFICATE

This is certified that **Krishnakant Doneriya 0901CS181056** has submitted the Internship report titled "**RESIDENTIAL SOCIETY MAINTENANCE PORTAL**" of the work he has done under the mentorship of **Dr. Ranjeet Kumar Singh**, in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.



**Dr. Ranjeet Kumar Singh**  
Faculty Mentor  
Computer Science and Engineering



**Dr. Manish Dixit**  
Professor & Head  
Computer Science and Engineering  
**Dr. Manish Dixit**  
Professor & HOD  
Department of CSE  
M.I.T.S. Gwalior



# **MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR**

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

## **DECLARATION**

I hereby declare that the work being presented in this Internship report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Dr. Ranjeet Kumar Singh**, Department of Computer Science and Engineering.

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Krishnakant Doneriya

0901CS181056

IV Year

Computer Science and Engineering

# **MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR**

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

## **ACKNOWLEDGEMENT**

The full semester internship has proved to be pivotal to my career. I am thankful to my institute, **Madhav institute of technology and science** to allow me to continue my disciplinary or interdisciplinary internship as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Computer Science and Engineering, for allowing** me to explore this internship. I humbly thank **Dr. Manish Dixit**, Professor and Head, Department of Computer Science and Engineering, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Dr. Ranjeet Kumar Singh**, Department of Computer Science and Engineering, for her continued support and close mentoring throughout the internship. I am also very thankful to the faculty and staff of the department.

Krishnakant Doneriya

0901CS181056

IV Year

Computer Science and Engineering



# ABSTRACT

This project report describes our work for the development of “Residential Society Maintenance Portal”, an enterprise application. An Enterprise Application is a software application that helps enterprises in managing their business-related activities. They are large and complex and solve business-related problems to fulfill the needs of the enterprises. The existing system is well and good also, but. they had some less features, so we want to provide the best option in the website. As the title suggests this is a online website which provide the platform, that any resident can be log their household maintenance request online on this website. The objective of our project is to provide a better user platform to the residential society to maintain the society activities. We surely provide some basic functionalities as well as more good option in this to improve the experience of user.

## **About the Organization:**

This Internship is a Java Full Stack Development Training By *Infosys Private Limited, Mysore* using spring and Angular as Frameworks. This Training is followed by a Project in which we have to implement what we’ve been taught throughout this Training.

## **Objective of Proposed Work:**

As the title suggests this is a online website which provide the platform, that any resident can be log their household maintenance request online on this website. The objective of our project is to provide a better user platform to the residential society to maintain the society activities. We surely provide some basic functionalities as well as more good option in this to improve the experience of user.

.

.

# TABLE OF CONTENTS

<b>TITLE</b>	<b>PAGE NO.</b>
<b>Introduction</b>	<b>i</b>
<b>Internship Certificate from Industry</b>	<b>iii</b>
<b>Institute Internship Certificate</b>	<b>iv</b>
<b>Declaration</b>	<b>v</b>
<b>Acknowledgement</b>	<b>vi</b>
<b>Abstract</b>	<b>vii</b>
<b>Table of Contents</b>	<b>viii</b>
<b>List of Figures</b>	<b>x</b>
<b>Abbreviation</b>	
<b>Chapter 1: Introduction</b>	<b>1</b>
<b>1.1 Objective</b>	
<b>1.2 Scope</b>	
<b>1.3 Problems in Existing System</b>	
<b>1.4 About this website</b>	
<b>Chapter 2: System Requirement</b>	<b>2</b>
<b>2.1 Information Gathering</b>	
<b>2.2 System Feasibility</b>	
<b>2.2.1 Software Implementation Technology</b>	
<b>Chapter 3: System Design</b>	<b>4</b>
<b>3.1 Architectural Design</b>	
<b>3.2 Modules Used</b>	
<b>3.2.1 Internal Data Structure</b>	
<b>3.3 Interface Design</b>	



## **Chapter 4: Implementation And Testing**

**15**

### **4.1 Implementation**

#### **4.1.1 Tools Used**

#### **4.1.2 Description of Main Module**

### **4.2 Testing**

## **Chapter 5: Conclusion and Future Scope**

**21**

### **5.1 Conclusion**

### **5.2 Future**

### **5.3 References**



## **LIST OF FIGURES**

<b>Figure Number No.</b>	<b>Figure caption</b>	<b>Page</b>
<b>1</b>	<b>Project Home Page</b>	<b>5</b>
<b>2.</b>	<b>Admin-Login Page</b>	<b>6</b>
<b>3.</b>	<b>Admin-Home Page</b>	<b>6</b>
<b>4.</b>	<b>Resident- Login Page</b>	<b>9</b>
<b>5.</b>	<b>Resident –Sign Up Page</b>	<b>10</b>
<b>6.</b>	<b>Resident –Home Page</b>	<b>10</b>
<b>7.</b>	<b>Worker- Login Page</b>	<b>12</b>
<b>8.</b>	<b>Worker –Home Page</b>	<b>13</b>
<b>9.</b>	<b>Worker –Sign Up Page</b>	<b>14</b>



# **Chapter 1**

## **Introduction**

### **1.1 Objective**

Residential Society Maintenance Portal is an online platform where residents of a society can register/login and then log requests for any household related maintenance work e.g. water supply, electricity, sanitation, carpentry etc. The admin of the portal will be responsible for assigning these requests/tasks to one of the available workers. Once the maintenance work starts, the portal will also be responsible for providing the updates regarding status of the work.

There will be three types/roles of users who can use the portal i.e., RESIDENT, WORKER, ADMIN.

### **1.2 Scope**

Scope of the project is very vast. We can build an android app, so the user can easily log the household related maintenance work through app..

### **1.3 Problems in existing system**

There are some system available but they do not have such functionality. So we have given a facility to the user , that he/she can log their request .

### **1.4 About this Website**

Residential Society Maintenance Portal is an online platform where residents of a society can register/login and then log requests for any household related maintenance work e.g. water supply, electricity, sanitation, carpentry etc. The admin of the portal will be responsible for assigning these requests/tasks to one of the available worker



# CHAPTER 2

## System Requirement

### 2.1 Information Gathering

We have gathered such a lot of information from our side. We have checked existing website and there features.

The first and foremost method for gathering and managing our project information was to organizing team meetings. By conducting consistent weekly internal meet- ings with your project team, makes this more simple.

### 2.2 System Feasibility

Feasibility study of the system is a very important state during system design. Fea-sibility study is a test of a system proposal according to its workability impact on the organization, ability to meet user needs, and use of resources. Feasibility study decides whether the system is properly developed or not.

#### 2.1.1 Software implementation language/technology

##### For FrontEnd

The Technologies are

1. **Angular** : Angular (commonly referred to as "Angular 2+" or "Angular CLI") is a TypeScript-based free and open-source web application framework led by the Angular Team at Google and by a community of individuals and corporations. Angular is a complete rewrite from the same team that built AngularJS. Angular is used as the frontend of the MEAN stack, consisting of MongoDB database, Express.js web application server framework.

2. **Typescript** : Typescript lets you write JavaScript the way you really want to. Typescript is a typed superset of JavaScript that compiles to plain JavaScript. TypeScript is pure object oriented with classes, interfaces and statically typed like C or Java. The popular JavaScript framework Angular 2.0 is written in TypeScript. Mastering TypeScript can help programmers to write object-oriented programs and have them compiled to JavaScript, both on server side and client side.

3. **HTML**: Hypertext Mark-up Language (HTML) is the standard mark-up language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

4. **CSS** : Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and customer interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media.



**5. Bootstrap:** Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

## **For BackEnd**

The Technologies are

**1. SpringBoot :** Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run". We take an opinionated view of the Spring platform and third-party libraries so you can get started with minimum fuss. Most Spring Boot applications need minimal Spring configuration.

**2. Java:** Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need to recompile. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages.

**3. MySQL:** MySQL is an open-source relational database management system (RDBMS). MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python". Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress, Simple Machines Forum, phpBB, 3MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.



## Chapter 3

### System Design

#### 3.1 Architectural Design

Describe Architectural design for your project. The software needs the architectural design to represent the design of software. IEEE defines architectural design as “the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.” The software that is built for computer-based systems can exhibit one of these many architectural styles.

#### 3.1 Modules used

Describe various modules / procedures of your project specific system.

Main modules of project is :

- Sign up :- In this user can sign up and can able to feed details in the database like username, email, password.
- Login :- From this user can login in website by username and password at any time to avail more features of website..
- Worker Page:- This is the page where all workers can sign-up/ login and can see work allocated to them.
- Resident Page:- This is the page where all residents can sign-up/login and log new maintenance request.
- Admin Page :- This is the page where admin can manage all the request submitted by the residents and allocate them to the worker.

#### 3.1.1 Internal Data Structures

Data Structure used in our website :

ArrayList : ArrayList class uses a dynamic array for storing the elements. It is like an array, but there is no size limit. We can add or remove elements anytime. So, it is much more flexible than the traditional array. It is found in the java.util package. It is like the Vector in C++. The ArrayList in Java can have the duplicate elements also. It implements the List interface so we can use all the methods of the List interface here. The ArrayList maintains the insertion order internally. It inherits the AbstractList class and implements List interface.

#### 3.2 Interface Design

User Interface in our project :

**Home** : Home is the main page which contains different pages of the project.



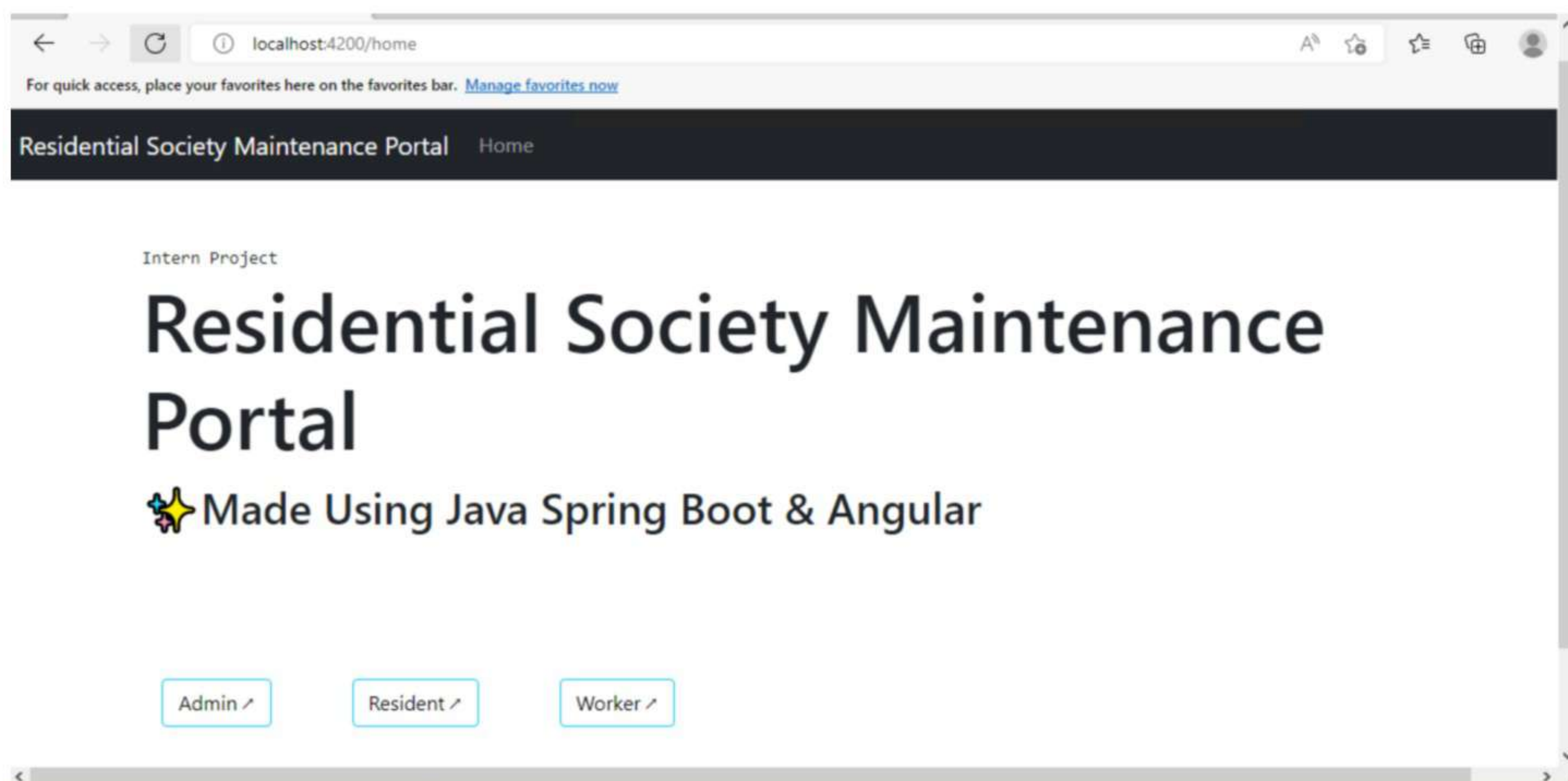


Figure 3.1: Project Home Page

## Admin Features

- can login
- can see the residents list
- can see the workers list
- can see the requests list
- can allocate the worker to the new requests

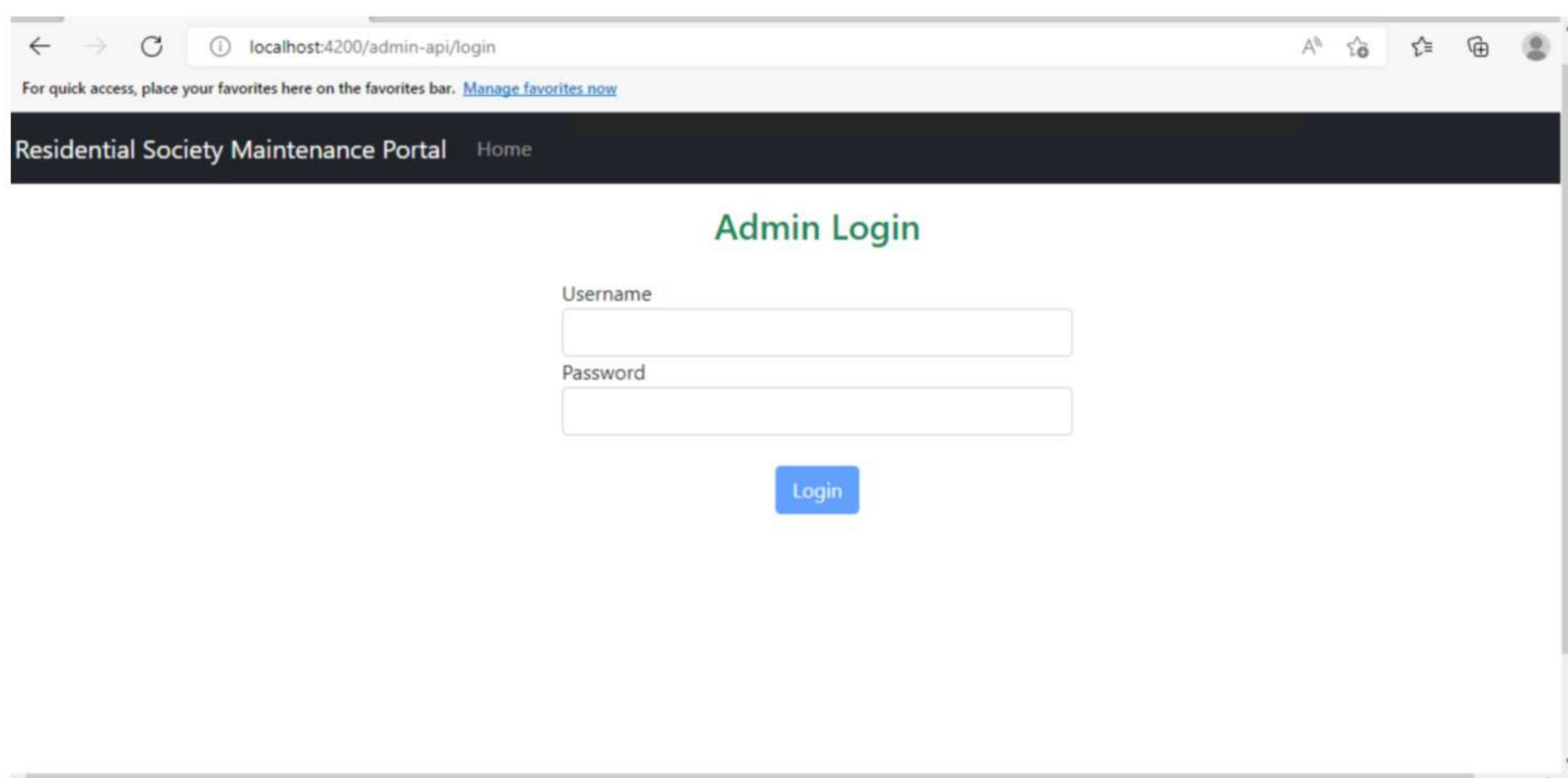


Figure 3.2: admin login page



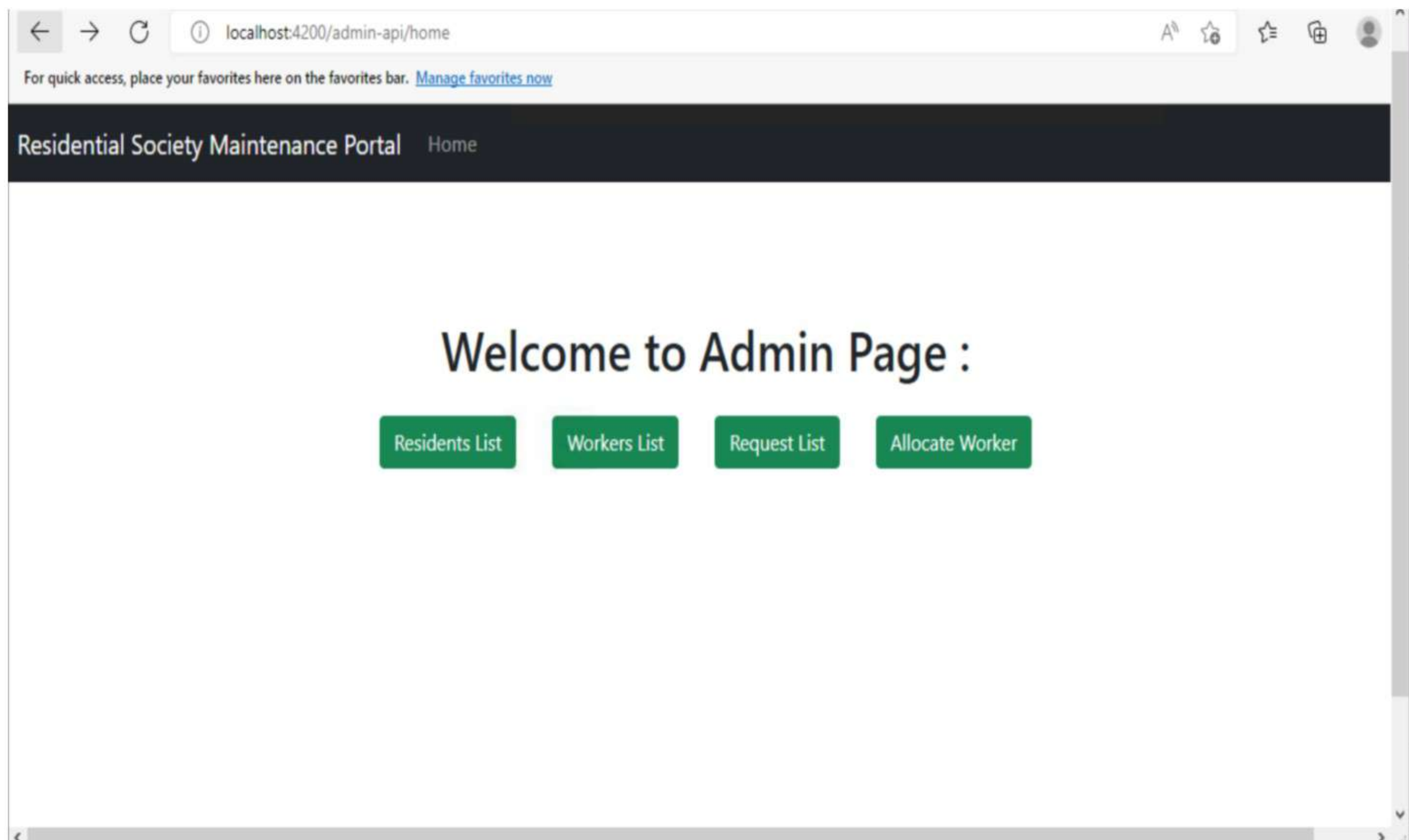


Figure 3.3: admin home page

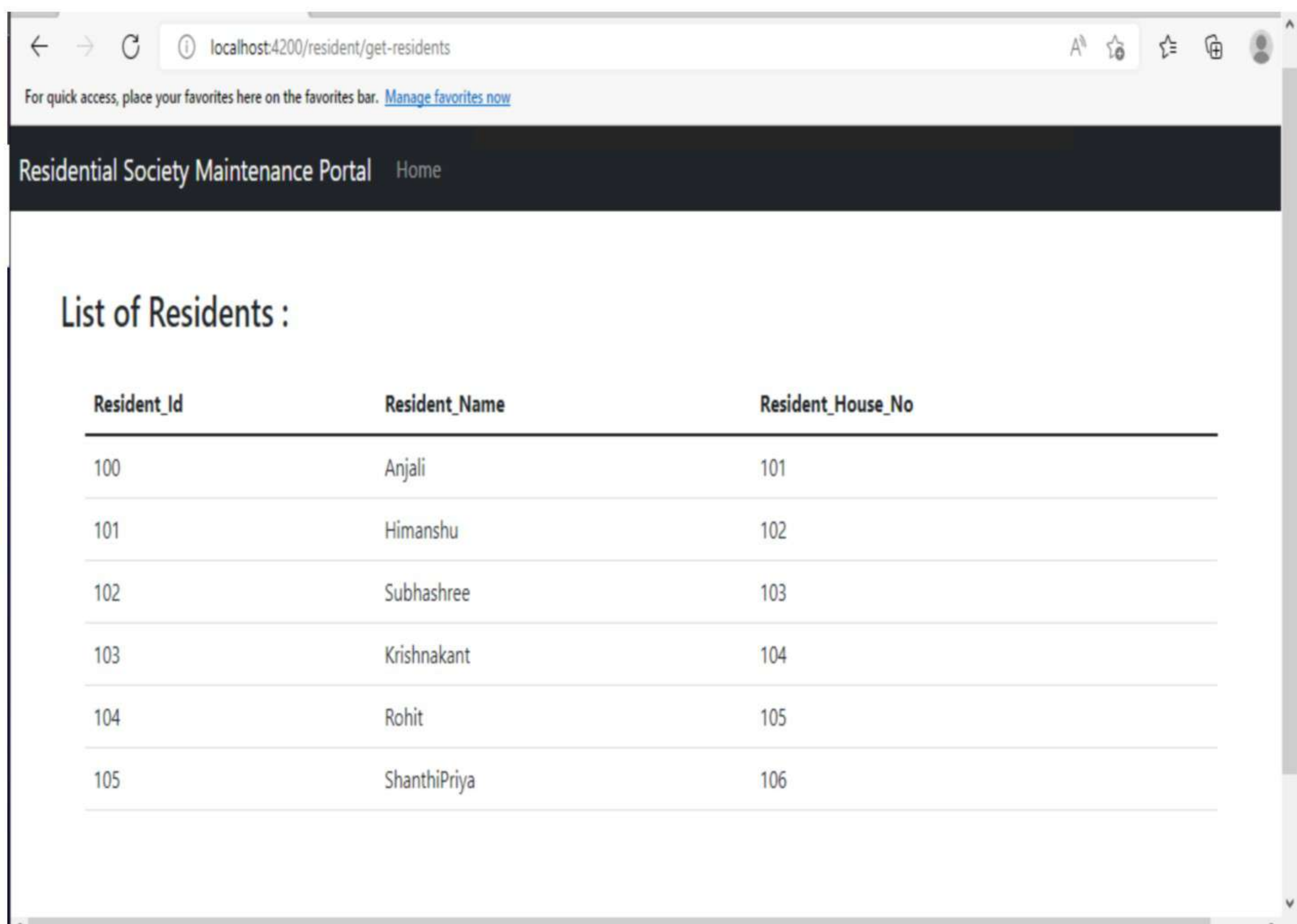
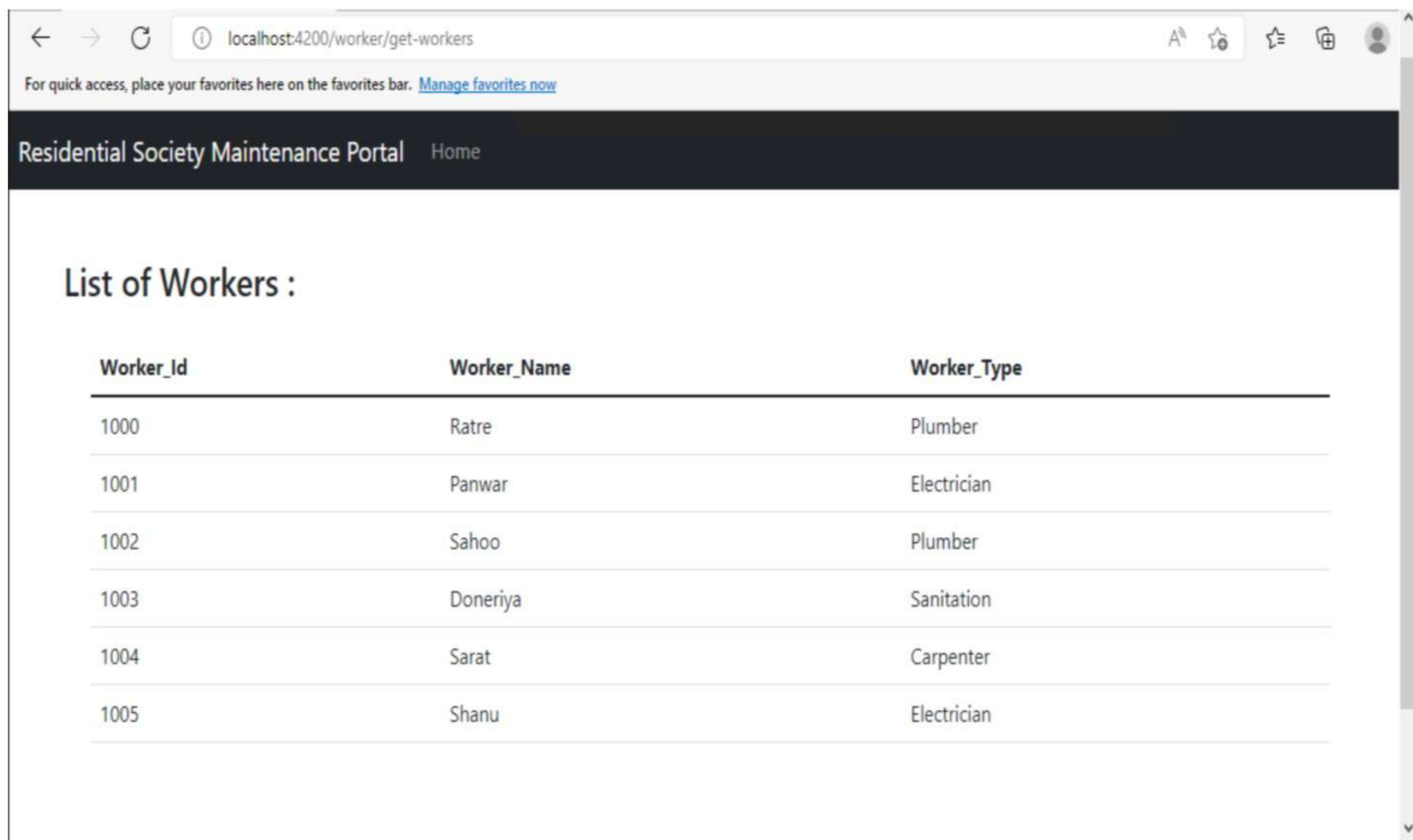


Figure3.4: list of residents shown to admin from database



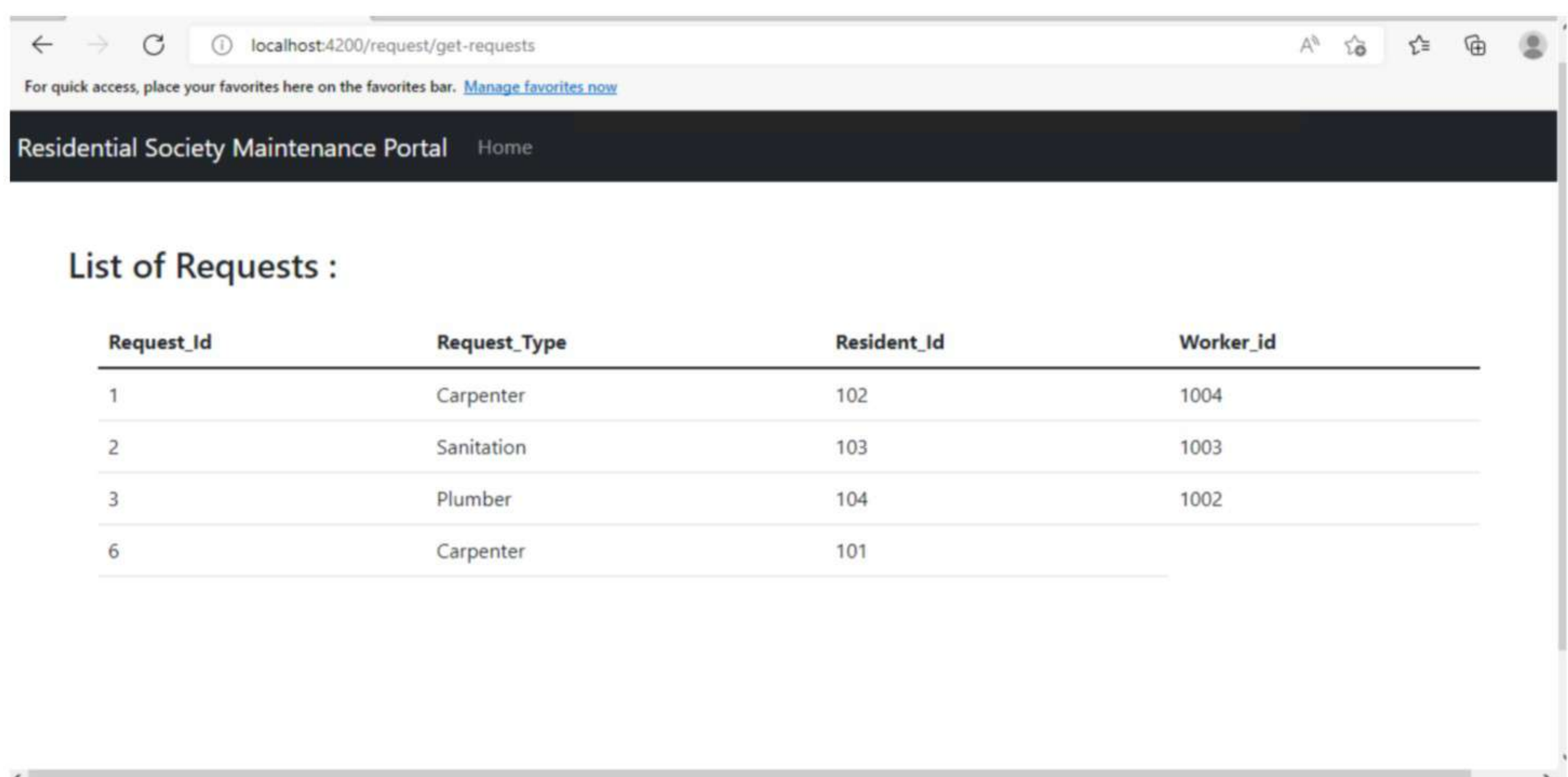


Residential Society Maintenance Portal Home

List of Workers :

Worker_Id	Worker_Name	Worker_Type
1000	Ratre	Plumber
1001	Panwar	Electrician
1002	Sahoo	Plumber
1003	Doneriya	Sanitation
1004	Sarat	Carpenter
1005	Shanu	Electrician

**Figure 3.5: list of workers shown to admin from the database**



Residential Society Maintenance Portal Home

List of Requests :

Request_Id	Request_Type	Resident_Id	Worker_id
1	Carpenter	102	1004
2	Sanitation	103	1003
3	Plumber	104	1002
6	Carpenter	101	

**Figure 3.6: list of requests shown to admin from database**



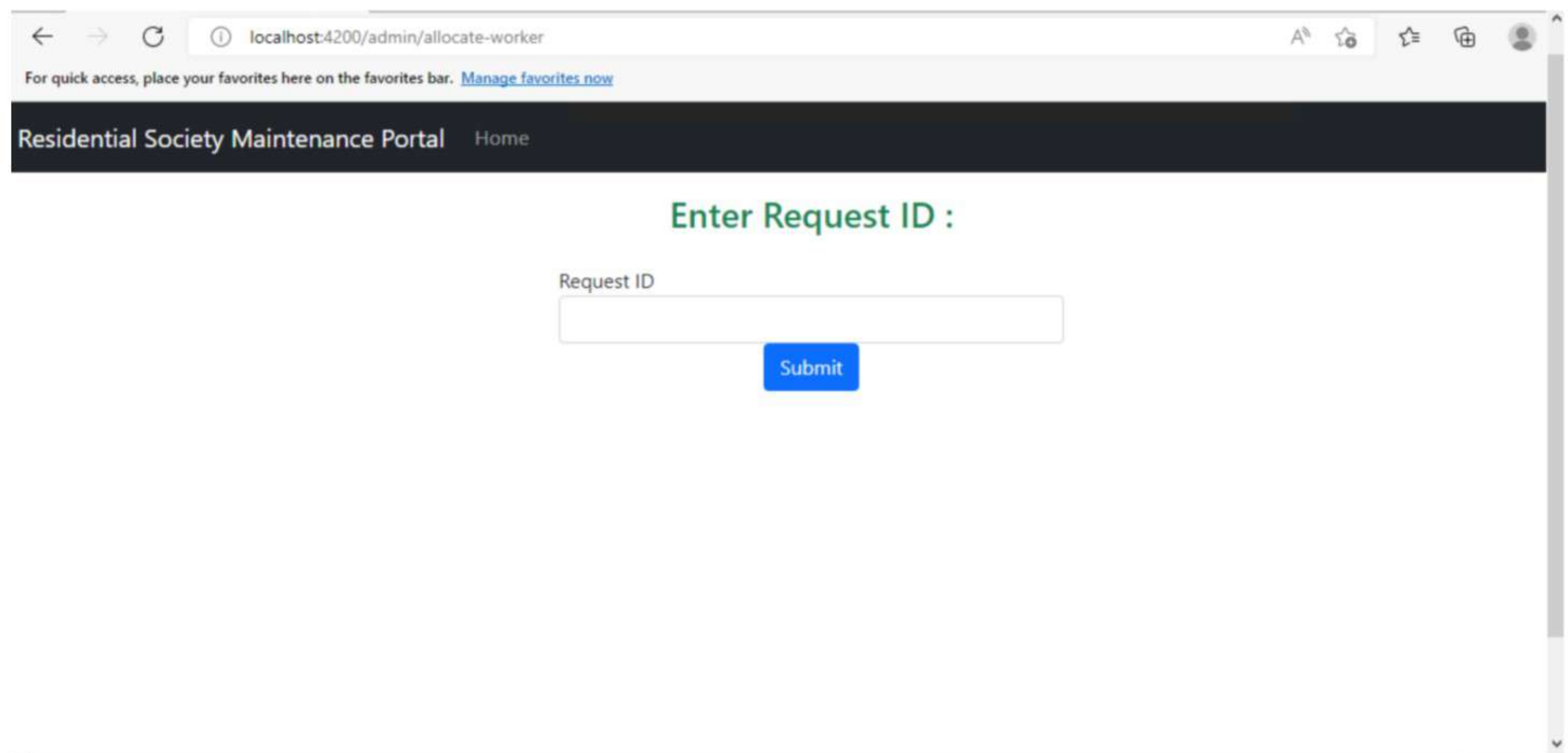
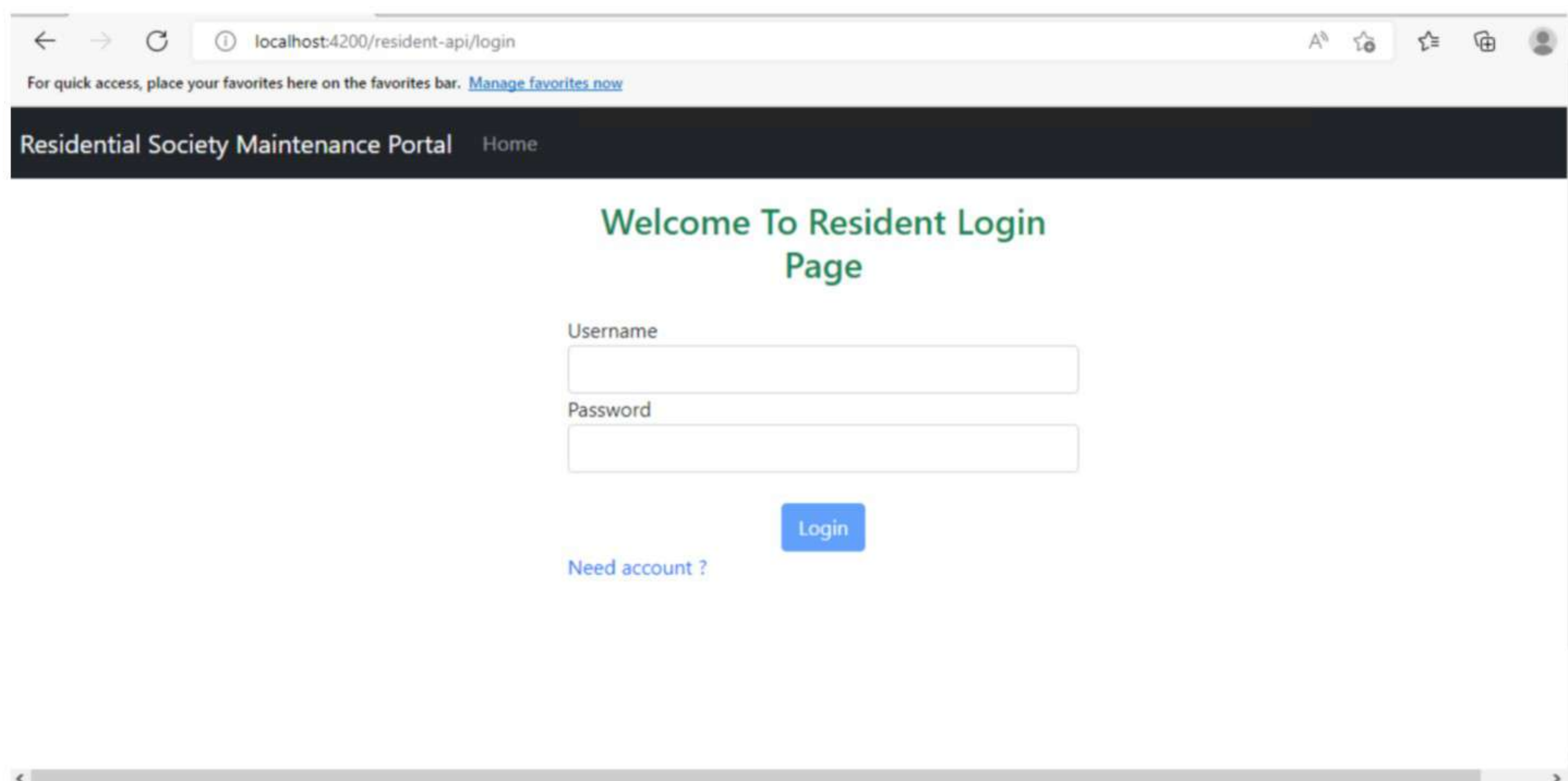


Figure 3.7: page where admin can allocate requests to the worker

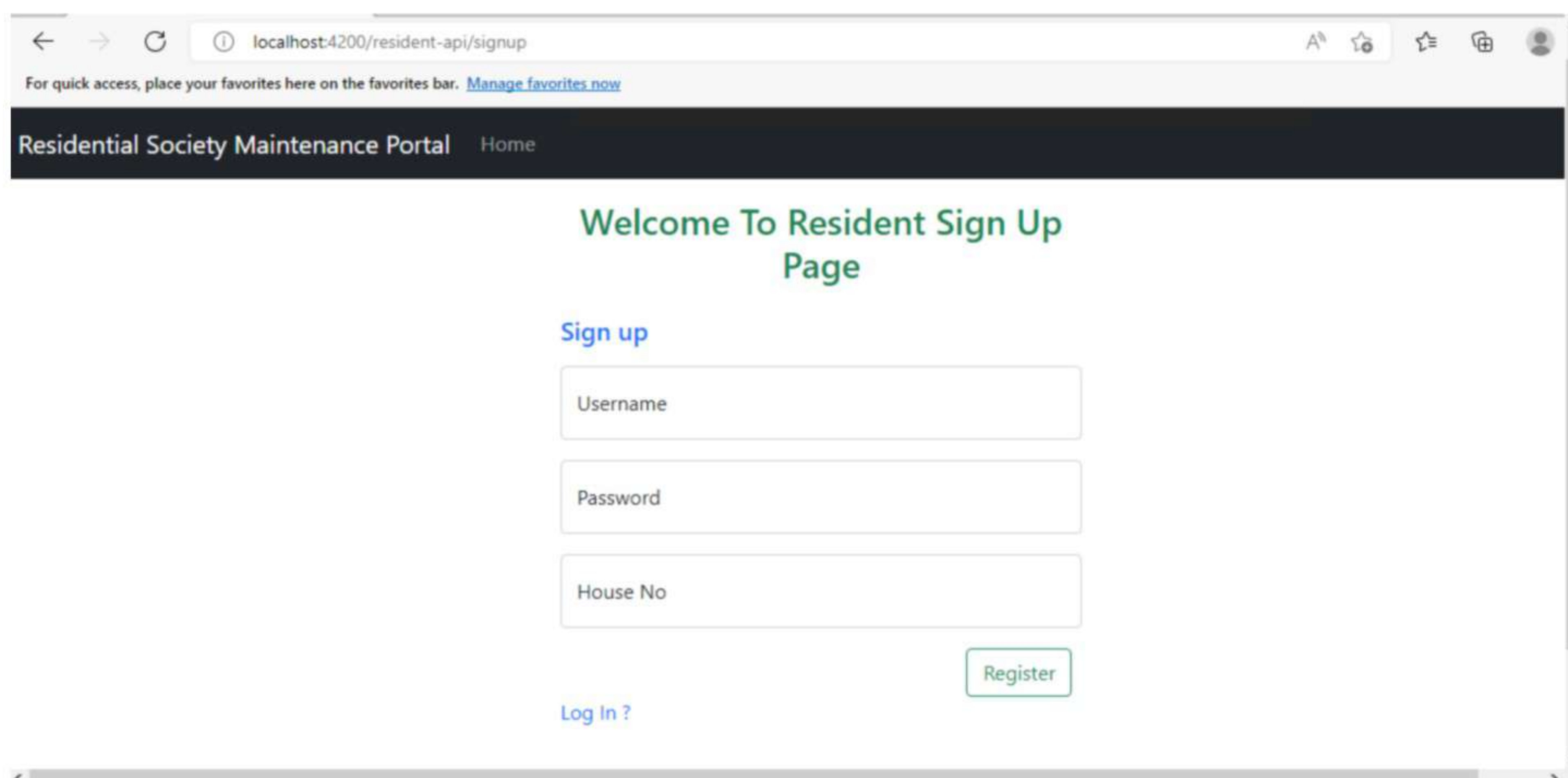
## Resident Features

- can login
- can sign up
- can see the list of requests logged by him/her
- can log new requests for the maintenance work
- can logout





**Figure 3.8: resident login page**



**Figure 3.9: resident sign up page**



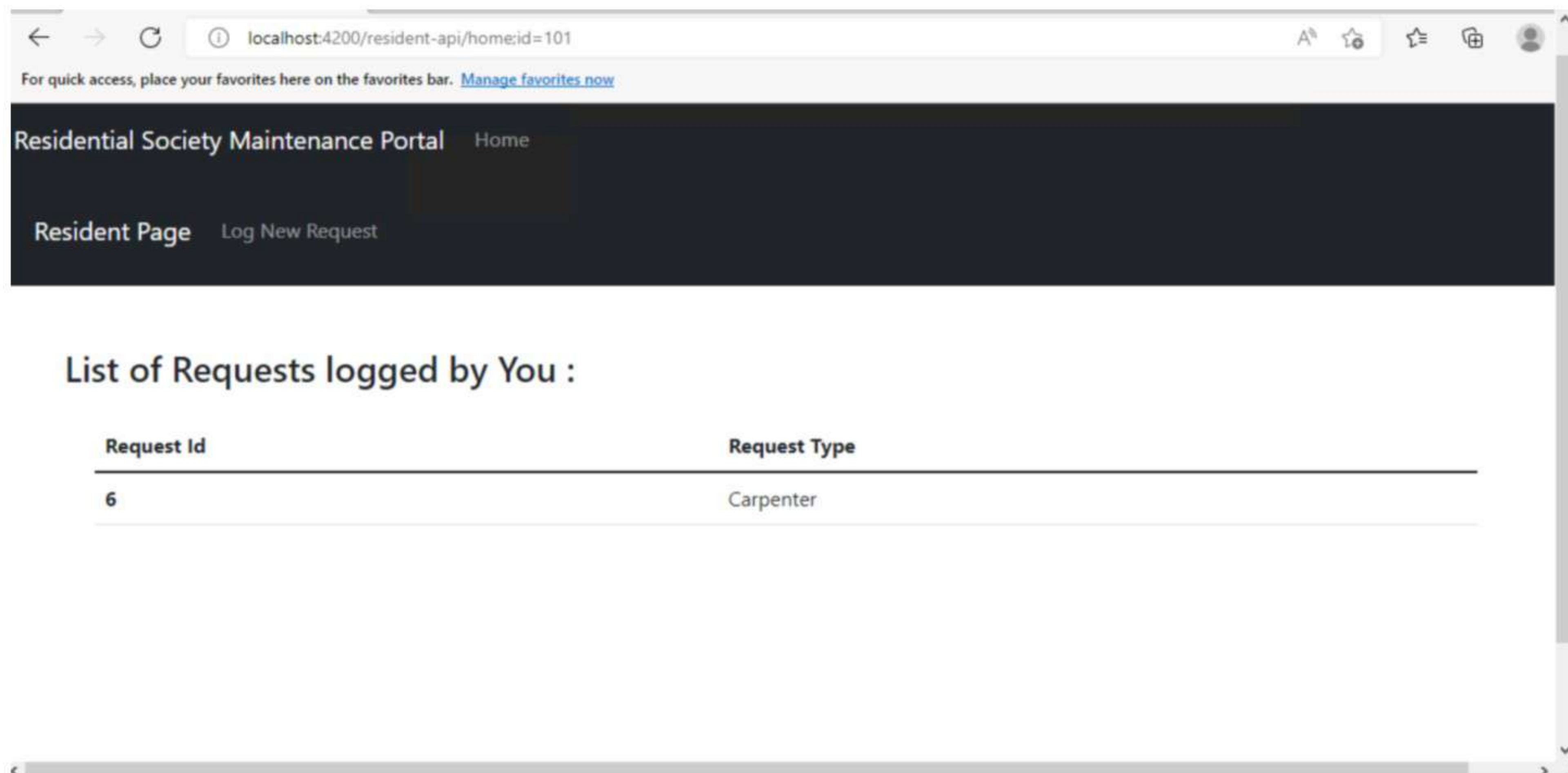


Figure 3.10: resident home page

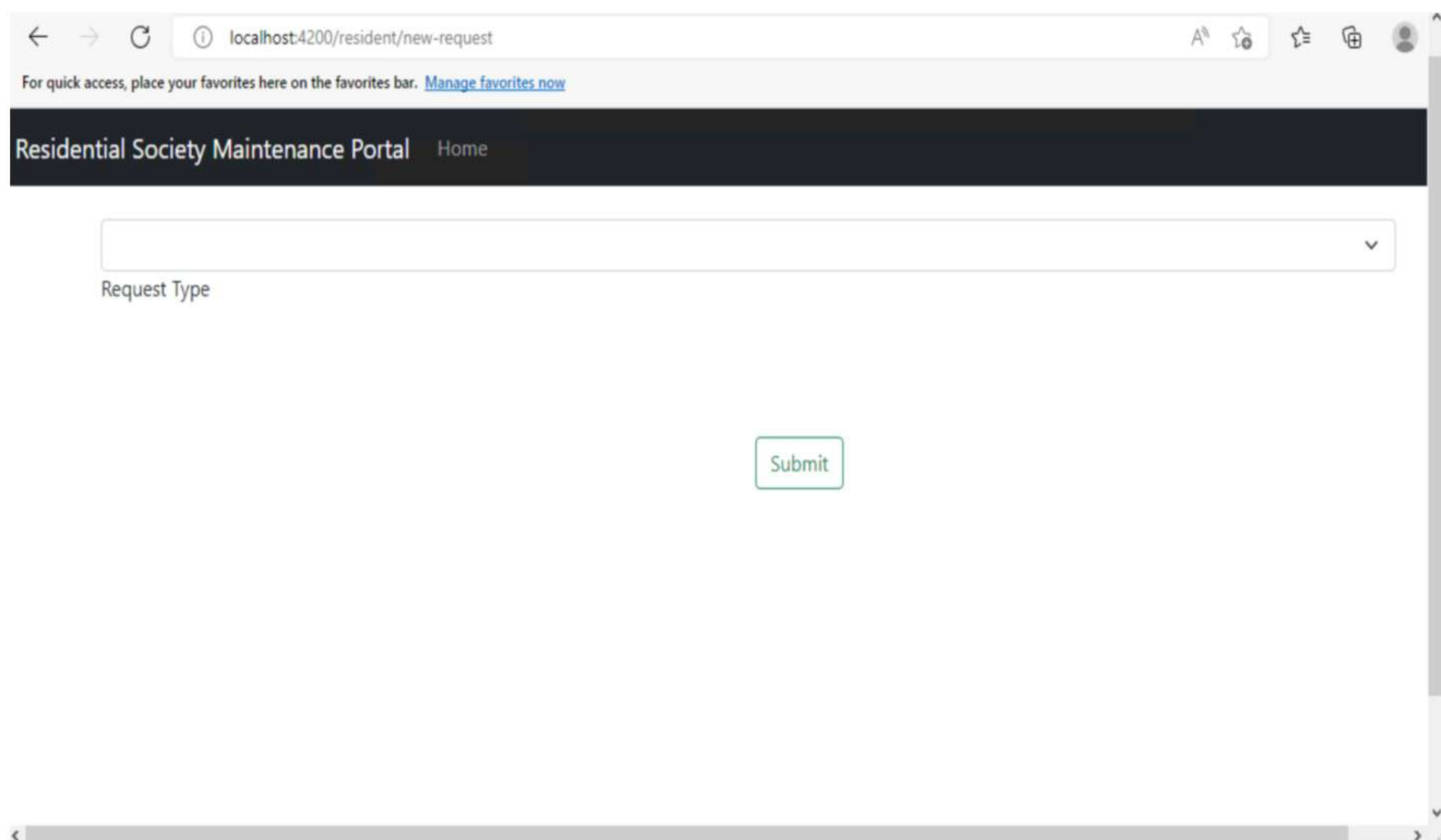
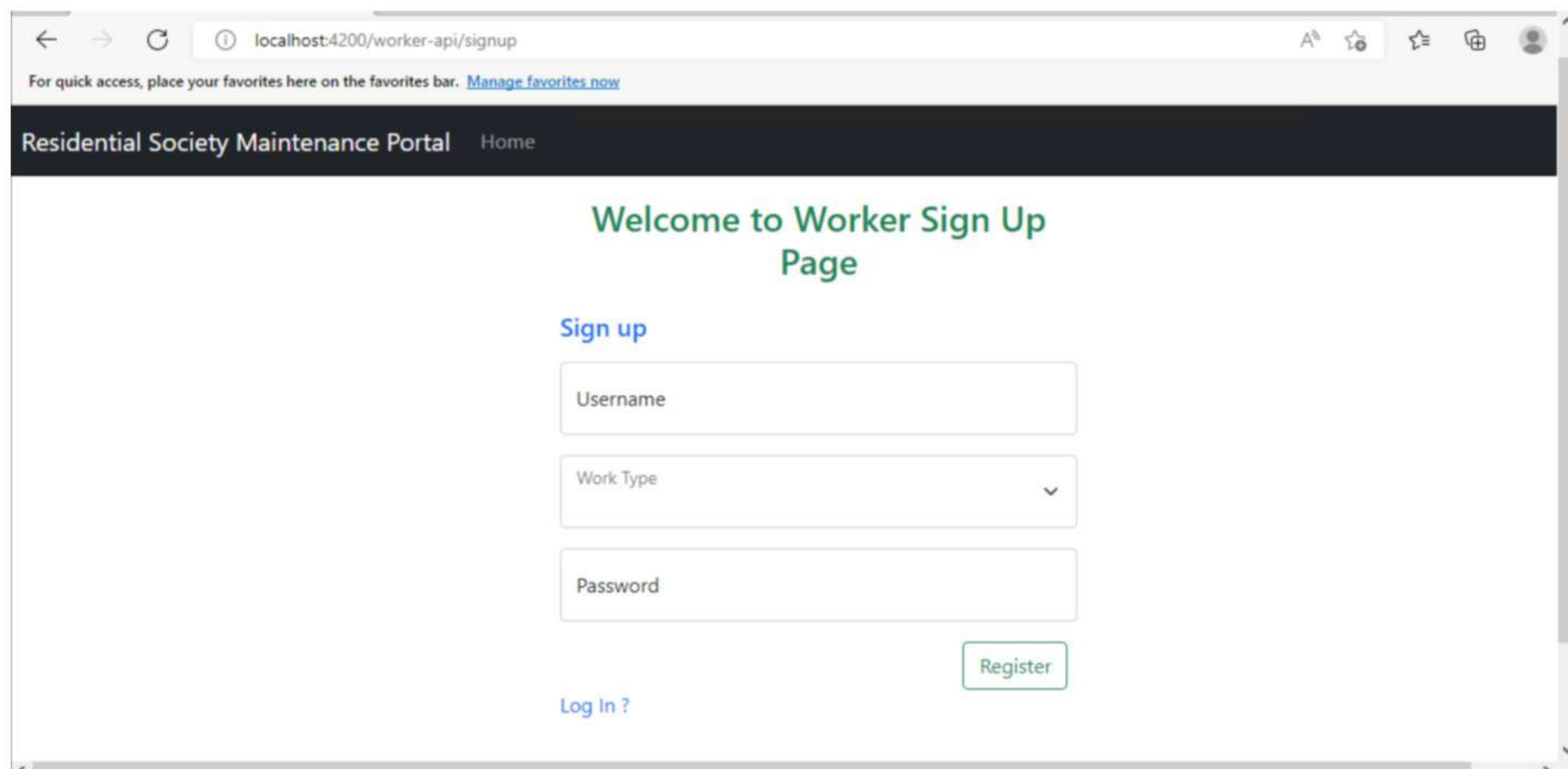


Figure 3.11: resident page to log new requests



## Worker Features

- can login
- can sign up
- can see the list of requests assigned to him/her
- can update the status of work to notify everyone.



The screenshot shows a web browser window with the address bar displaying `localhost:4200/worker-api/signup`. The page title is "Residential Society Maintenance Portal" with a "Home" link. The main heading is "Welcome to Worker Sign Up Page" in green. Below this is a "Sign up" link. The form contains three input fields: "Username", "Work Type" (a dropdown menu), and "Password". A "Register" button is positioned to the right of the "Password" field. At the bottom left, there is a "Log In ?" link.

Figure 3.12: worker sign up page



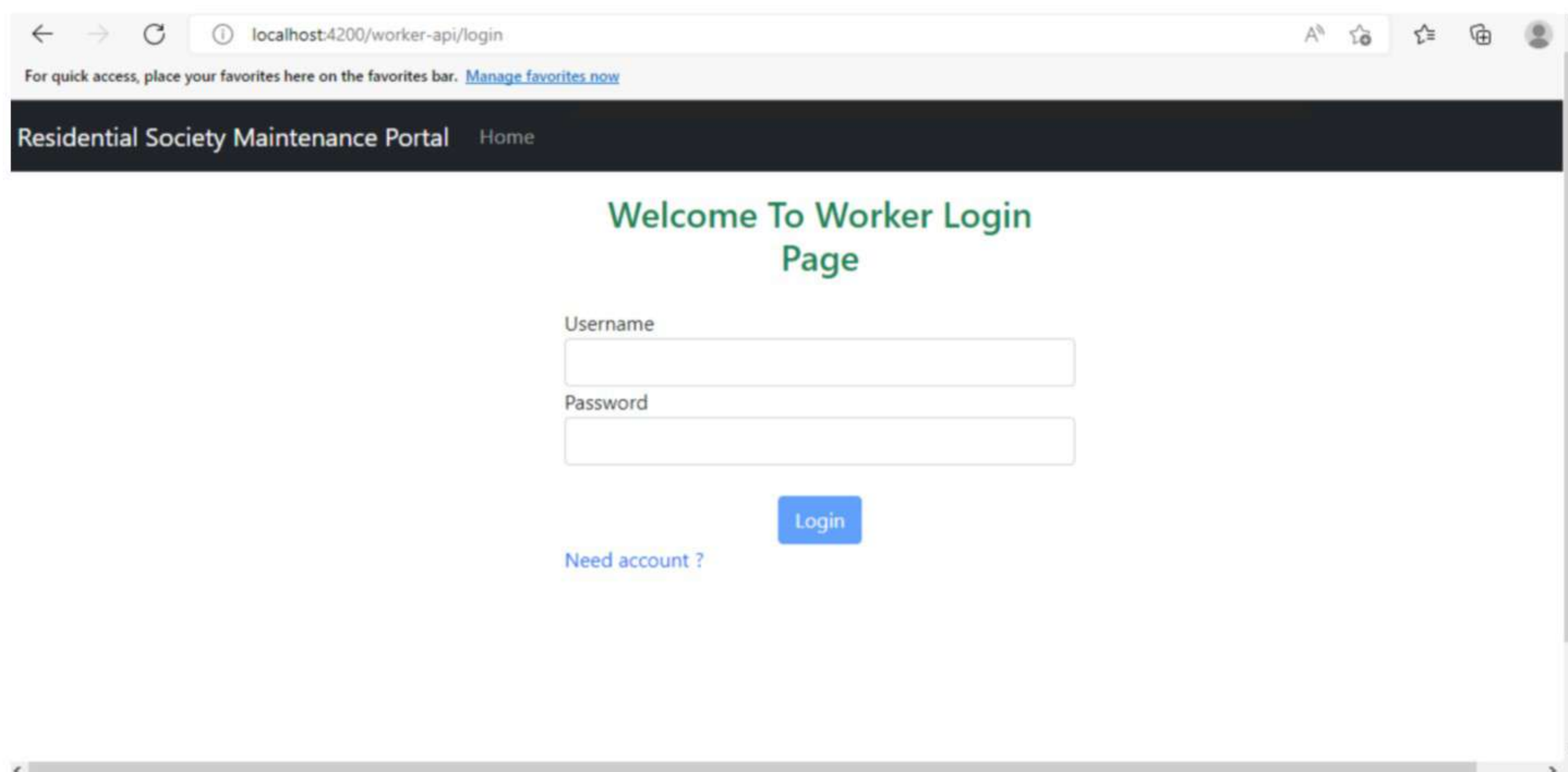


Figure 3.13: worker login page

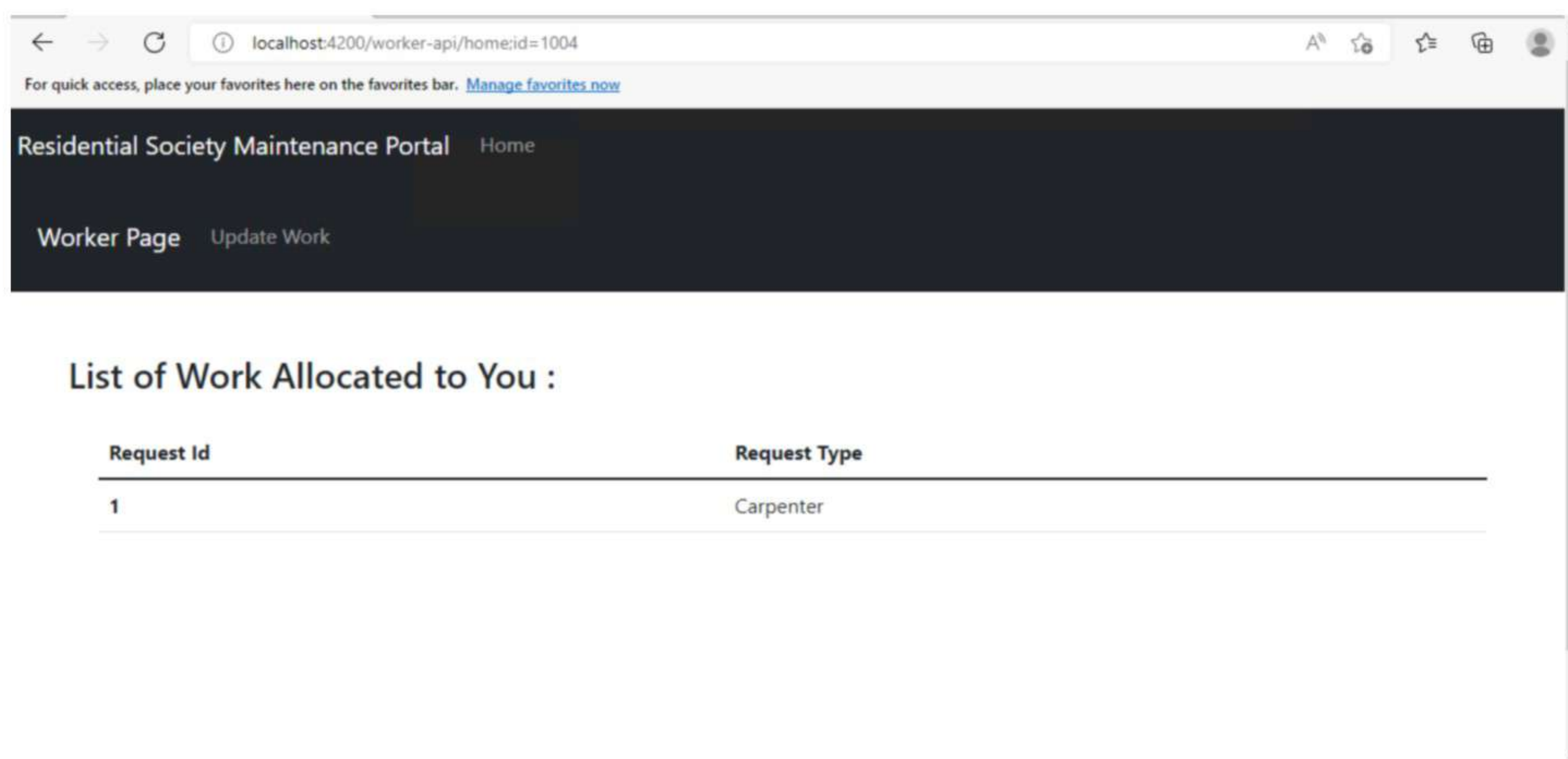


Figure 3.14: worker home page



localhost:4200/worker/update-work

For quick access, place your favorites here on the favorites bar. [Manage favorites now](#)

Residential Society Maintenance Portal Home

### Enter Request ID :

Request ID

Submit

**Figure 3.15: worker page to update the status of work**



## Chapter 4

# IMPLEMENTATION AND TESTING

Write one paragraph describing implementation and testing. Also write what you have done in this chapter (i.e. In the previous chapter, system design was explained in detail. In line with the design, the system implementation is presented in this paper)

### 4.1 Implementation

#### 1) Requirements

Recommended System Requirements:

- A good CPU and a GPU with at least 8GB memory.
- At least 8GB of RAM.
- Active internet connection so that the system can access the online resources through links.

Recommended Software Requirements:

- Eclipse IDE
- Visual Studio Code
- Postman-win64 - 7.26.1

Required Libraries for this project along with their version numbers used while making and testing of this project:

- Angular CLI - 13.2.5
- Node - 16.14.0
- Package Manager - npm 8.3.1
- OS – win32 x64
- Typescript – 4.5.5
- rxjs – 6.6.6
- apache-maven – 3.3.9



2) Procedure to configure Eclipse for Backend Set the following environment Path

variables

1. Set the following environment Path variables:
  - ✓ C:\Program Files\Zulu\zulu-11\bin\
  - ✓ %AppData%\npm
  - ✓ C:\Java\_Artifacts\Maven\apache-maven-3.3.9\bin\
2. Open Eclipse > click on windows > preferences > java > installed JRE's > add > Standard VM > set directory – C:\Program Files\Zulu\zulu-11\bin\ > finish > apply and close.
3. Open C drive > java artifacts > maven > copy settings and paste it in C:\Users\(your username)\.m2\ (paste settings inside .m2 folder).
4. Clone repository to preserve directory structure:
  - Go to File dropdown in the navigation bar.
  - Click on import...
  - Under Git folder, select – Projects from Git (with smart import) and click on next.
  - Select Clone URI and click on next.
  - Copy the web URL of your backend repository from Code dropdown of GitHub () and paste it in the URI field in Eclipse import projects from Git dialog box. And click on next.
  - Select branches to clone from remote repository and click on next.
  - Select the initial branch which will be considered as origin and click on next.
  - Click on Finish.
5. Once the repository is cloned successfully in the Eclipse, right click on project > build path > configure build path/select classpath > add library > JRE System Library > click on finish/apply & close.
6. Right click on project > maven project/update project... > select force update of snapshots > releases/ok.

3) Procedure to execute the Backend :

1. Open cmd > type mysqld – console.
2. Open another cmd > type mysql -u root > copy the TableScript > press enter.
3. Right click on novespoinapplication.java file > Run As > Java Application.
- 4) Procedure to configure Visual Studio Code for Frontend :    ▢

1. Set the following environment Path variables:
  - ✓ C:\Program Files\nodejs\
  - ✓ C:\Program Files\Git\bin\
2. Clone repository to preserve directory structure:
  - Go to View dropdown in the navigation bar.
  - Click on Command Palette...
  - Search and select Git: Clone option below the text box.
  - Copy the web URL of your frontend repository from Code dropdown of GitHub () and paste it in the text box in Vscode and press Enter.
  - Select .git folder in C:\Users\ (your username) \ directory and click on open folder.
3. Once the repository is cloned successfully in the Vscode, go to View dropdown in the navigation bar and click on Terminal.
4. Execute the following commands in the opened Terminal:
  1. npm install –g @angular/cli@13.2.5
  2. npm install
  3. ng v



5) Procedure to execute the Frontend :

1. Go to View dropdown in the navigation bar and click on Terminal.
2. Execute the following command in the opened Terminal:ng  
serve –open

NOTE: Execute the backend before executing the frontend. Both frontend and backend should be working simultaneously for the proper smooth successful work- ing of the Website.

### **4.1.1 Tools used**

Tools for project -

1. Eclipse: Eclipse is an integrated development environment used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment. It is the second-most-popular IDE for Java develop- ment, and, until 2016, was the most popular.

2. VS Code: Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features in- clude support for debugging, syntax highlighting, intelligent code completion, snip- pets, code refactoring, and embedded Git.

3. GitHub: GitHub, Inc. is a provider of Internet hosting for software devel- opment and version control using Git. It offers the distributed version control and source code management (SCM) functionality of Git, plus its own features. It pro- vides access control and several collaboration features such as bug tracking, feature requests, task management, continuous integration and wikis for every project.

4. MySQL Server: The MySQL server provides a database management system with querying and connectivity capabilities, as well as the ability to have excellent data structure and integration with many different platforms. It can handle large databases reliably and quickly in high-demanding production environments. The MySQL server also provides rich function such as its connectivity, speed, and se- curity that make it suitable for accessing databases. The MySQL server works in a client and server system. This system includes a multiple-threaded SQL server that supports varied back ends, different client programs and libraries, administrative tools, and many application programming interfaces (API)s.

### **4.1.2 Description of main modules**

Main modules of project is:

- Home Page:- This is the home page of the project.



- Login:- From this user can login in website by username and password at anytime to avail more features of website



- Sign up :- In this user can sign up and can able to feed details in the databaselike username , email , password.

## 4.2 Testing

- 1. Testing objective :-  
Software testing is a process of executing a program or application with the intent of finding the software bugs. It can also be stated as the process of validating and verifying that a software program or application or product: Meets the business and technical requirements that guided it's design and development Works as expected Can be implemented with the same characteristic.
- 2. Testing scop:-
  - 1) Process: Testing is a process rather than a single activity.
  - 2) All Life Cycle Activities: Testing is a process that's take place throughout the Software Development Life Cycle (SDLC). The process of designing tests early in the life cycle can help to prevent defects from being introduced in the code. Sometimes it's referred as "verifying the test basis via the test design". The test basis includes documents such as the requirements and design specifications.
  - 3) Static Testing: It can test and find defects without executing code. Static Testing is done during verification process. This testing includes reviewing of the documents (including source code) and static analysis. This is useful and cost-effective way of testing. For example: reviewing, walk through, inspection, etc.
  - 4) Dynamic Testing: In dynamic testing the software code is executed to demonstrate the result of running tests. It's done during validation process. For example: unit testing, integration testing, system testing, etc.



5) Planning: We need to plan as what we want to do. We control the test activities, we report on testing progress and the status of the software under test.

6) Preparation: We need to choose what testing we will do, by selecting test conditions and designing test cases.

7) Evaluation: During evaluation we must check the results and evaluate the software under test and the completion criteria, which helps us to decide whether we have finished testing and whether the software product has passed the tests.

- 3. Testing principles:-

Before applying methods to design effective test cases, a software engineer must understand the basic principles that guide software testing. All tests should be traceable to customer requirements. As we have seen, the objective of software testing is to uncover errors. It follows that the most severe defects (from the customer's point of view) are those that cause the program to fail to meet its requirements. Tests should be planned long before testing begins. Test planning can begin as soon as the requirements model is complete. Detailed definition of test cases can begin as soon as the design model has been solidified. Therefore, all tests can be planned and designed before any code has been generated. The Pareto principle applies to software testing. Stated simply, the Pareto principle implies that 80 percent of all errors uncovered during testing will likely be traceable to 20 percent of all program components. The problem, of course, is to isolate these suspect components and to thoroughly test them. Testing should begin "in the small" and progress toward testing "in the large." The first tests planned and executed generally focus on individual components. As testing progresses, focus shifts in an attempt to find errors in integrated clusters of components and ultimately in the entire system. Exhaustive testing is not possible. The number of path permutations for even a moderately sized program is exceptionally large. For this reason, it is impossible to execute every combination of paths during testing. It is possible, however, to adequately cover program logic and to ensure that all conditions in the component-level design have been exercised. To be most effective, testing should be conducted by an independent third party. By most effective, we mean testing that has the highest probability of finding errors (the primary objective of testing).

- 4. Testing method use:-

System Testing (ST) is a black box testing technique performed to evaluate the complete system the system's compliance against specified requirements. In System



testing, the functionalities of the system are tested from an end-to-end perspective. System Testing is usually carried out by a team that is independent of the development team in order to measure the quality of the system unbiased. It includes both functional and Non-Functional testing.

1) Black Box Testing: Black box testing is a software testing technique in which functionality of the software under test (SUT) is tested without looking at the internal code structure, implementation details and knowledge of internal



paths of the software. This type of testing is based entirely on the software requirements and specifications. In Black Box Testing we just focus on inputs and output of the software system without bothering about internal knowledge of the software program. The above Black-Box can be any software system you want to test. For example: an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation. Types of Black Box Testing: There are many types of Black Box Testing but following are the prominent ones -

a) Functional testing - This black box testing type is related to functional requirements of a system; it is done by software testers.

b) Non-functional testing - This type of black box testing is not related to testing of a specific functionality, but non-functional requirements such as performance, scalability, usability.

c) Regression testing - Regression Testing is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

2) White Box Testing: White Box testing (also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing) is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. Programming know-how and the implementation knowledge is essential. White box testing is testing beyond the user interface and into the nitty-gritty of a system. This method is named so because the software program, in the eyes of the tester, is like a white/transparent box; inside which one clearly sees. Types of White Box Testing:

- Unit Testing: It is often the first type of testing done on an application. Unit Testing is performed on each unit or block of code as it is developed. Unit Testing is essentially done by the programmer. As a software developer, you develop a few lines of code, a single function or an object and test it to make sure it works before continuing. Unit Testing helps identify majority of bugs, early in the software development lifecycle. Bugs identified in this stage are cheaper and easy to fix.

- White Box Penetration Testing: In this testing, the tester/developer has full



information of the application's source code, detailed network information, IP addresses involved and all server information the application runs on. The aim is to attack the code from several angles to expose security threats. · White Box Mutation Testing: Mutation testing is often used to discover the best coding techniques to use for expanding a software solution.



## **Chapter 5**

# **Conclusion and Future Scope**

### **5.1 Conclusion**

We had this conclusion that we can log household related maintenance request and the get the problem solved. Everything nowadays is going online and more and more people are interested in these online services.

### **5.2 Future Scope**

Scope of the project is limited.

1. We can build an android app, so the user can easily access the portal in phone through app.
2. We can add renting facility feature in future.



## References

- Infosys global application **LEX** for Education and Training purposes  
URL: <https://lex.infosysapps.com>
- Infosys global application **SPRING BOARD** for Education and Training purposes  
URL: <https://infyspringboard.onwingspan.com/>
- Application **SPRING** for spring boot application for backend configurations purposes  
URL: <https://spring.io/projects/spring-boot/>
- Application for to guide for developing service layer, persistence layer and rest API's for spring boot application.  
URL: <https://www.javacodegeeks.com/2012/09/spring-dao-and-service-layer.html>
- Application for to guide for developing service layer, persistence layer and rest API's for spring boot application  
URL: <https://www.sourcecodeexamples.net/2021/08/spring-boot-project-with-controller.html>
- Application for to guide for developing service layer, persistence layer and rest API's for spring boot application  
URL: <https://www.javatpoint.com/spring-tutorial>



## **Appendices:**

**Note:** As per the Infosys Private Limited, Mysore's privacy policies, Industrial mentor cannot give signature on FPR reports and FPR has been provided on monthly bases.



