**“HouseKeepers”**

**Major project report**

***Submitted by:-***

0206CS191119

0206CS191100

0206CS191090

0206CS191123

0206CS191062

**Piyush Raj Kumar**

**Manul Shrivastava**

**Kshitij Chauhan**

**Prateek Pathak**

**Deepak Manglani**

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

***in***

**COMPUTER SCIENCE &ENGINEERING**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

GYAN GANGA INSTITUTE OF TECHNOLOGY & SCIENCES

JABALPUR (M.P.)

RAJIV GANDHI PRODYOGIKI VISHWAVIDYALAYA,

BHOPAL (M.P.)

**December- 2022**

#### CERTIFICATE

This is to certify that the Major Project Report entitled “**HouseKeepers**” submitted by **Piyush Raj Kumar , Manul Shrivastava , Prateek Pathak , Kshitij Chauhan , Deepak Manglani**  has been carried out under my guidance & supervision. The project report is approved for submission towards partial fulfillment of the requirement for the award of degree of **BACHELOR OF ENGINEERING** in **COMPUTER SCIENCE & ENGINEERING** from **RAJIV GANDHI PROUDYOGIKI VISHWA-VIDYALAYA, BHOPAL (M.P).**

|  |  |
| --- | --- |
| **Dr. Ashish Mishra**  Guide  **Dept. of Computer Science and Engineering** | **Dr. Ashok Verma**  HoD  **Dept. of Computer Science and Engineering** |

#### CERTIFICATE

This is to certify that the Major Project Report entitled “**HouseKeepers**” is submitted by **Piyush Raj Kumar , Manul Shrivastava , Prateek Pathak , Kshitij Chauhan , Deepak Manglani**  for the partial fulfillment of the requirement for the award of degree of **BACHELOR OF ENGINEERING** in **COMPUTER SCIENCE & ENGINEERING** from **RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P).**

Internal Examiner External Examiner

Date: Date:

DECLARATION

We hereby declare that the project report entitled **“HouseKeepers”** which is being submitted in partial fulfillment of the requirement for award of the Degree of Bachelor of Engineering in Computer Science and Engineering to **“RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P.)”** is an authentic record of our own work done under the guidance of **Dr. Ashish Mishra , Department of Computer Science & Engineering,** **GYAN GANGA INSTITUTE OF TECHNOLOGY & SCIENCES, JABALPUR**.

The matter reported in this report has not been submitted earlier for the award of any other degree.

**Date:**

**Place: JABALPUR**

#### ACKNOWLEDGEMENT

We sincerely express indebtedness to esteemed and revered guide **Dr. Ashish Mishra of Department of Computer Science and Engineering**for his invaluable guidance, supervision and encouragement throughout the work. Without his kind patronage and guidance the synopsis would not have taken shape.

We take this opportunity to express deep sense of gratitude to **Dr. Ashok Verma, Head of Department of Computer Science and Engineering**for his encouragement and kind approval. Also we thank him in providing the computer lab facility. We would like to express our sincere regards to him for advice and counseling from time to time.

We owe sincere thanks to all the faculties in Department of Computer Science and Engineeringfor their advice and counseling time to time.

**Piyush Raj Kumar**

**Manul Shrivastava**

**Prateek Pathak**

**Kshitij Chauhan**

**Deepak Manglani**

**Date :**

##### **Place : JABALPUR**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **TABLE OF CONTENTS**   |  |  |  | | --- | --- | --- | | **Serial No.** | **Title Page No.** | **Page No.** | |  | ABSTRACT 8 | 1 | | **1.** | **INTRODUCTION 8** | 2 | |  | * 1. Purpose of Project 8   1.2 Scope of the Project 8 | 3 | |  | * 1. Project and Product Overview 8 | 4 | |  | * 1. Designs Goals 9   1.5 Intended Audience 9 | 4 | |  | 1.6 Team Architecture 9  1.7 Survey of Technology 11 | 4 | | **2.** | **PROBLEM STATEMENT 15** | 9 | |  | 2.1 Selection of Product 15 | 9 | |  | 2.2 System Requirement 15 | 9 | |  | 2.3 Proposed System 16 | 10 | |  | 2.5Limitations of Proposed System 16  2.8 Application 16 | 10 | | **3.** | **SPECIFIC REQUIREMENTS 16** |  | |  | 3.1 User Interface 16  3.2 Hardware Interface 17  3.3 Software Interface 17  3.4 Communication Interface 17  3.5 Non functional Requirements 17  3.6 Software System Attributes 18 | 12 | |  |  |  | | **4.** | **SOFTWARE PROCESS MODEL 19** | 14 | |  | |  |  | | --- | --- | | 4.1 Determining project feasibility 19 |  | |  |  | |  | | **5.** | **OVERALL DESIGN AND IMPLEMENTATION 20** |  | | **6.** | **LIST OF FIGURES 26- 32**  6.1 List of figures 20-36  6.2 List of Tables 27  6.3 Screenshots 28-32 |  | | **7.**  **8.**  **9.**  **10.** | **TEST CASES 33-36**  **FUTURE ENHANCEMENT 36**  **CONCLUSION 37**  **REFERENCES 37** |  | |  |  |  | |  |  | |  |  | |  |  | |  |  | |  | **LIST OF FIGURES** |  | |  |  |  | |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **S.No** | **Figure Name** | **Page No.** |
| 6.1 | Use case diagram | 21 |
| 6.2 | Sequence diagram | 21 |
| 6.3 | Class diagram | 22 |
| 6.4 | E-R diagram | 23 |
| 6.5 | Data flow diagram level 0 | 25 |
| 6.6 | Data flow diagram level 1 | 25 |
| 6.7 | Activity diagram | 26 |

**ABSTRACT**

# Time is the essence. Most, if not all the inventions and discoveries are aimed at us humans saving time. We want everything done quicker, faster, and sooner. We live as though we just do not have enough time – nevertheless we are all equally endowed with 24 hours to spend. Apps have come into existence to save time in diverse domains. House cleaning services App saves us time and energy in the upkeep of our homes and offices.

# **1.Introduction**

# **1.1Purpose Of the Project**

Design an application which hostel students can use to get their rooms cleaned on time. The current management system found it difficult to schedule cleaning services as students have varying class schedules which resulted in improper utilization of hostel room cleaning services.

**1.2 Scope of the project**

## Create a application that hostel guests can use to schedule timely room cleanings. Because students have variable class schedules, the current management system found it challenging to plan cleaning services, which led to an incorrect use of the hostel room cleaning services.

## **1.3Project And ProductOverview**

A web application that hostels inmates can use to schedule timely cleanings of their rooms. Due to students' variable class schedules and the current management system's difficulty in scheduling cleaning services, hostel room cleaning services were not used properly.

**1.4 Design Goals**

1. **Admin Login:** Admin Logins to the system by entering his/her personal details like ID and password through which he can login to the systems database.
2. **Dashboard (User Level) :** This page will show the current status of the application.
3. **Request:** All the requests that are processed by the user will be visible here.
4. **Feedback:** Comments given to the workers that how is their work according to the user.
5. **Profile: The student’s profile is visible here, the student can change his/her profile as per them.**
6. **Admin Level Dashboard: It monitores the workflow of workers and the registered students**.
7. **Allotment**: The Workers are allotted to the student as per their individual work
8. **Suggestions: The Suggestions are the feedback which was given by the user which shows that what needs to be changed or improved**
9. **Register Student: The students need to register in this application to use this application by adding room no, enrollment no and floors.**
10. **Register Housekeeper: The housekeepers record and how they are managed**

**1.5 Intended Audience**

The expected audience of this document is the users of the Hostel and Students in it. Students and hostel guests are the intended audience for this publication.

**1.6 Team Architecture**

There were two teams and each team has two members. Complete project was divided and given to the each team

* **TEAM 1**

**The members in team 1 are:**

**Manul Shrivastava**

**Deepak Manglani**

**Kshitij Chauhan**

**REQUIREMENT GATHERING AND ANALYSIS**

It is the first and most important part of project. With the correct analysis and requirement gathering chances of project to be successful increases

The process to gather the software requirements from client, analyze and document them is known as requirement engineering.

The goal of requirement engineering is to develop and maintain sophisticated and descriptive ‘System Requirements Specification’ document.

**DESIGNING PART - 1**

This part will be responsible for the creation of database. There will be total of three database tables The E-R model will be generated first, followed by the relational model and then finally the database tables will be created

**CODING PART – 1**

This will involve the conversion of designing part-1 into the code. The front end would be generated from this. The proper GUI would be made at the end of this phase. The technologies used for the front end will be JavaScript, HTML and CSS. The modules of Admin, Students and Workers are made by this team.

* **TEAM 2**

**The members in team 2 are:**

**Piyush Raj Kumar**

**Prateek Pathak**

**DESIGNING PART – 2**

Designing phase is very important. With the help of designing the interface of the application is created. It is the interface only with which the user will interact with the application This part of designing consists of the outline of the basic structure of the project. The design of the modules of our project that is:

1.Login module:

**CODING PART – 2**

In this part the modules that were left over in the coding part-1 will be made. The modules such as Students and the workers will be made here.

2.Students module

Student Information

3.Worker’s module

List of workers

**TESTING**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects), and to verify that the software product is fit for use. Software testing involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

* meets the requirements that guided its design and development,
* responds correctly to all kinds of inputs,
* performs its functions within an acceptable time,
* is sufficiently usable,
* can be installed and run in its intended environments, and
* achieves the general result its stakeholder’s desire.

**1.7 Survey Of Technology**

Design of the site has been done using the following technologies: -

* HTML
* PHP
* MYSQL
* CSS
* JavaScript
* Bootstrap

**HTML: HYPER TEXT MARKUP LANGUAGE**

In computing, Hypertext Markup Language (HTML) is a markup language designed for creation of web pages with hypertext and other information to be displayed in a web browser. HTML is used to structure information denoting certain text as headings, paragraphs, lists and so on and can be used to describe, to some degree, the appearance and semantics of a document. HTML’s grammar structure is the HTML DTD that was created using SGML syntax.

The HTML document format is used on the Web. Web pages are built with HTML tags (codes) embedded in the text. HTML defines the page layout, fonts and graphic element as well as the hypertext links to other documents on the web. Each link contains the URL, or address, of a Web page residing on the same server or any server worldwide, hence “World Wide Web”.

HTML is a markup language (the ML in HTML) that uses a fixed set of markup tags. A markup language can also be thought of as a “Presentation Language”, but it is not a programming language. You cannot “if this-do that” like you can in Java, JavaScript or C++. However, in order to make pages interactive, programming code can be embedded in an HTML page.

**MySQL:**

Modern day web sites seem to be relying more and more on complex database systems. These systems store all of their critical data, and allow for easy maintenance in some cases. MySQL is a freely available open-source relational database management system (RDMS) that is structure query language (SQL). With its proven performance, reliability and ease of use, MySQL has become the leading database choice for web-based application.

The Structured Query Language (SQL) is a very popular database language, and its standardization makes it quite easy to store, update and access data. One of the most powerful SQL servers out there is called MySQL and surprisingly enough, its free.

Some of the features of MySQL Include: Handles large databases, in the area of 50,000,000+ records. No memory leaks. Tested with a commercial memory leakage detector (purify). A privilege and password system which is very flexible and secure, and which allows host-based verification. Passwords are secure since all password traffic when connecting to a server is encrypted.

**PHP: HYPERTEXT PREPROCESSOR**

PHP is a [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting) language designed for [web development](https://en.wikipedia.org/wiki/Web_development) but also used as a [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). Originally created by [Rasmus Lerdorf](https://en.wikipedia.org/wiki/Rasmus_Lerdorf" \o "Rasmus Lerdorf) in 1994, the PHP [reference implementation](https://en.wikipedia.org/wiki/Reference_implementation) is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the [recursive](https://en.wikipedia.org/wiki/Recursive_acronym) [backronym](https://en.wikipedia.org/wiki/Backronym) PHP: Hypertext Preprocessor.

**CSS: CASCADING STYLE SHEETS**

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

**JAVASCRIPT:**

JavaScript (js) is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses js to provide several forms of interactivity and simplicity.

**BOOTSTRAP:**

Bootstrap is a free and open-source web development framework. It’s designed to ease the web development process of responsive, mobile-first websites by providing a collection of syntax for template designs.

**TIER ARCHITECTURE.**

The various classes as obtained from the business class diagram is categorized as follows-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Form of the project** |  | **Class** |  | **Class** |
| **Application or Presentation Layer** |  | **Business Layer or Logical Layer** |  | **Data Layer or Data Access Layer** |

The 3-tier architecture consists of three layers:

**Presentation Layer** - The web site or windows forms application is called the presentation layer. The presentation layer is the most important layer simply because it’s the one that everyone sees and uses. Even with a well-structured business and data layer, if the presentation layer is designed poorly, this gives the users a poor view of the system. Presentation layer is the form where we design using the controls like textbox, labels, command buttons etc.

**Business Layer** - Though a web site could talk to the data access layer directly, it usually goes through another layer called the business layer.

This layer is a class which we use to write the function which works as a mediator to transfer the data from Application or presentation layer data layer. In the three-tier architecture we never let the data access layer to interact with the presentation layer.

This layer is also a class where we declare the variable corresponding to the fields of the database which can be required for the application and make the properties so that we can get or set the data using these properties into the variables. These properties are public so that we can access its values.

One of the best reasons for reusing logic is that applications that start off small usually grow in functionality. For instance, a company begins to develop a web site, and as they realize their business needs, they later decide to add a smart client application and windows service to supplement the web site. The business layer helps move logic to a central layer for “maximum reusability.”

Business layer have been presented having two roles

* client application
* server component

**EXAMPLE OF BUSINESS LAYER**

The Business layer has functions of which takes the parameters from the example given in the presentation layer .As the user inputs the data values, corresponding functions are called in the business layer and corresponding procedures are called and the data is been updated.

Business layer is the class where we write the functions which get the data from the application layer and passes through the data access layer.

**Data layer** - The key component to most applications is the data. The data has to be served to the presentation layer somehow. The data layer is a separate component whose sole purpose is to serve up the data from the database and return it to the caller. This layer is also a class which we use to get or set the data to the database back and forth. This layer only interacts with the database. We write the database queries or use stored procedures to access the data from the database or to perform any operation to the database.

**ADVANTAGE OF 3 TIER ARCHITECTURE**

* Client-Server architecture is 2-Tier architecture because the client does not distinguish between Presentation layer and business layer.
* The increasing demands on GUI controls caused difficulty to manage the mixture of source code from GUI and Business Logic.
* Further, Client Server Architecture does not support enough the Change Management. Let suppose that the government increases the Entertainment tax rate from 4% to 8 %, then in the Client-Server case, we have to send an update to each client and they must update synchronously on a specific time otherwise we may store invalid or wrong information.
* The Client-Server Architecture is also a burden to network traffic and resources. Let us assume that about five hundred clients are working on a data server then we will have five hundred ODBC connections and several ruffian record sets, which must be transported from the server to the clients.
* This categorization of the application makes the function more reusable easily and it becomes too easy to find the functions which have been written previously. If programmer wants to make further update in the application, then he easily can understand the previous written code and can update easily.

**DISADVANTAGES**

* Increase complexity /effort
* More difficult to build 3 tier architecture rather than a 2 tier.
* Points of communication are doubled
* Maintenance tools are currently inadequate for maintaining server libraries.
* The system will provide fast and efficient automated environment instead of slow and error prone manual system, thus reducing both time and man power spent in running the system.
* The system will have GUI interface and very less user-training is required to learn it.
* The system will provide service to view various information for proper decision making

**2.Problem Statement:**

## Entry Point

## **The system is required to have two entry points:**

1)  **Admin Direct**: An “Admin Direct” entry point is where the administrator can manage the store products and service the customer requests.

2)  **Customer Linking**: A “Customer Linking” entry point is where a prospective customer can view the details of products and promotions and offers.

**2.1 Selection of Product**

Based upon the customer’s request the feasibility of providing guidance is evaluated. The customer request is fulfilled according to the respective symtoms.

**2.2 System Requirement**

 Performance and Scalability: The system is required to scale to provide guidance to the customer instantly. WebPages should be light and render fast.

* + 1. **Usage**

The web forms should be self-explanatory and usable. We do not want prospective clients dropping of the website because they cannot understand the forms and find them cumbersome.

**2.2Proposed system**

In the modern world people are aware of so many Sciences Like computer sciences, Electronics, commerce, political sciences etc. But they did not know about medical sciences and its related features much, because usage of scientific words which are difficult to remember and also difficult to pronounce. At this juncture there is a need for creating awareness in the public about the general topics of health sciences. Smart Health Prediction is the one, which provides direct communication with the client where he gets a remedy for his problem automatically by responding to small combination

# 2.3 Limitations:

* The system is not fully automated.
* It only works in online mode and will not be working offline.

# 2.4 Application

This system can be used by all patients or their family members who need help in emergency.

**3 .SPECIFIC REQUIREMENTS**

## **3.1 User Interface**

The user interface required to be developed for the system should be user friendly and attractive. All operations will be of point and click nature with all navigations performed through windows of the system specifically buttons and menus:The app will display the user interface to users which will be a GUI.

***Buttons:*** The button is activated when the user will click with the left click of the mouse within the bounds of the button. And thus the action associated with it will be carried out.

***Menu:***All the operations will be arranged

**3.2 Hardware Interface**

Here's What You Need to Use the health prediction system:

•    20 GB HDD  
 •    256 MB RAM  
 •    Pentium IV Processor  
 •    Input Devices: Keyboard, Mouse  
 •    Output Devices: Monitor, Printer

We use the above mentioned hardware tools for increased speed, reduced complexity and for improved capacity.

**3.3****Software Interface**

* Operating System: windows
* Language – php&mysql.
* Platform: Dreamweaver
* Supporting Tools : Microsoft @ Word 2007,Wamp server,photoscape

**3.4 Communication Interface**

* The health prediction app will display the user interface to users which will be a GUI.
* The customers while using the app will be communicating in online mode.

**3.5 Non-functional Requirements**

**User friendly:**

Our app should be more users friendly. The user interface should be kept simple and uncluttered. Since different type of people will interact in this process so our project should be very easy to them to understand.

**Flexibility:**

Our project should be so flexible that whenever we want to make changes in it very easily it can be done on.

**Security:**

All the records of users are more secure and arrange in a good manner

**Extensibility:**

Our project should be able to accommodate the variations like:

Different features should be handled easily.

Client interaction after sending his/her request.

**Reusable:**

All the client that are using our project should be easily get processed so that many clients can interact with us very easily and very fast without any information destroy.

**Cost:**

Our project should be very feasible and of lower cost so that maximum users will be able to get its benefit.

.

**3.6 Software System Attributes**

**Reliability:**

The prediction app should be easy and without any mistakes so that clients should be able to handle and make use of it very safely.

**Availability:**

The project should be available 24 hours a day, 7 days a week. The system will be available to the user whenever the user needs it.

**Maintainability:**

Our project should be easy to maintain by administrator. After certain of time system should be added with new features so as to make our users more interactive , deal with them according to market and time.

**Portability:**

Our project will be portable on any platform that allows the user to access it easily anywhere and at a faster speed than others.

# **4.Software Process Model**

**4.1 Why not Evolutionary models?**

Evolutionary models are suited for the systems where requirements keep on changing. But for our system, requirements are crystal clear so it is not feasible to adopt any of the evolutionary models.

**4.2 Why not Waterfall model?**

Waterfall model can be adopted because in our case because requirements are known in advance

but there are some limitations of waterfall model due to which it is not feasible to adopt:

* No parallelism of work.
* Time consuming

**4.3 Why Incremental RAD model?**

Incremental model is advisable where requirements are clear and the development time is less. The striking feature of incremental model is that each module can be completed and released as and when requirement arises because of lack of time.

As in our system many of the modules are not inter-related so can be released in isolation. The user can thus get a feel of these modules and give his feedback which can be utilized for making the software more user friendly and inline with the user requirements.

Not only that the deadline set for this project is 3 months and we need a high adaptation model and again will be concentrating on parallelism because our teams will be working on different module on same time and that to faster. So looking into all these requirements we find Incremental RAD model is best suited for our system because it enables the development team to create a fully functional system within very short period of time.

**4.4 Determining Project Feasibility**

The feasibility study is not a full-blown systems study. Rather, the feasibility study is used to gather broad data to decide on whether to proceed with system study. System project feasibility is assessed in three principal ways:

* Economically
* Technically
* Operationally

The organization has evaluated cost of software and hardware required for the system including the storage of data. The benefits expected from the system are studied to assess the reduced cost due to the new system.

**Economic Feasibility:**

The organization has evaluated cost of software and hardware required for the system including the storage of data. The benefits expected from the system are studied to assess the reduced cost due to the new system.

**Technical Feasibility:**

Organization has shown willingness to purchase all hardware and software tools which we recommend to successfully implement the system. Hence technically there are no limitations for the development of the system. As far as programming efforts are concerned, we are familiar with java programming. Thus the project is technically feasible.

**Operational Feasibility:**

Operational feasibility is dependent on the humans who will be using the software once it’s ready and installed for use. The software will have a user-friendly interface which will be much convenient. Thus, the project is operationally feasible.

**5.Overall Design And Implementation**

**5.1 About the Front End**

HTML, CSS and JavaScript

**5.2 About the Back End**

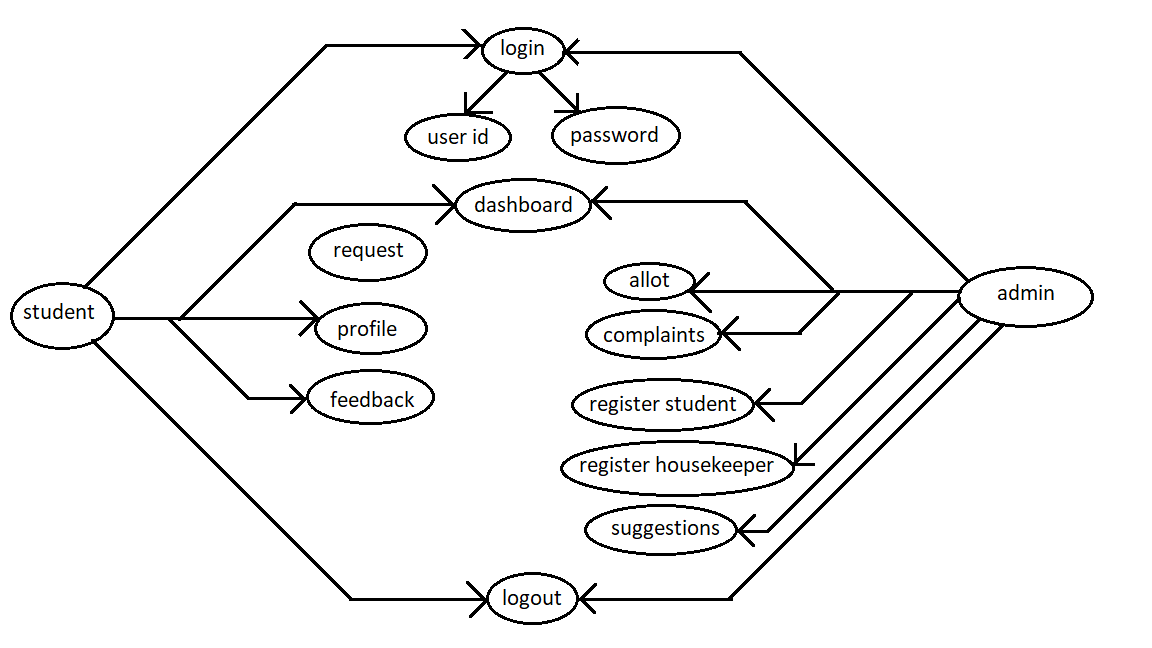
PHP, MySQL and Bootstrap

**6. List of Figures**

1. Use case Diagram
2. Sequence Diagram
3. Class Diagram
4. ER Diagram
5. Data Flow Diagram
6. Activity Diagram

**(A) Use Case Diagram:**

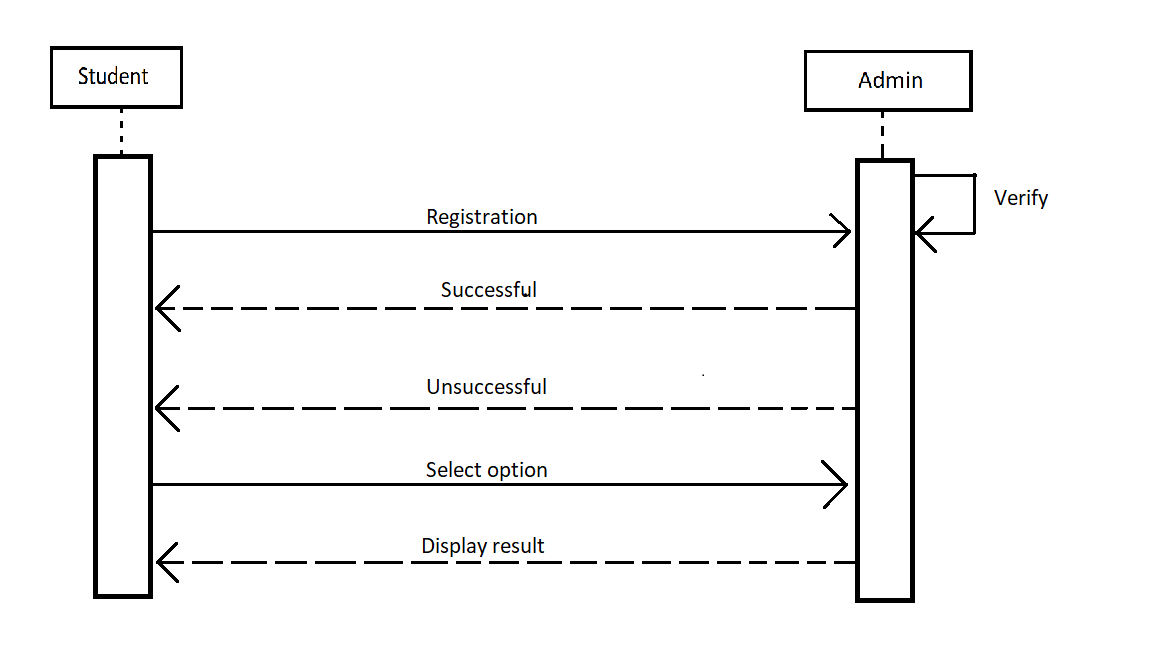
A use case diagram at its simplest is a representation of a user’s interaction the system and depicting the specification of a use case. A use case is a description of how end-user will use a software code. It describes a task or a series of tasks that user will accomplish using the software and include the responses of the software to user actions.



**(B) Sequence Diagram:**

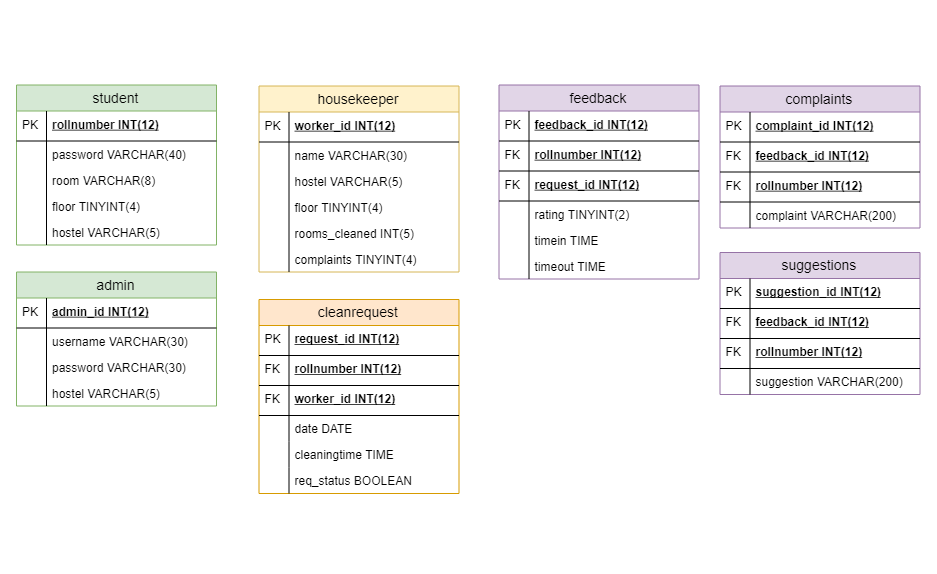
A sequence diagram in unified modelling language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a message sequence diagram are sometimes called event diagram, event scenarios called event diagram, event scenarios and timing diagram.

A sequence diagram shows, as Parallel vertical lines (Lifeline), different processes or objects that live simultaneously and as horizontal arrows, the message exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.



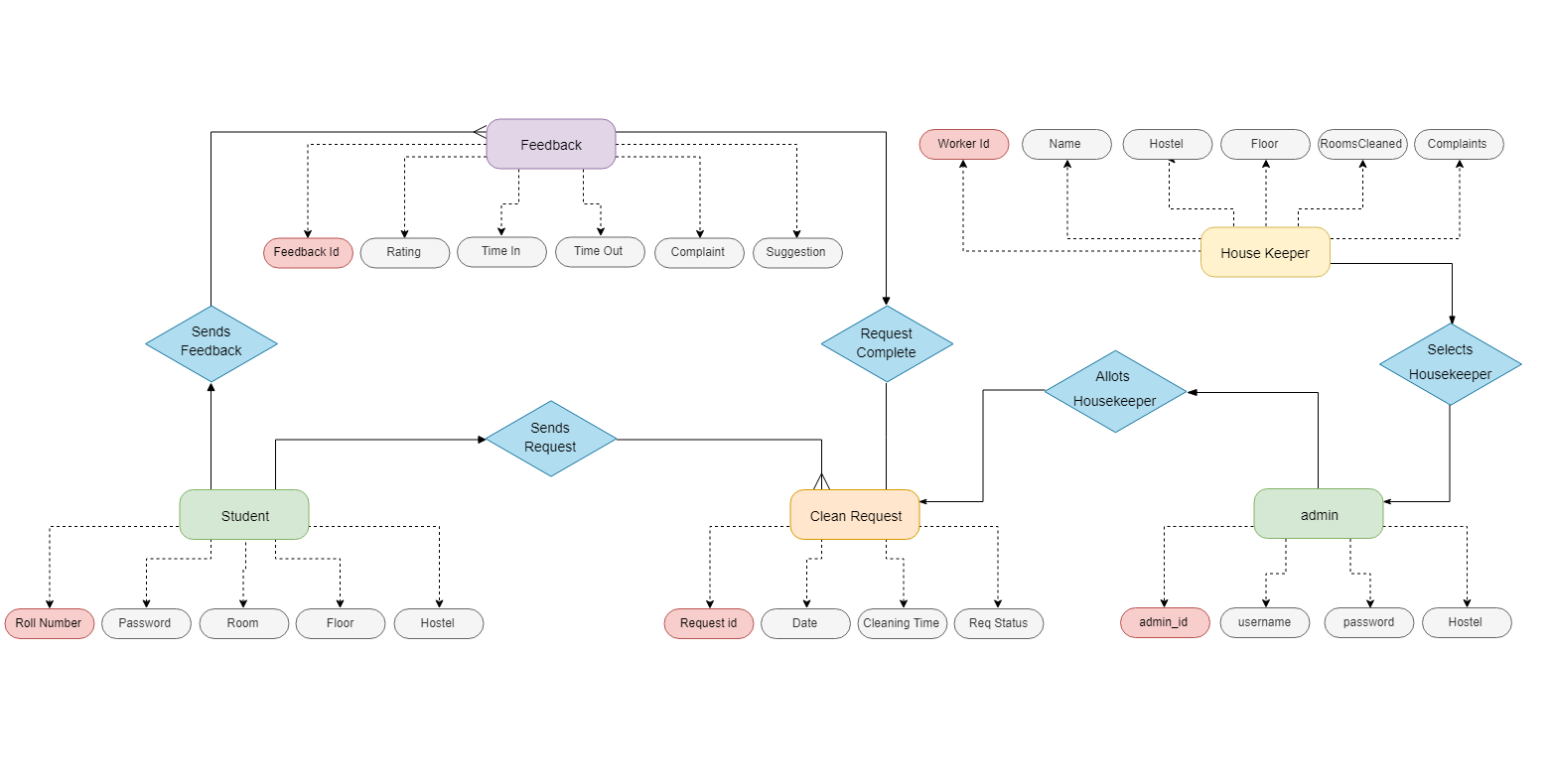
**(C) Class Diagram:**

In the unified modelling language (UML) a class diagram is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attributes, and the relationships between the classes.



**(D) ER diagram**:

An entity relationship(ER) diagram is a graphical representation of an information system that shows the relationship between people, object, places, concepts or event within that system. An ER diagram is a data modelling technique that can help define business process and can be used as the foundation for a relational database.



**(E) DFD:-**

DataFlow Diagram isthe graphical description of the system’s data and how the processes transform the data. The information flow and the transform that are applied as data move from the input to output. It is starting point of the design phase that functionally decomposes the requirement specifications down to the lowest level of details. Thus a DFD describes what data flow(logical) rather than how they are processed.

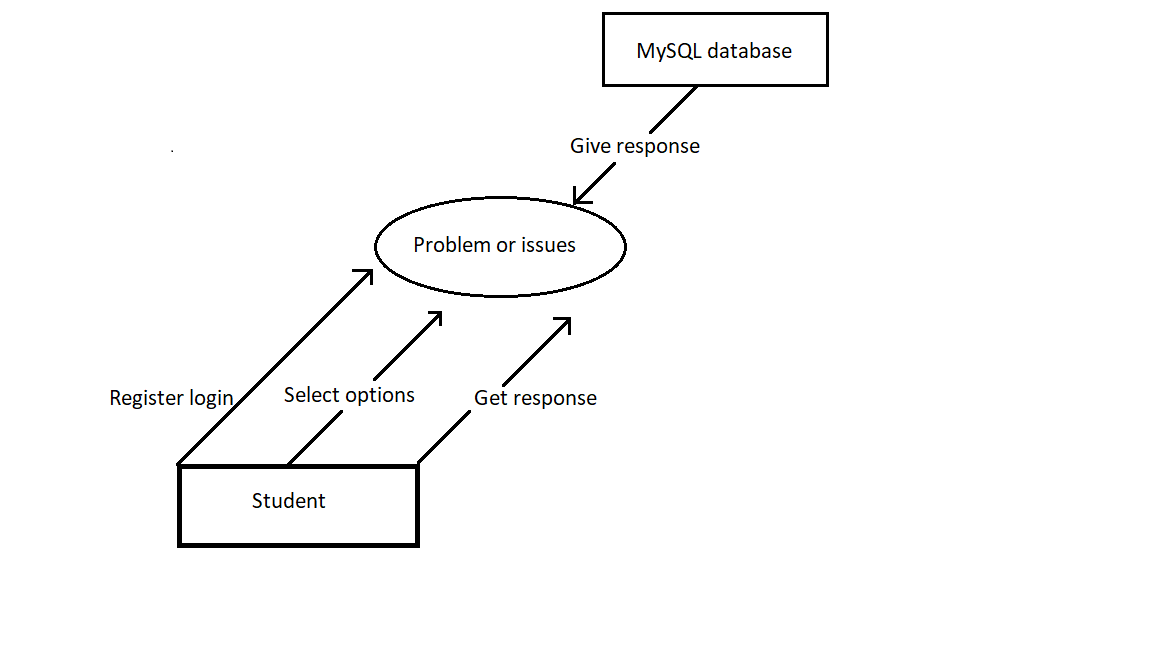
Unlike details flowchart, data flow diagram do no supply detailed description of the module but graphically describes a system’s data interact with the system. to construct a dataflow diagram, we use-

* Arrows
* Circles
* Open end box
* Square

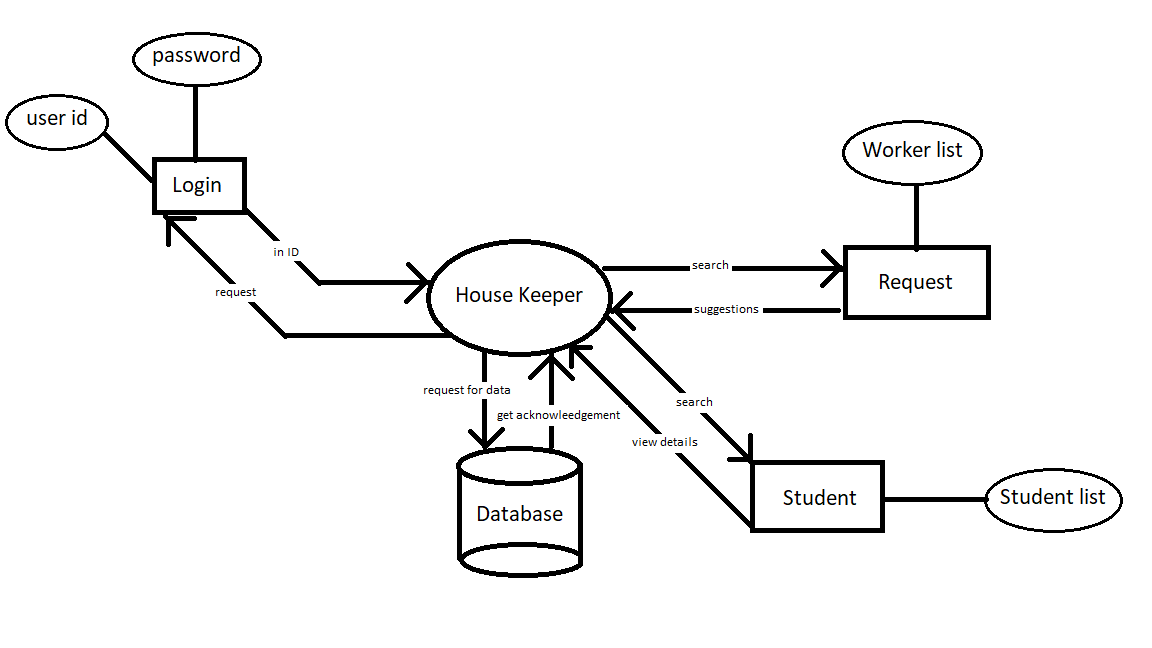
An arrow identifies the dataflow in motion. it is a pipeline though which information is flows like the rectangle in the flowchart. A circle stands for process that converts data into information. An open-ended box represents a data store, Data at rest or a temporary repository of data. Square defines a source or destination of system data.

Rule for constructing a data flow diagram

* Arrows should not cross each other.
* Square, circle and file must be name.
* Decomposed data flow square and circle can have same name.
* Choose meaningful names for dataflow.

****

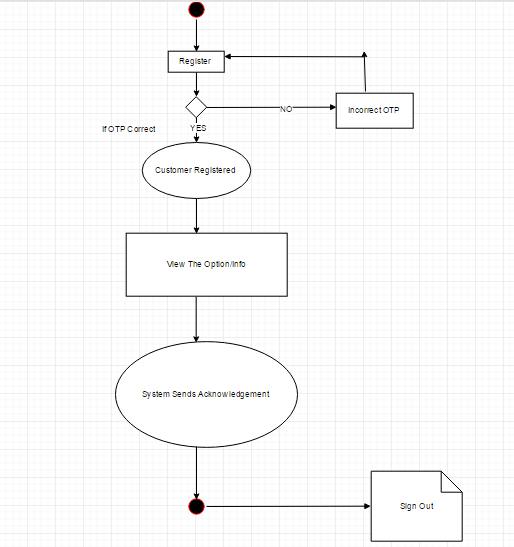
DFD LEVEL 0

****

DFD LEVEL 1

**(F) Activity diagram**

Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system.  It captures the dynamic behavior of the system.

****

**6.2 LIST OF TABLES**

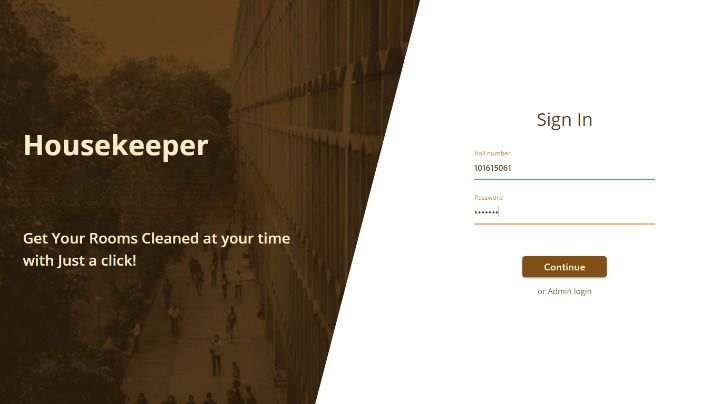
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table 2** | **Customers** | | | |
| Description | This table will contain the information about the Students in the hostel. | | | |
| Primary Key | CID | | | |
| Foreign Key | NONE | | | |
|  | | | | |
| **Field Name** | | **Data Type** | **Constraints** | **Comments** |
| SID | | int | PK | Student Id |
| IMAGES | | Varchar | NN | Student Image |
| AGE | | int | NN | Student Age |
| QLF | | Varchar | NN | Qualification |
| CON | | Varchar | NN | Contact Number |
| EMAIL | | Varchar | NN | Email id |
| ADDRESS | | Varchar | NN | Address |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 1** | **Admin** | | |
| Description | This table will contain the complete information about the admin and provide update ,delete option to admin. | | |
| Primary Key | AID | | |
| Foreign Key | NONE | | |
|  |  |  |  |
| **Field Name** | **Data Type** | **Constraints** | **Comments** |
| AID | int | PK | ID of admin. |
| USERID | Varchar | NN | useriD of admin |
| PASSWORD | Varchar | NN | Password of the admin |

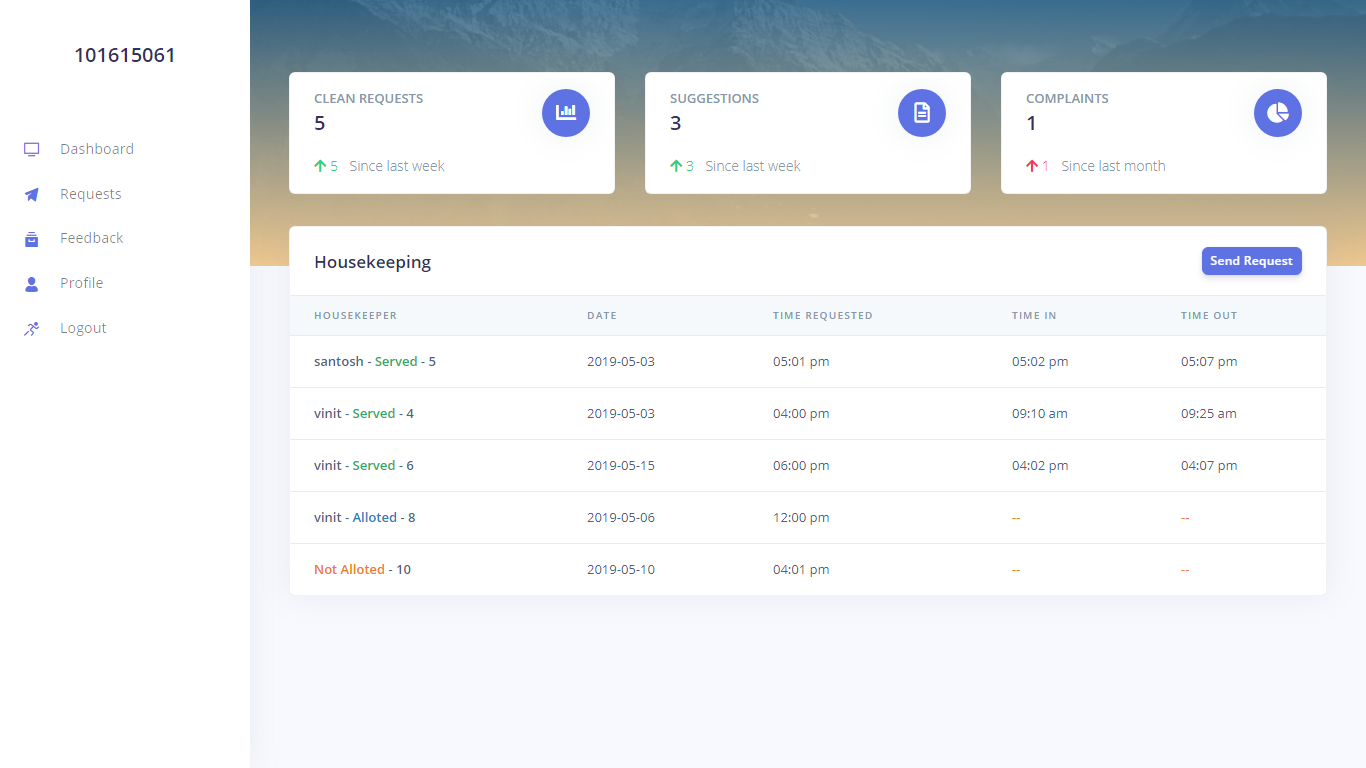
|  |  |  |  |
| --- | --- | --- | --- |
| **Table 3** | **Workers** | | |
| Description | This table will contain the information about the housekeeping workers in the hostel. | | |
| Primary Key | WID | | |
| Foreign Key | NONE | | |
|  | | | |
| **Field Name** | **Data Type** | **Constraints** | **Comments** |
| WID | Int | PK | Worker id. |
| WIW | Varchar | NN | Worker’s Individual Work |
| WNAME | Varchar | NN | Worker’s name. |

**6.3 Screen Shots:**

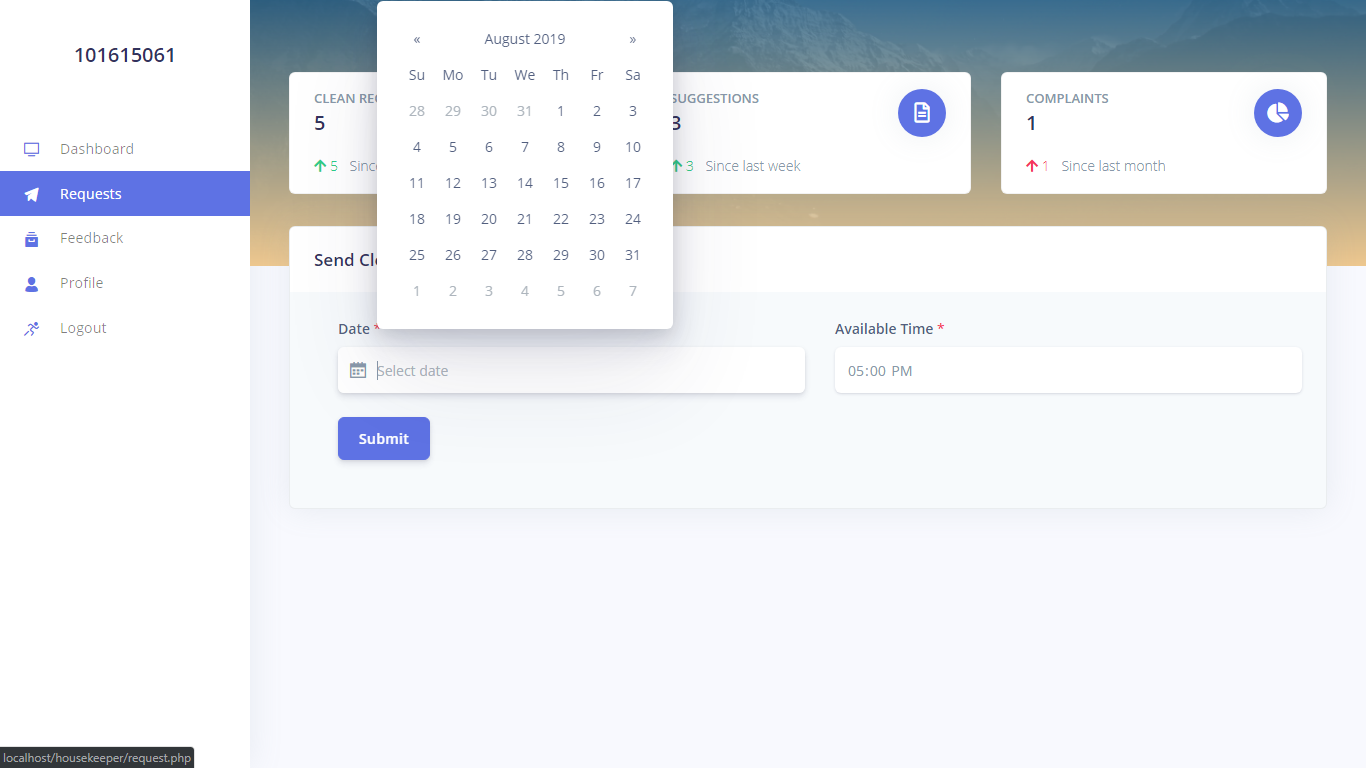
**Home Page**

****

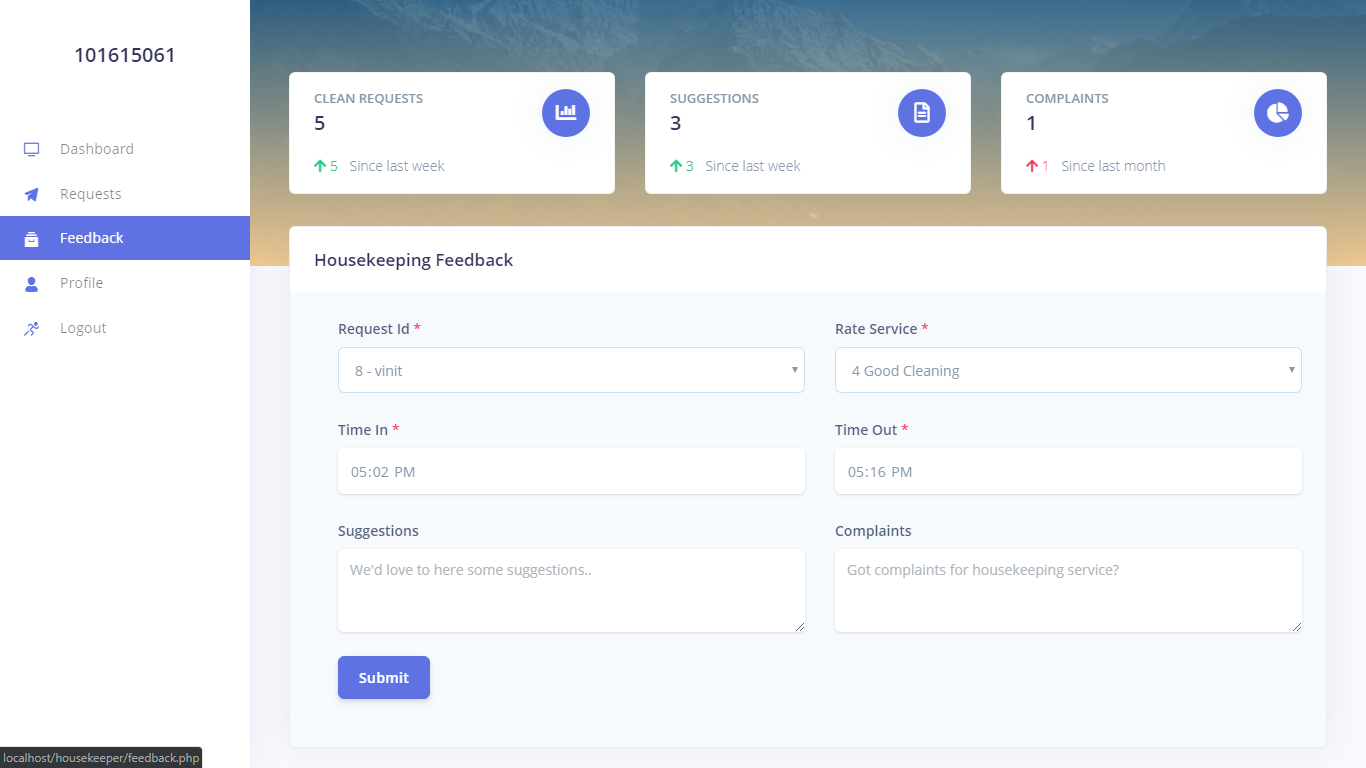
**Dashboard – User Level :-**



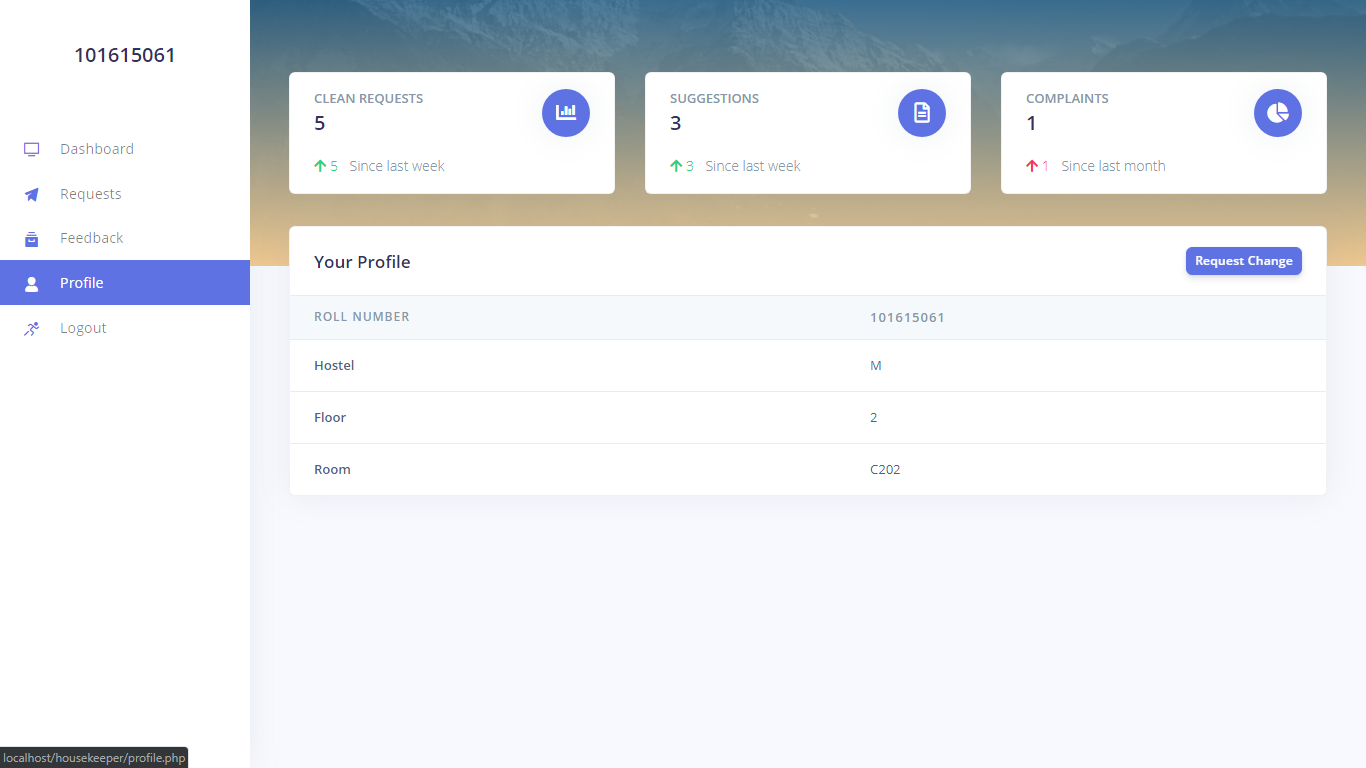
**Request :-**



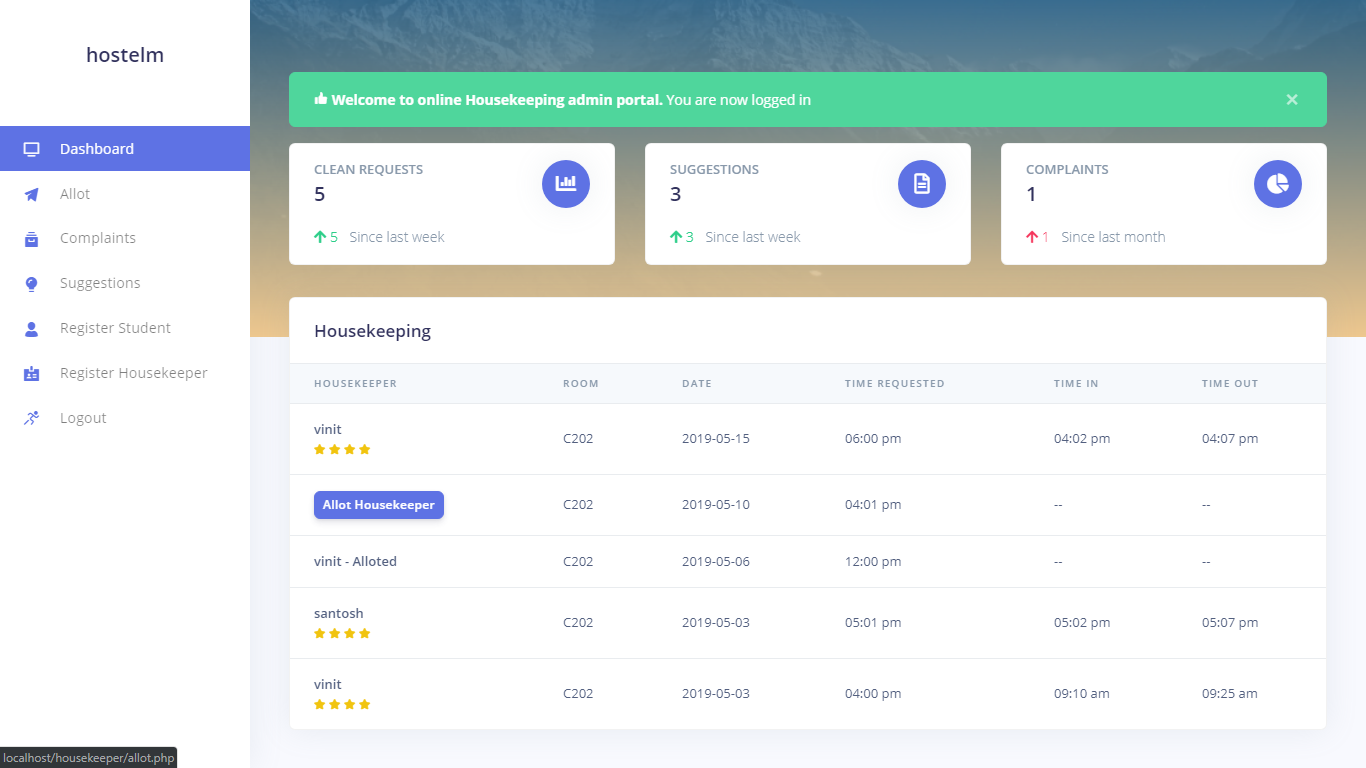
**Feedback :-**



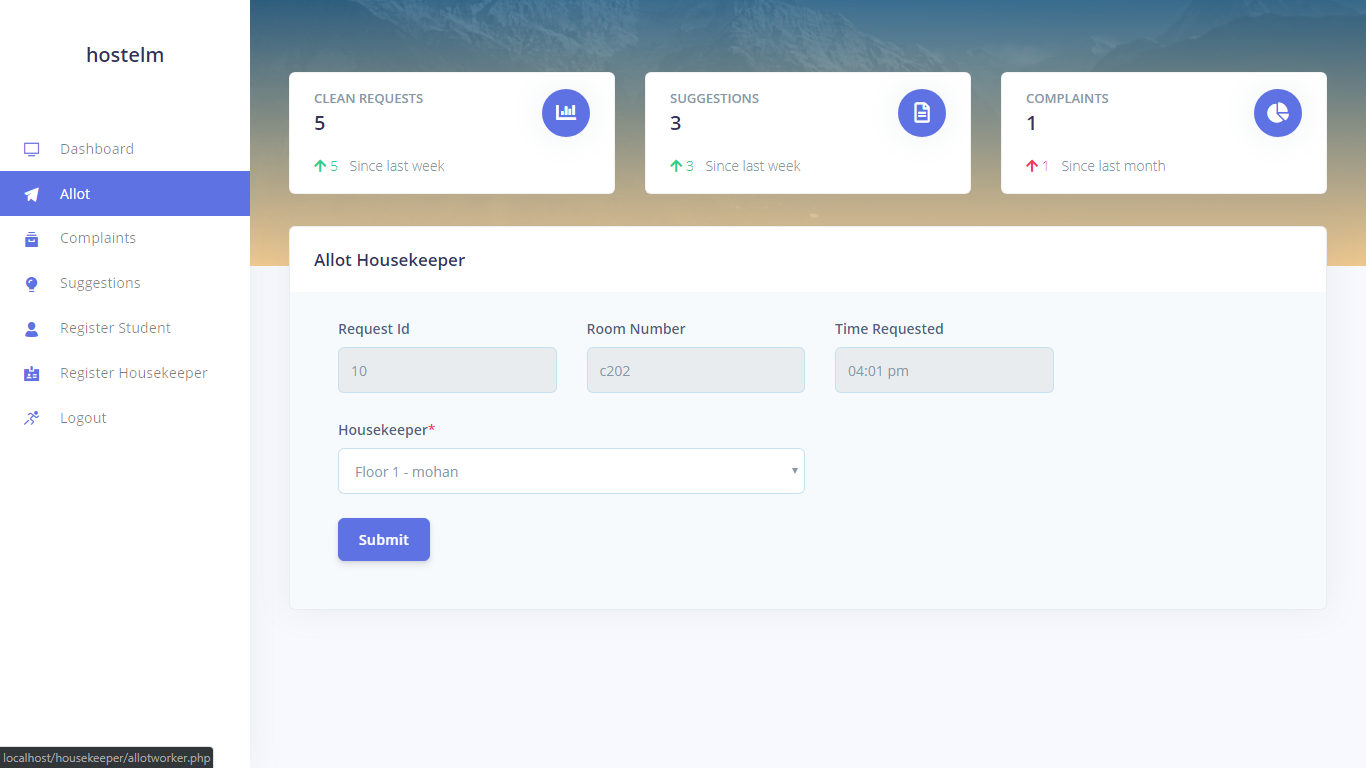
**Profile :-**



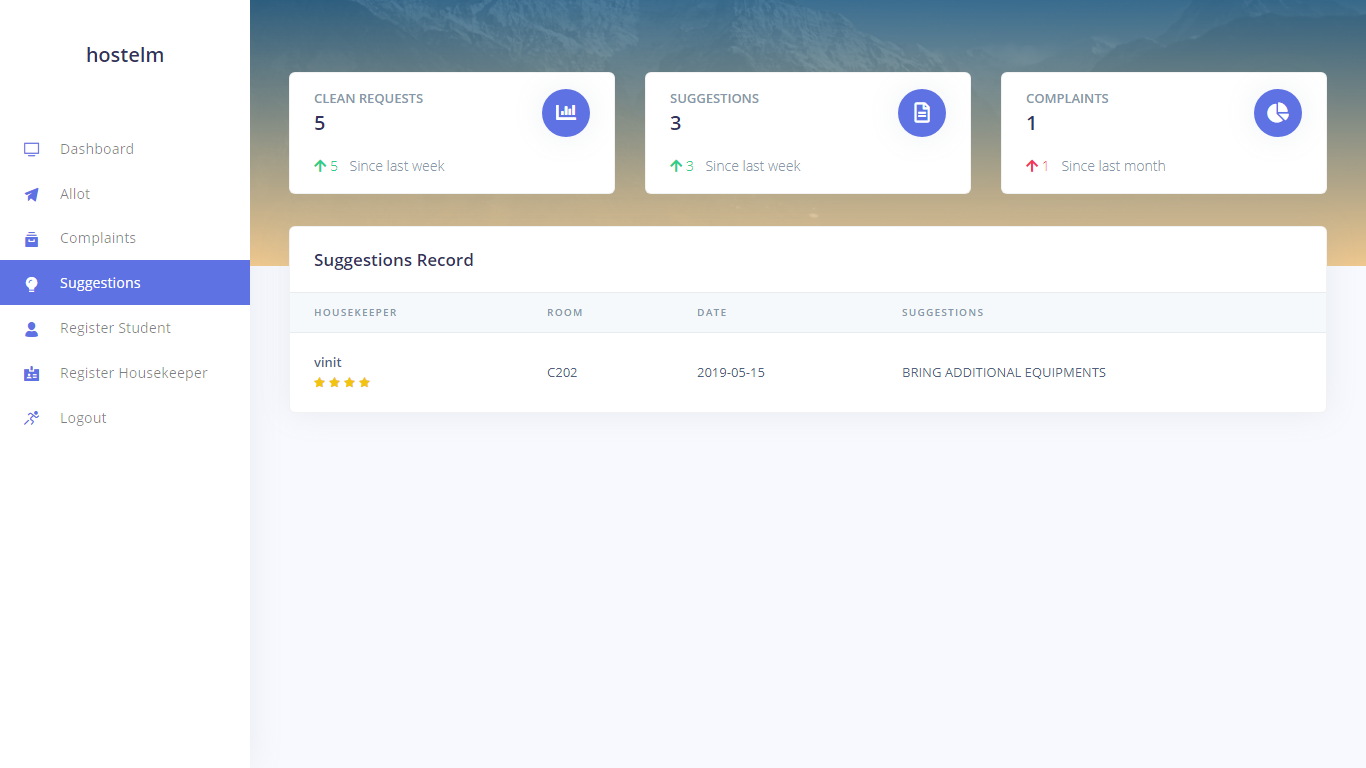
**Dashboard – Admin level:-**



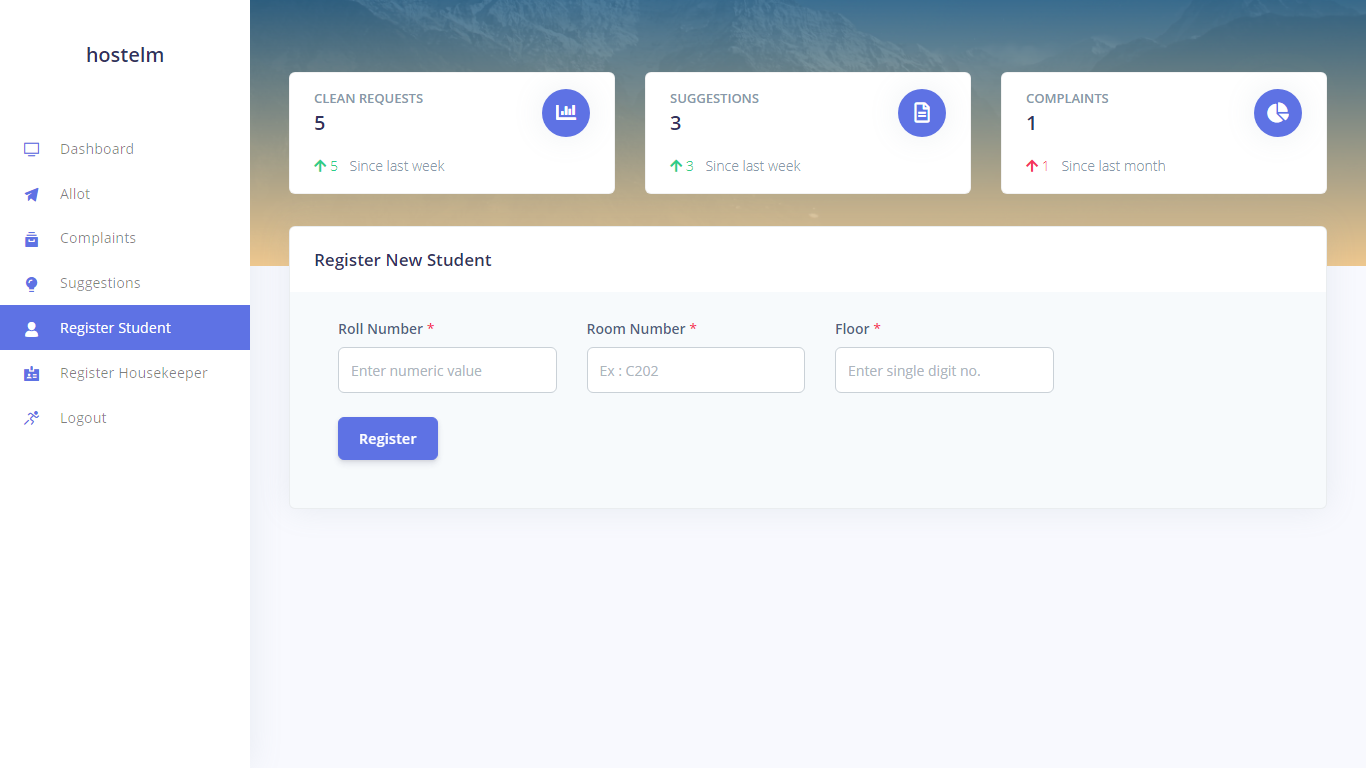
**Allot :-**



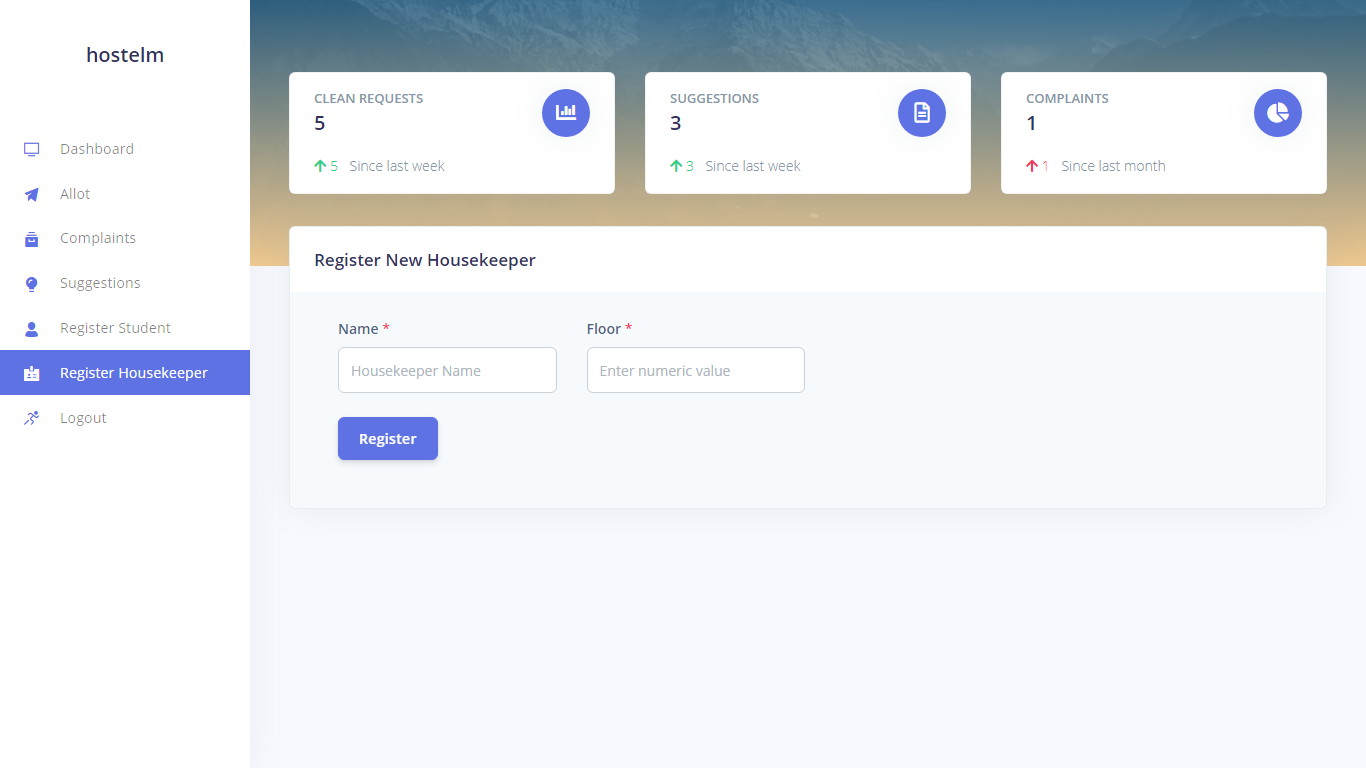
**Suggestions :-**



**Register Student :-**



**Register Housekeeper :-**



**7. TEST CASES**

**7.1. Purpose**

To determine the list of Test Cases that needs to be executed when assuring the system's quality. Each test case is described below in accordance with Test Case Specification.

**7.1.1. Background**

The purpose of project is fight against Brute force hacking that should have a feature of virtually locking a customer account by providing a new concept of authorization (for eg. Email), which can be easily implemented and used, have future scope and flexibility and will provide new way of Store At Your Service account’s security.

**7.1.2. Scope**

Test Cases document contains the list of test cases which has to be executed on the project. All the account users will get benefited.

**7.2. Test Level (Functional Testing)**

Functional testing is a quality assurance (QA) process and a type of black box testing that bases its test cases on the specifications of the software component under test. Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered (not like in white-box testing). Functional Testing usually describes what the system does.

Functional testing differs from system testing in that functional testing "verifies a program by checking it against ... design document(s) or specification(s)", while system testing "validate[s] a program by checking it against the published user or system requirements" (Kaner, Falk, Nguyen 1999, p. 52).

Functional testing means testing the application against business requirements. Functional testing is executed using the functional specifications given by the client or by the design specifications according to use cases given by the design team. Role of functional testing is to validating the behavior of an application.

Functional testing is more important because it always verifies that your system is fixed for release. The functional tests define your working system in a useful manner. In functional testing tester has to validate the application to see that all specified requirements of the client whatever we have said in SRS or BRS have been incorporated or not.

Functional testing is always concentrating on customer requirements and whereas the Non Functional testing is always concentrating on customer expectations.

Functional test cases target business goals and Non Functional test cases target performance, resource utilization, usability, compatibility etc. Functional testing is a part of system testing.

Example of functional testing is explained below –

Considering example if you are functionally testing a word processing application, a partial list of checks you would perform minimally includes creating, saving, editing, spell checking and printing documents.

**7.3. Types of Functional Testing**

Functional testing falls in to two categories –

**7.3.1. Positive functional testing**

This testing carries exercising the application’s functions with valid input and also verifying that the outputs are correct.

Example - Again continuing with the word processing example, a positive test for the printing function might be to print a document containing both text as well as graphics to a printer that is online, filled with paper and for which the correct drivers are installed.

**7.3.2. Negative functional testing**

This testing involves exercising application functionality using a combination of invalid inputs, some unexpected operating conditions and by some other “out-of-bounds” scenarios.

Example - Again continuing with the word processing example, a negative test for the printing function might be to disconnect the printer from the computers while a document is printing. What probably should happen in these scenarios are a plain-English error message displayed, informing the user what happened and instructing him/her on how to fix the problem.

**7.3.3. Conclusion**

At last we conclude that in functional testing functionality of the module is tested and structure is not considered. It is performed, based on user's perspective. These tests ensure that the system does what users are expecting it to do.

This type of testing means testing the functionality which include input the proper data and checking the output as per the requirement documents.

**7.4.1 MODULE 1 (ADMIN LOGIN)**

|  |  |
| --- | --- |
| Module Name: | Login |
| Dependencies: | Connection with Database |
| Screen Name: | Main Page |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case# | Test Objective | Precondition | Test Steps | Test Data | Expected Result | Actual Result | Status |
| FT001 | Successful administrator sign in to SAYS | Administrator User Id and its password | In the sign in Panel, enter the user id and password and Click "Log In” | "admin",” admin” | Check whether no same ID is already present | Successfully sign in | Pass |

**7.4.2 MODULE 2 (STUDENT)**

|  |  |
| --- | --- |
| Module Name: | Student |
| Dependencies: | Connection with Database |
| Screen Name: | Main Page |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case# | Test Objective | Pre - condition | Test Steps | Test Data | Expected Result | Actual Result | Status |
| FT002 | To view student list of the database | \_ | In the Student page, choose option and click on "search” | Show list of Students | List of Students with details present in the database | Successfully view the List of all the Students | Pass |

**7.4.3 MODULE 3 (WORKERS)**

|  |  |
| --- | --- |
| Module Name: | Workers |
| Dependencies: | Connection with Database |
| Screen Name: | Main Page |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case# | Test Objective | Pre - condition | Test Steps | Test Data | Expected Result | Actual Result | Status |
| FT003 | To view workers list and their individual work of the database | \_ | In the workers Page, choose option of request and click on "search” | Show list of workers with chosen request. | List of workers with details present in the database. | Successfully view the List of all the Workers | Pass |

**7.4.4 HOME PAGE**

|  |  |
| --- | --- |
| Module Name: | Home Page |
| Dependencies: | Put server online |
| Screen Name: | Main Page |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case# | Test Objective | Pre - condition | Test Steps | Test Data | Expected Result | Actual Result | Status |
| FT004 | To view Dashboard |  | In the home Page, click "dashboard” |  | Details about the current status | Successfully view the current status | Pass |
| FT005 | To view the Profile page. |  | In the home Page, click "Profile” |  | Details about the Profile | Successfully view the Profile details of Students | Pass |
| FT006 | To view the Request page |  | In the Home Page, click on "Request” |  | Different requests are visible | Successfully all requests appear | Pass |
| FT007 | To view the feedback page |  | In the Home Page click on “Feedback” |  | Comment box will appear for the feedback | Successfully feedback inputs are visible | Pass |

**8. Future Enhancements:**

Housekeeper is well documented, scalable and maintainable application. It is optimized for speed but kept simple to use and understand. So it could be upgraded very easily. These features can be add in future

* Automatically assign the Workers according to the user’s past record of housekeeping
* Workers will also have the access to the application to keep the record.
* Convert into real time web-based application.

**9. Conclusion:**

The module developed by us is very handy, user friendly and efficient. It is a very stable, less costly and has a lot of features like:

* Fast to access
* Easy for modify and add into database
* Highly comfortable and user friendly

The application made by us is perfectly suited to implement all these features. The system include security, this is secure from unauthorized person to use. Easy navigation is provided by us in whole project. We conclude that the functionalities done were well implemented successfully. We implemented housekeeping system which ensures the basic functionalities for the users. The students can successfully get the quick solution for their problems.

**10. References**

Information from Web sites:

* https://www.youtube.com/
* https://www.google.com/
* https://www.wikipedia.org/
* https://www.w3schools.com/
* https://www.mysql.com/
* https://getbootstrap.com/