

SERVICE HUB

done by

Alwin Paul

Reg No:180021092593

Under the guidance of

Ms. NIBY BABU

*In partial fulfillment of the
requirements for the award of
the Degree of*

Bachelor of Computer Applications

Of

***Mahatma Gandhi University, Kottayam,
Kerala***



Department of Computer Science

**NIRMALA COLLEGE,
MUVATTUPUZHA**

(Affiliated to Mahatma Gandhi University)

2018-2021



CERTIFICATE

Certified that this is a bonafide report on the project work entitled

.....

done by.....

Reg.No......

*during the year.....in the partial fulfillment of the
requirements for the award of the degree of Bachelor of Computer
Applications of Mahatma Gandhi University, Kottayam, Kerala.*

Internal Guide

Head of the Department

Submitted for the Viva-voce Examination held on.....

External Examiner

ACKNOWLEDGEMENT

We give all honor and praise to the Lord who gave wisdom and enable to complete our project successfully.

We express our sincere and heartfelt thanks to respected **Dr. Thomas K.V**, our| beloved principal for providing necessary facilities for the completion of our project successfully.

We think profusely **Ms. Arya S Nair**, Course coordinator of Computer Science| Department for her guidance and inspiration throughout our course of study.

We express our sincere gratitude to our project guide, **Ms. Niby Babu**, Assistant| Professor in Computer Science Department for the valuable advices and guidance throughout the completion of our project.

We also express our gratitude and thanks to all my teachers and friends for their sincere and friendly cooperation in the successful completion of our project.

CONTENTS

1. Introduction	1
2. System Analysis	2
2.1 Existing System	2
2.2 Proposed System	3
2.3 System Requirement Specification (SRS)	
2.3.1 Hardware specification	4
2.3.2 Software specification	4
2.3.3 Front End	5
2.3.4 Back End	5
2.4 Feasibility Analysis	6
2.5 Data Flow Diagram	
2.5.1 Introduction	7
3. System Design	
3.1 Module description	14
3.2 Input Design	15
3.3 Database Design	16
3.4 Table Design	17
3.5 Output Design	21
4. System Testing	22
5. Implementation	
5.1 System Implementation	25
5.2 System description	25
6. Maintenance	27
7. Conclusion	29
8. Bibliography	30
9. Appendix	
9.1 Screenshots	31
9.2 Code	32

1. INTRODUCTION

In this application we can get the details of authorized service centers of various brands & devices like Apple, Samsung, Asus, HP, DELL and the services provided by these centers. We can also get the details of troubleshooting that is when we enter the complaint details and answer some questions then the system will give the basic trouble shooting methods. It helps the users to rescue the system from small issues. And after the question answer section the complaint is not a small one then the system will show the nearest location and show the route. The customer can book the service through online and know the status through online.

Management of information of different types of service centres is often difficult for a service receiver due to communication gap between provider and receiver. As a result, not only service receivers face an unexpected trouble but also service providers fail to get their jobs. A generic platform providing two-way communication between service provider and service receiver can eliminate the exhausting efforts of searching the suitable service provider and also ensures the intended job for the service provider. A feedback-based rating system can improve the rating of any service provider and service receiver can eventually get better service.

2. SYSTEM ANALYSIS

System analysis is a detailed study of the various operations performed by the system and their relationship within and module of the system. It is a structured method for solving the problems related to the development of a new system. The detailed investigation of the present system is the focal point of system analysis. This phase involves the study of parent system and identification of system objectives. Information has to be collected from all people who are affected by or who use the system. During analysis, data are collected on the variable files, decision point and transactions handled by the present system. The main aim of system is to provide the efficient and user-friendly automation. So, the system analysis process should be performed with extreme precision, so that an accurate picture of existing system, its disadvantages and the requirements of the new system can be obtained.

In this phase, the problem is identified and an alternate system solution for solving it is recognized. System Analysis not only includes the process of synthesis, which is a process of putting parts together to form a new product, but also the requirement identification and specification. The System Analysis include the studying of the existing system and its drawback, designing a new system and conducting cost benefit analysis. The costs and benefits of each alternative guide the selection of the best system for the job. System analysis is a problem-solving activity that requires intensive communication between the system user and the system developer.

2.1 EXISTING SYSTEM

Now days if we want to know the service centers, we can search through Google then we can get the site name of particular service agency or its Phone number or it address and we directly go to the service center and register the complaints. Or if we want to get some ideas for basic trouble shooting, we can search in YouTube and get the video about the trouble shooting but most of these cases it is not full fill our needs.

Limitations of Existing System

- Time consuming
- Slow response
- Basic trouble shooting must not full fill our needs
- Communication gap between the service provider and the customer
- Needs more time for processing

2.2 PROPOSED SYSTEM

In proposed system our site solves all these problems. We introduced an intelligent troubleshooting method if we enter our complaint to the system the system automatically gives some trouble shooting technique according to system. If we can't recover the system from complaint then the complaint is automatically register to the nearest service center and get the id no and also the user can track the complaint status by getting alerts through SMS and mail.

ADVANTAGES OF PROPOSED SYSTEM

- Provides tracking feature to track the service
- Added FAQ's section
- Customer could upload the image of the issue
- Service center and customer can communicate with each other
- Reliable
- Customer could provide feedback based on the service provided

2.3 System Requirement Specification (SRS)

2.3.1 Hardware Specification

Microprocessor	: Any 64-bit processor
Speed	: 2.44 GHz
RAM	: 256 MB and above
Hard Disk	: 40 GB and above
Key board	: 108 key standard key board
Mouse	: Standard Mouse
Device	: Serial Communication Device

2.3.2 Software Specification

Platform	: Windows
Front end	: Python
Scripting Language: JavaScript, HTML	
Back end	: MySQL
IDE	: Visual Studio Code

2.3.3 Front End: PYTHON

Python is an interpreter, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

2.3.4 Back End: MySQL

MySQL is the world's most popular open-source database. With its proven performance, reliability and ease-of-use, MySQL has become the leading database choice for web-based applications, used by high profile web properties including Facebook, Twitter, YouTube, Yahoo! and many more. Oracle drives MySQL innovation, delivering new capabilities to power next generation web, cloud, mobile and embedded applications. They differ in the

scope of services and in price. Additionally, a number of third-party organizations exist to provide support and services, including Maria DB and Persona.

2.4 FEASIBILITY ANALYSIS

Feasibility study is carried out to select the best system that meets performance requirements. This entails identification, description, and evaluation of candidate systems and selection of the best system for the job. The main aim of the feasibility study is to evaluate alternative system and to propose the most feasible and desirable system for development. During the detailed study conducted in existing system, I have noticed several problems. It is a time consuming and a job pending one. After Initial investigation culminates in a proposal that determines whether an alternative system is feasible.

Mainly there are four types of feasibility analysis:

- Economic Feasibility
- Technical Feasibility
- Behavioral Feasibility
- Operational Feasibility

Economic Feasibility

Economic feasibility analysis is also known as cost/benefit analysis. The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. The proposed system reduces the operating cost in terms of time by automating the process. This system is economically feasible.

Technical Feasibility

Technical feasibility centers on the existing system and to what extent it can support the proposed addition. This system must be evaluated from technical view point first. The assessment of this feasibility must base on the outline design of the system requirements in the terms of inputs, outputs program procedure. The minimum requirement for this proposed system is a system. The existing system has this entire requirement. So, the proposed system is said to be technically feasible.

Behavioral Feasibility

People are inherently resistant to changes and computer is known for facilitating the changes. An estimate should be made to how strongly the users react towards the development of the system. The proposed system consumes less time. Thus, the people are made to engage in some other important work.

Operational Feasibility

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

2.5 DATAFLOW DIAGRAM

2.5.1 Introduction to Data Flow Diagram

A graphical representation is used to describe and analyze the movement of data through a system manual or automated including the processes, storing of data and delays in system.

Data flow diagrams are central tool and the basis from which other components are developed. The transformation of data, from input to output through process maybe described logically and independently of the physical components associated with the system. They are termed logical data flow diagrams, showing the actual implementation and the movement of data between people, departments and workstations. DFD is one of the most important modelling tools used in system designed shows the flow of data through different process in system purpose.

Throughout my project, the context flow diagram, flow charts have been extensively used to achieve the successful design of the system. The efficient design of the dataflow and context flow diagram helps to design the system successfully without much major flows within the scheduled time. This is the most complicated part in project. The designing process, my project took more than the activities in the software life cycle. Structured analysis is a set of techniques and graphical tools that help the analyst to develop a new kind of system specification that are easily understandable to the user.

DFD's show the major decompositions of the system functions and their interfaces. The DFD is graphic and presents a picture of what is being specified and is conceptually easy to understand presentation of the application.

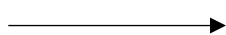
One important feature of DFD's is that it is logical rather than physical. The elements of the system do not depend on vendor or hardware. They specify in precise, concise manner the working of the system and how it hangs together.

DFD is the graphic representation of data movement process, and files used in support of an information system. There are several rules of thumb used in drawing DFDs.

- Process should be named and numbered for easy references.

- The direction of flow is from top to bottom and from left to right.
- When a process is imported in the lower-level details, they must be numbered.
- Process and data flow names have the first letter of each word must be a capital letter

The four main symbols used for developing a DFD are shown below:



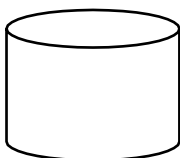
Data flow



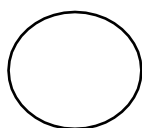
Data Source or Destination



Table



Data Store

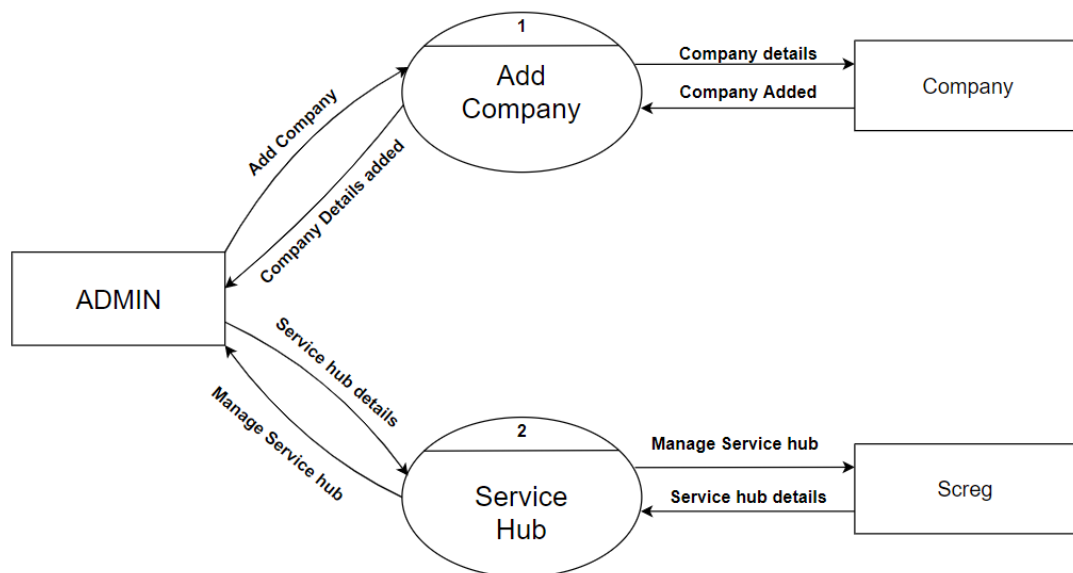


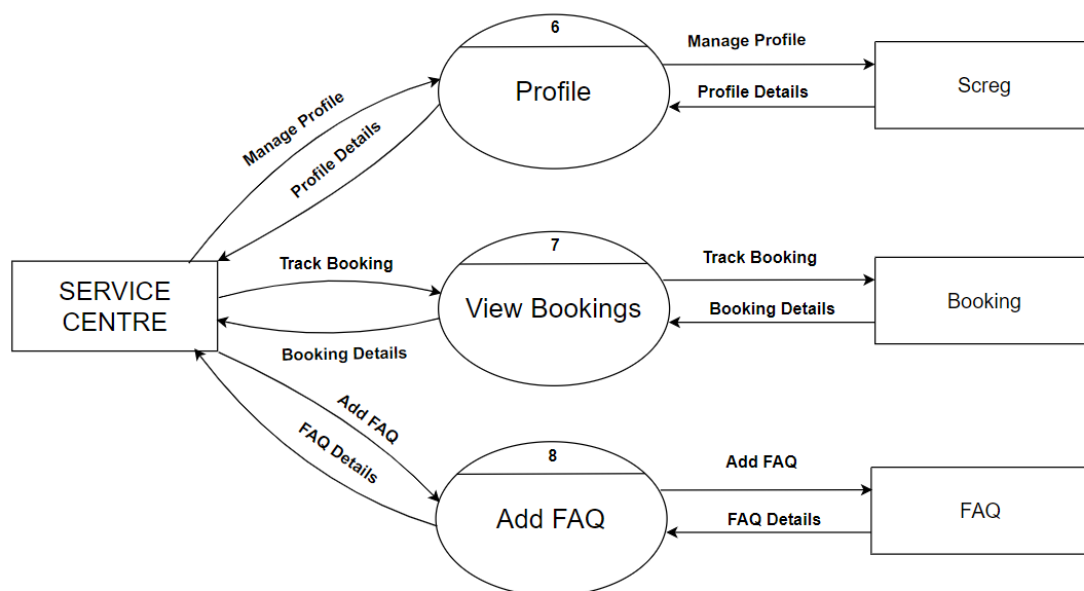
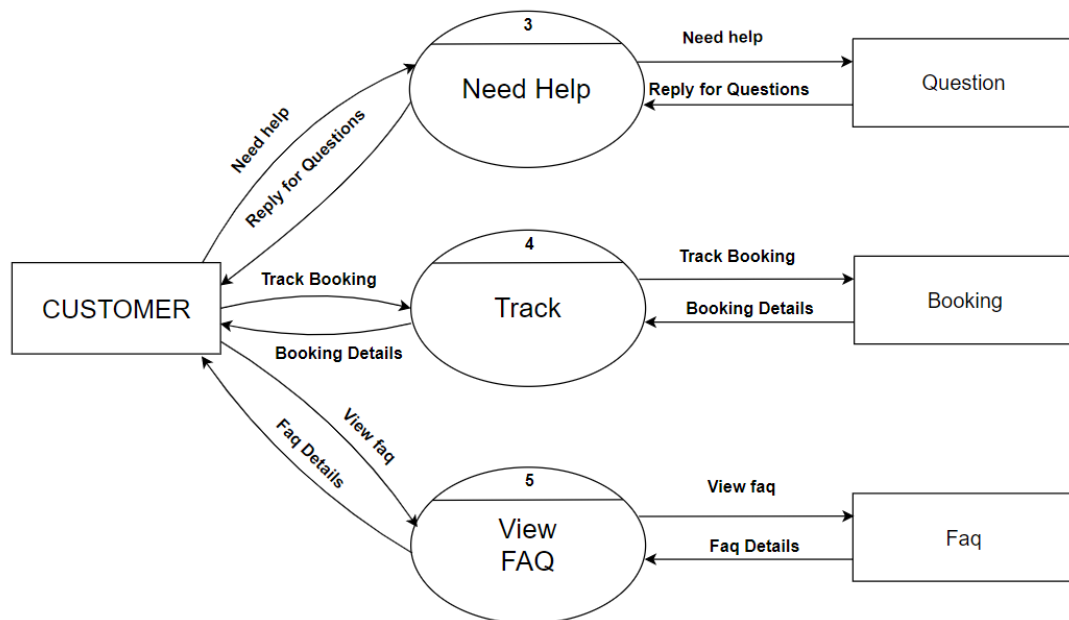
Represents a process that transforms data streams

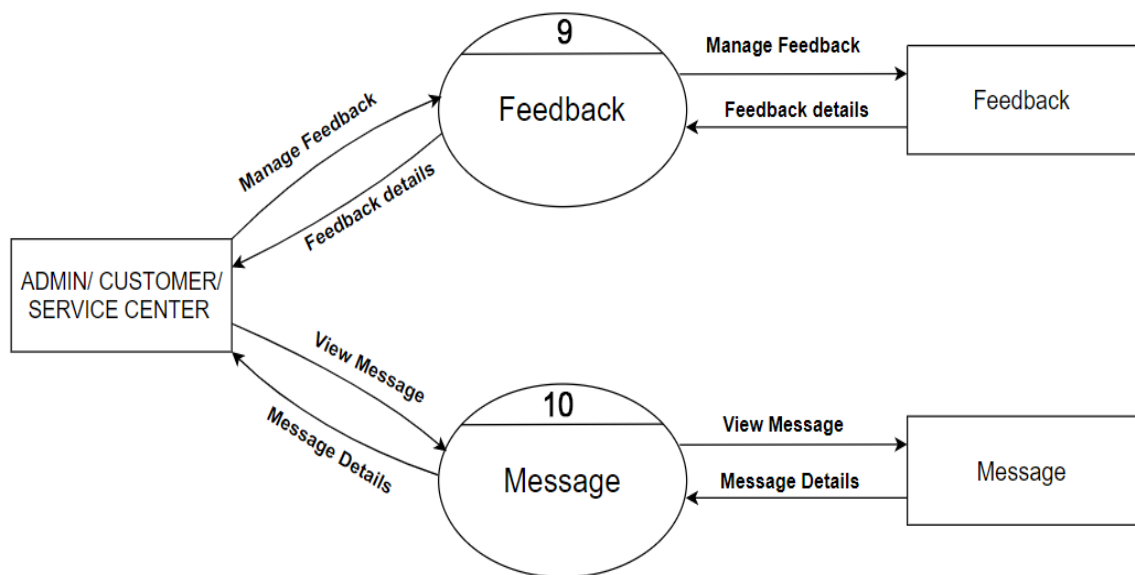
Level 0 DFD



Level 1 DFD







3. SYSTEM DESIGN

System design is the creative act of invention developing new inputs, a database offline files, method, procedures and output for processing business data to meet organization objectives. The design phase focuses on the detailed implementation of the system recommended in the feasibility study. The design phase is a transition from user-oriented document to a document oriented to the programmers or database personnel. Characteristics of a well- designed system are:

- Accessibility
- Decision making ability
- Economy
- Flexibility
- Reliability
- Simplicity

The design will determine the success of the system. System design is based on the information gathered during system analysis. System design goes through two phases of development.

- Logical Design –DFD shows the logical flow of a system and defines the boundaries of the system. For the candidate system it describes the inputs, outputs, databases and procedures-all in a format that meets the user's requirements.
- Physical Design - This produces the working system by defining the design specification that tells programmers exactly what the candidate system must do.

3.1 MODULE DESCRIPTION

Registration

In this module users that is service agents, customers can register to the site to get and provide the services.

User Module

i. Search for online service

It is an intelligent module for get online help to the user for repair the system. In this module user can get basic trouble shooting .by asking some questions to the user.

ii. Service booking module

In this module the complaint is not maintained by the basic trouble shooting. Then the system can search the nearest service center. And book the service.

iii. Alert module

In this module user can get alerts from the service center about the status of the complaints

Service Centre Module

i. View Complaint module

In this module admin can view the complaints and make dissection on the complaints and send alert to them.

ii. Add Troubleshooting

In this module they can add trouble shooting methods and add FAQ

Admin Module

i. Approve module

In this module Admin can check and verify the service center.

ii. Feedback module

In this module Admin can check the feedback from the users.

Intelligent Tracker

In this module user can track the status updates of their services and Spare's movement using data mining technology

3.2 INPUT DESIGN

Input is the process of converting user inputs in computer-based formats. The project requires a set of information's from the user to prepare a report. In the order, when organized input data are needed.

In the system design phase, the expanded DFD identifies logical data flow, data stores and destination. The input of the system includes registration of migrant workers, society, contractors and even police station registration. The goal behind designing input data is to make the data entry easy and make it free from logical error. The input entry to all types of clients is the username and password. If they are valid, they are allowed to enter into the website.

OBJECTIVES

- To produce cost effective method of input.
- To achieve the highest possible level of accuracy.
- To ensure that the input is acceptable and understandable.

Here in our system our website has to collect all the details of the migrant workers. As we know a wide range of personal details are necessary as input. Every entered input is validated before store. These designs are mentioned in the appendix.

3.3 DATABASE DESIGN

The data design transforms the information domain model created during analysis into the data structures that will be required to implement the software. The data objects and relationships defined in the entity relationship diagram and the detailed data content depicted in the data dictionary provide the basis for the data design activity. The overall objective in the development of database technology has been to treat data as an organizational resource and as an integrated whole. Database Management System allows data to be protected and organized separately from other resources. Database is an integrated collection of data. This is the difference between logical and physical data.

The organization of data in the database aims to achieve three major objectives:

- Data integration
- Data integrity
- Data independence

The databases are implemented using a DBMS package. Each particular DBMS has unique characteristics and general techniques for database design. There are 6 major steps in design process. The first 5 steps are usually done on paper and finally the design is implemented.

- Identify the table and relationships
- Identify the data that is needed for each table and relationship
- Resolve the relationship
- Verify the design
- Implement the design

The database uses tables for storage. A table also contains records, which is a set of fields. All records, in a table have the same set of fields with different information. Each table contains key fields that establish relationships in the database and how the records are stored. There are primary key fields that uniquely identify a record in a table. There are also fields that contain the primary key from another table called foreign keys.

The various database tables that are used in this project are the following:

TABLE DESIGN

Table Name: SCREG

Description: Store the registration details of Service Centre

Primary Key: scid

Field Name	Data Type	Size	Description
scid	int		Service Centre ID
name	varchar	20	Name of Service Centre
email	varchar	20	Email of Service Centre
address	varchar	20	Address of Service Centre
phoneno	int		Phone Number of Service Centre
company	varchar	20	Company which the Service Centre is authorized to

product	varchar	20	Product which the service center services
aid	int	20	Authorization ID of Service Centre
password	varchar	20	Password
district	varchar	20	District in which Service Centre is located

Table Name: USERREG**Description:** Stores the registration details of the user**Primary Key:** uid

Field Name	Data Type	Size	Description
uid	int		User ID
name	varchar	20	User's Name
email	varchar	40	Email of User
address	varchar	30	Address of User
phoneno	bigint	8	Phone Number of User

Table Name: LOGIN**Description:** Login Details**Primary Key:** uname

Field Name	Data Type	Size	Description
uname	Varchar	50	User name
password	Varchar	12	Password of User
usertype	bigint	8	Phone number
status	varchar	50	Email Id

Table Name: BOOKING**Description:** To store booking details**Primary Key:** bid**Foreign Key:** company, scid

Field Name	Data Type	Size	Description
bid	int		Booking ID
company	varchar	20	Company which the Service Centre is authorized to
bookingdate	date		Booking date
problem	varchar	300	Problem faced
image	varchar	500	Image of the issue
status	varchar	30	Status of the service
userid	int		User ID
scid	int		service center ID
enddate	date		Product return date

Table Name: Company**Description:** Stores details of Companies**Primary Key:** cid

Field Name	Data Type	Size	Description
cid	int		Company ID
companyname	int		Company Name
photo	Varchar	300	Logo of Company

Table Name: FAQ**Description:** Stores Frequently Asked Questions added by Service Centers**Primary Key:** id

Field Name	Data Type	Size	Description
id	int	8	FAQ ID
company	varchar		Company which the Service Centre is authorised to
faq	varchar	100	Frequently Asked Question
answer	varchar	200	Solution to FAQ

Table Name: FEEDBACK**Description:** To store the feedback of customers on service received**Primary Key:** fid**Foreign Key:** userid, scid

Field Name	Data Type	Size	Description
fid	int		Feedback ID
feedback	varchar	100	Feedback to be given by customer
userid	int		User ID
scid	int		Service Centre ID

Table Name: MESSAGE

Description: To store messages sent between customer and service center

Primary Key: id

Field Name	Data Type	Size	Description
id	int		Message ID
message	varchar	300	Message
messenger	varchar	50	Sender of message
recipient	varchar	50	Receiver of message

3.5 OUTPUT DESIGN

One of the important features of an information system for users is the output produces. Output is the information delivered to users through the information system. Without quality of the output, the entire system appears to be unnecessary that users will avoid using it. Users generally merit the system solely by its output. In order to create the most useful output possible. One works closely with the user though an interactive process. until the result is considered to be satisfactory.

4. SYSTEM TESTING

Testing is the process of examining the software to compare the actual behavior with that of the expected behavior. The major goal of software testing is to demonstrate that faults are not present. In order to achieve this goal, the tester executes the program with the intent of finding errors. Though testing cannot show absence of errors but by not showing their presence it is considered that these are not present. System testing is defined as the process by which one detects the defects in the software. Any software development organization or team has to perform several processes. Software testing is one among them. It is the final opportunity of any programmer to detect and rectify any defects that may have appeared during the software development stage. Testing is a process of testing a program with the explicit intention of finding errors that make the program fail. In short system testing and quality assurance is a review in software products and related documentation for completion, correctness, reliability and maintainability.

System testing is the first stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. testing is vital to the success of the system. System testing make a logical assumption that if all the parts of the system are correct and the goal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptance testing.

The testing steps are:

- Unit testing
- Integration testing
- Validation
- Output testing
- Acceptance testing

System testing provides the file assurance that software once validated must be combined with all other system elements. System testing verifies whether all elements have been combined properly and that overall system function and performance is achieved. After the integration of modules, the validation test was carried out over the system. It was found that all the modules work well together and meet the overall system function and performance.

a. Unit Testing

Unit testing is carried out screen-wise, each screen being identified as an object. Attention is diverted to individual modules, independently to one another to located errors. This has enabled the detection of errors in coding and logic. Various test cases are prepared. For each module these test cases are implemented and it is checked whether the module is executed as per the requirements and outputs the desired result. In this test each service input and output parameters are checked. In unit testing

- Module interface was tested to ensure that information properly flows into and out of the program under test.
- Boundary condition was tested to ensure that module operates properly at boundaries established to limit or restrict processing.
- All independent parts through the control structures were executed to ensure that all statements in the modules have been executed at least once.
- Error handling paths were also tested.

b. Integration Testing

Integration testing is a systematic technique for constructing the program structure while the same time conducting test to uncover errors associated with interfacing. Unit tested module were taken and a single program structure was built that has been dictated by the design. Incremental integration has been adopted here. The modules are tested separately for

accuracy and modules are integrated together using bottom-up integration i.e., by integrating from moving from bottom to the top. The system is checked and errors found during Integration and rectified. In this testing, all the individual modules were combined and the module wise shifting was verified to be alright. The entire software was developed and tested in small segments, where errors were easy to locate and rectify. Program builds (group of modules) were constructed corresponding to the successful testing of user interaction, data manipulation analysis and display processing and database management.

c. Validation Testing

Validation testing is done to ensure complete assembly of the error-free software. Validation can be termed successful only if it functions in manner. Reasonably expected by the customer under validation is alpha and beta testing. The customer-side validation is done in the testing phase. It is checked whether the data passed to each service is valid or not. Entering incorrect values does the validation testing and it is checked whether the errors are being considered. Incorrect values are to be discarded. The errors are rectified.

d. Output Testing

After performing the validation testing the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format. The output format on the screen was found to be correct as the format was designed in the system design phase according to the user needs. For the hard copy also, the output comes out as specified requirement by the user. Hence output testing does not result in any correction the system. The project is developed based on the user choice. It is user friendly. The output format is very clear to user. Output testing is done on smart builders correctly.

e. Acceptance Testing

Acceptance testing generally involves running a suit of tests on the completed condition of the user's environment or future of the system, and will result in a pass or fail, or Boolean outcome

5. SYSTEM IMPLEMENTATION

5.1 SYSTEM DESCRIPTION

The application developed is designed in such a way that any further enhancement can be done with ease. The system has the capability for easy integration. New modules can be added to the system with less effort. The website is developed in python which make it more reliable and compatible with other environments. The application proves better extensibility and flexibility for future enhancements.

5.2 SYSTEM IMPLEMENTATION

The implantation is the final state and it is an important phase. It involves the individual programming, system testing, user training and the operational running of developed proposed system that constitutes the application subsystems. A major task of preparing for implementation is education of users, which should really have been taken place much earlier in the project when they were being involved in the investigation and design work. During the implementation phase system actually takes physical shape. In order develop a system implemented planning is very essential the implementation phase of the software development is concerned with translating design specification into source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented. Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period of time. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from old system to new system. The system can be implemented only after testing is done and is found to be working to specifications. The implementation stage is a systems project in its own right. The implementation stage involves following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of method to achieve change over
- Evaluation of the changeover method.

In the case of this project all the screens are designed first. For making it to be executable, codes are written on each screen and performs the implementation by creating the database and connecting to the server. After that the system, is Checked, whether it performs all the transactions correctly. Then databases are cleared and made it to be usable to the technicians.

6. MAINTENANCE

Software maintenance is the modification of a software product after delivery to correct faults, to improve performance or other attributes. Maintenance is the ease with which a program can be corrected if any error is encountered, adapted if its environment changes or enhanced if the customer desires a change in requirement. Maintenance follows conversation to extend those changes are necessary to maintain satisfactory operations relative to changes in the user's environment. Maintenance often includes minor enhancements or corrections to problems that surface in the system's operation. Maintenance is also done based on fixing the problems reported, changing the interface with other software or hardware enhancing the software.

CATEGORIES OF MAINTENANCE

a) Corrective Maintenance

Corrective maintenance is the most commonly used maintenance approach, but it is easy to see its limitations. When equipment fails, it often leads to downtime in production, and sometimes damages other parts. In most cases, this is expensive. Also, if the equipment needs to be replaced, the cost of replacing it alone can be substantial. Reliability of systems maintained by this type of maintenance is unknown and cannot be measured. Corrective maintenance is possible since the consequences of failure or wearing out are not significant and the cost of this maintenance is not great

b) Perfective Maintenance

Modification of a software product alters delivery to improve performance or maintainability. This term is used to describe changes undertaken to expand the existing requirements of the system. A successful piece of software tends to be subjected to the

Succession of changes resulting in an increase in user requirements. This is based on the premise that as the software becomes useful, the user experiments with new cases beyond the scope for which it was initially developed. Expansion requirements can take the form of enhancement of existing system functionality and improvement in computational efficiency.

c) Adaptive Maintenance

Modification of a software product performed after delivery to keep a product usable in a changed or changing environment. Adaptive maintenance includes any work initiated as a consequence of moving the software to a different hardware or software platform. It is a change driven by the need to accommodate modifications in the environment of software system. The environment in this context refers to the totality of all conditions and influences which act from outside upon the system. A change to the whole or part of this environment will warrant a corresponding modification of the software.

d) Preventive Maintenance

Preventive maintenance is a schedule of planned maintenance actions aimed at the prevention of breakdowns and failures. The primary goal of preventive maintenance is to prevent the failure of equipment before it actually occurs. It is designed to preserve and enhance equipment reliability by replacing worn components before they actually fail. Preventive maintenance activities include equipment checks, partial or complete overhauls at specified periods.

Long-term benefits of preventive maintenance include:

- ❖ Improved system reliability.
- ❖ Decreased cost of replacement.
- ❖ Decreased system downtime.

7. CONCLUSION

It was a wonderful learning experience for me while working on this project. This project took me through the various phases of project development and gave me real insight into the world of software engineering. The motive behind the project is to provide a generic platform for providing two-way communication between service provider and service receiver can eliminate the exhausting efforts of searching the suitable service provider and also ensures the intended job for the service provider. A feedback-based rating system can improve the skills of any service provider and service receiver can eventually get better service.

The joy of working and the thrill involved while tackling the various problems and challenges gave me a feel of the developers' industry. .It was due to this project I came to know how professional software is designed.

8.BIBILOGRAPHY

- PYTHON PHRASEBOOK- *Brad Dayley (Sams publishing)*
- SYSTEM ANALYSIS AND DESIGN –*Elias m Award (Second Edition 1999)*
- DATABASE SYSTEM CONCEPTS- *Silberschatz (Fourth Edition, February 2004)*
- COMPUTER PROGRAMMING JAVASCRIPT, PYTHON, HTML, SQL, CSS-

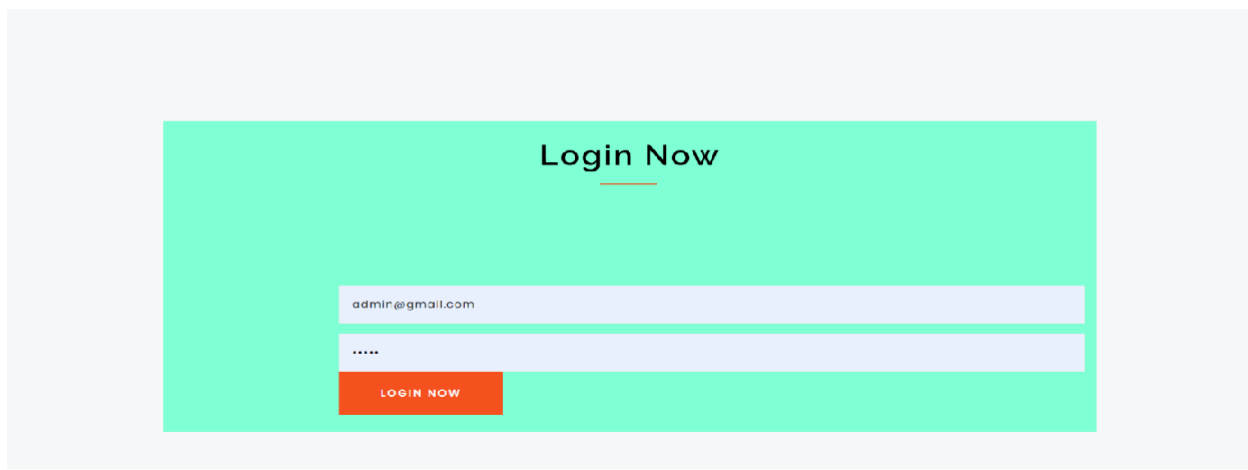
William Alwin Newton

- <https://www.w3schools.com/>
- <https://www.nrega.nic.in/>

9. APPENDIX

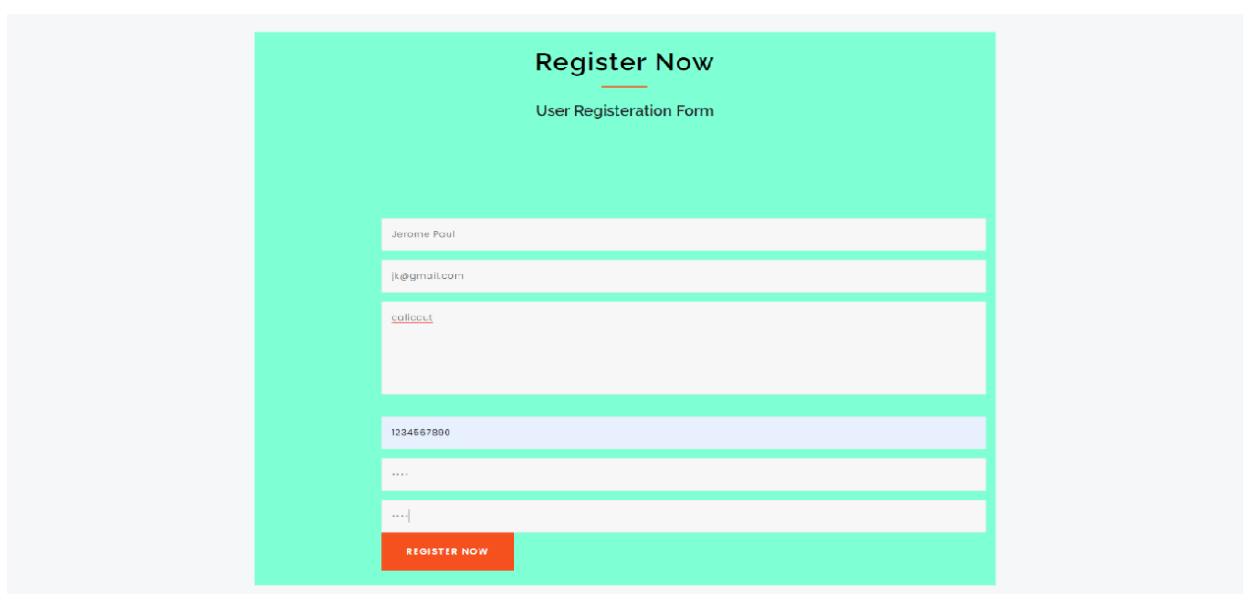
9.1 SCREEN SHOTS

Login Page



The screenshot shows a login form titled "Login Now" with a red underline. The form is set against a light blue background. It contains two input fields: the first is for an email address, with "admin@gmail.com" entered, and the second is for a password, with "*****" entered. Below these fields is a red button labeled "LOGIN NOW".

Customer Registration



The screenshot shows a registration form titled "Register Now" with a red underline, followed by the subtitle "User Registration Form". The form is set against a light blue background. It contains several input fields: a name field with "Jerome Paul", an email field with "jk@gmail.com", a password field with "c@lloct", a phone number field with "1234567890", and a captcha field with "....". Below these fields is a red button labeled "REGISTER NOW".

Service Centre Registration

Service Hub Registration Form

Asus Home Care

asus@gmail.com

Kozhikode

kzhikode

1234567890

asus

Laptop

1234567890





REGISTER NOW

Service Centre Approve




Registered Service Hubs							
Name	Email	Address	Phone No	Company	Product	aid	status
Lenovo	l@gmail.com	Adimaly	2147483647	Lenovo	Laptop	15245	Approve Now
Approved Service Hubs							
Name	Email	Address	Phone No	Company	Product	aid	status
Apple	apple@gmail.com	Edappaly	2147483647	Apple	Laptop	12345	Delete

Booking a Service Centre - Customer





+021 365 777

Service Hub
[Home](#)
[Need Help..](#)
[Track..](#)
[Add Feedback..](#)
[View FAQ](#)
[Message](#)
[View Message](#)
[Log Out](#)







#	Name	Email	Address	Phone No	Company	Book Now
1	Lenovo	l@gmail.com	Adimaly	2147483647	Lenovo	Book Now

© 2018 Home Loan. All Rights Reserved | Design by [W3layouts](#)


Follow us on:
 




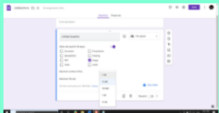
Track status of complaint

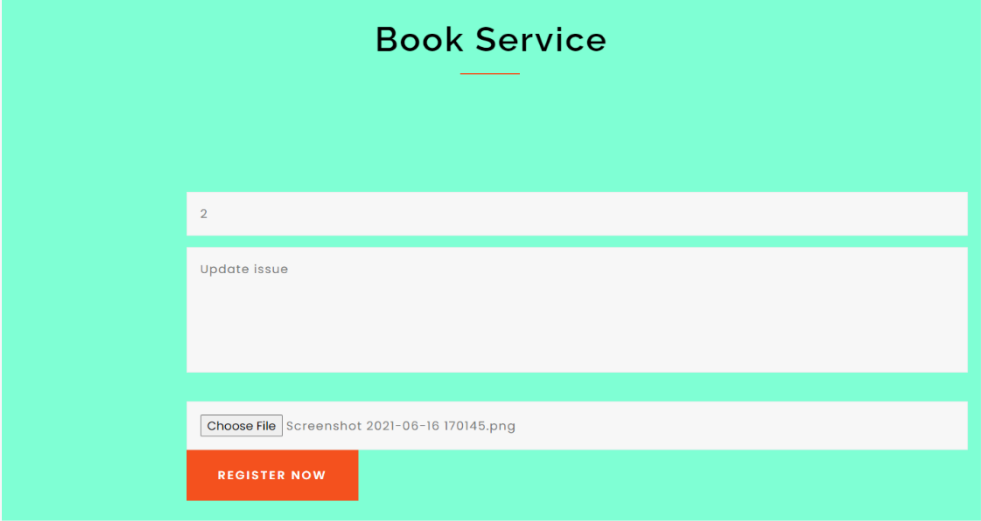
+021 365 777

Service Hub
[Home](#)
[Need Help..](#)
[Track..](#)
[Add Feedback..](#)
[View FAQ](#)
[Message](#)
[View Message](#)
[Log Out](#)



Company	problem	Image	status
Lenovo	Update issue		complaint registered

Book Service



The screenshot displays a web form titled "Book Service" on a light blue background. The form is white and contains the following elements:

- A text input field with the number "2" entered.
- A text input field with the placeholder text "Update Issue".
- A file upload section with a "Choose File" button and the text "Screenshot 2021-06-16 170145.png".
- An orange button labeled "REGISTER NOW".

10.CODE

Login

```
{% extends 'common/commonbase.html' %}
{% block content %}
<section class="contact py-5">
<div class="container py-md-4 mt-md-3" style="background-color: aquamarine;">
<h2 class="heading-agileinfo">Login Now<span></span></h2>
<span class="w3-line black"></span>
<div class="inner-sec-w3layouts-agileinfo mt-md-5 pt-5">

<div class="contact_grid_right mt-5">

<form method="post">
<div class="contact_left_grid" style="padding-left: 200px;">

{% csrf_token %}
<input type="email" name="email" placeholder="Email" required="">
<input type="password" name="password" placeholder="password" required="">
<input type="submit" value="login Now">

<div class="clearfix"> </div>
</div>
</form>
</div>
</div>
</div>
</div>
</section>
{% if msg %}
<script>alert('{{ msg }}');</script>
{% endif %}
{% endblock %}
```

Home

```
{% load static %}

<!--
Author: W3layouts
Author URL: http://w3layouts.com
License: Creative Commons Attribution 3.0 Unported
License URL: http://creativecommons.org/licenses/by/3.0/
-->
<!DOCTYPE HTML>
<html lang="zxx">

<head>
<title>Service Hub</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta charset="utf-8">
<meta name="keywords" content="Home Loan Responsive web template, Bootstrap Web Templates,
Flat Web Templates, Android Compatible web template,
SmartPhone Compatible web template, free WebDesigns for Nokia, Samsung, LG, SonyEricsson,
Motorola web design" />
<script>
addEventListener("load", function () {
setTimeout(hideURLbar, 0);
}, false);

function hideURLbar() {
window.scrollTo(0, 1);
}
</script>

<!-- Bootstrap Core CSS -->
<link href={% static 'css/bootstrap.css' %} rel='stylesheet' type='text/css' />
<link href={% static 'css/style.css' %} rel='stylesheet' type='text/css' />
<!-- pop up box -->
<link href={% static 'css/popuo-box.css' %} rel="stylesheet" type="text/css" media="all" />
<!-- font-awesome icons -->
<link href={% static 'css/fontawesome-all.min.css' %} rel="stylesheet">
<!-- //Custom Theme files -->
<!-- online fonts -->
<!-- titles -->
<link href="//fonts.googleapis.com/css?family=Raleway:300,400,500,600,700,800,900"
rel="stylesheet">
<!-- body -->
```

```

<link href="//fonts.googleapis.com/css?family=Poppins:400,500,600,700,800" rel="stylesheet">
</head>

<body>

<!-- header -->
<header>
<div class="top">
<div class="container">
<div class="t-op row">
<div class="col-sm-6 top-middle">
<ul>
<li><a href="#"><i class="fab fa-facebook-f"></i></a></li>
<li><a href="#"><i class="fab fa-twitter"></i></a></li>
<li><a href="#"><i class="fab fa-google-plus-g"></i></a></li>
<li><a href="#"><i class="fab fa-linkedin-in"></i></a></li>
</ul>
</div>
<div class="col-sm-6 top-left">
<ul>
<li><i class="fas fa-phone"></i> +021 365 777</li>
</ul>
</div>
</div>
</div>
</div>
</div>
<div class="container">
<nav class="navbar navbar-expand-lg navbar-light">
<h1>
<a class="navbar-brand text-capitalize" href="/index/">
Service hub
</a>
</h1>
<button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
<span class="navbar-toggler-icon"></span>
</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">
<ul class="navbar-nav text-center ml-lg-auto">
<li class="nav-item mr-3">
<a class="nav-link" href="/index/">Home
<span class="sr-only">(current)</span>
</a>
</li>

```

```

<li class="nav-item mr-3">
<a class="nav-link" href="/userreg/">User</a>
</li>
<li class="nav-item mr-3">
<a class="nav-link" href="/screg/">Service</a>
</li>

<li class="nav-item ">
<a class="nav-link" href="/login/">Login</a>
</li>
</ul>
</div>
</nav>
</div>
</header>
<!-- //header -->
<!-- banner -->
<section class="banner">
<div class="callbacks_container">
<ul class="rslides" id="slider3">
<li>
<div class="slider-info bg1">
<div class="banner-text container">
<h4 class="movetxt text-left mb-3 agile-title text-capitalize">service hub </h4>
<p class="text-white mb-3">online solution for your tech related problems </p>

<i class="fas fa-angle-double-right"></i>
</a>
</div>
</div>
</li>
<li>
<div class="slider-info bg2">
<div class="banner-text container">
<h4 class="movetxt text-left mb-3 agile-title text-capitalize">service hub </h4>
<p class="text-white mb-3">online solution for your tech related problems </p>

<i class="fas fa-angle-double-right"></i>
</a>
</div>
</div>
</li>
<li>
<div class="slider-info bg3">
<div class="banner-text container">
<h4 class="movetxt text-left mb-3 agile-title text-capitalize">service hub </h4>

```

```

<p class="text-white mb-3">online solution for your tech related problems </p>
<i class="fas fa-angle-double-right"></i>
</a>
</div>
</div>
</li>
</ul>
</div>
</section>
<!-- //banner -->
<!-- Products -->

<!--footer-->
<footer>
<div class="container py-md-4 mt-md-3">
<div class="row footer-top-w3layouts-agile py-5">
<div class="col-lg-3 col-md-6 col-sm-6 footer-grid">
<div class="footer-title">
<!-- <h3>About Us</h3> -->
</div>
<div class="footer-text">
<!-- <p>Curabitur non nulla sit amet nisl tempus convallis quis ac lectus. lacinia eget consectetur sed, convallis at tellus..</p> -->

</div>
</div>
<div class="col-lg-3 col-md-6 col-sm-6 footer-grid">
<!-- <div class="footer-title">
<h3>Contact Us</h3>
</div> -->
<!-- <div class="footer-office-hour">
<ul>
<li class="hd">Address :</li>
<li>No.27 - 5549436 street lorem, Newyork City, USA</li>

</ul>
<ul>
<li class="hd">Phone:+ 1 (234) 567 8901</li>
<li class="hd">Email:
<a href="mailto:info@example.com">info@example.com</a>
</li>
<li class="hd">Fax: 1(234) 567 8901</li>
</ul>
</div> -->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 footer-grid">

```

```

<div class="footer-title">
<!-- <!-- <h3>Recent Posts</h3> -->
</div>
<!-- <div class="footer-list">
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>

```

```

<div class="flickr-grid">
<a href="#" data-toggle="modal" data-target="#myModal">

</a>
</div>
<div class="clearfix"></div>
</div> -->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 footer-grid">
<!-- <div class="footer-title">
<h3>Subscribe</h3>
</div> -->
<!-- <p>Vivamus magna justo, lacinia eget consectetur sed.</p> -->
<!-- <form action="#" method="post" class="newsletter">
<input class="email" type="email" placeholder="Your email..." required="">
<button class="btn1">
<i class="far fa-envelope"></i>
</button>
</form> -->
<div class="clearfix"></div>

</div>
</div>
</div>
</footer>
<!------>
<div class="copyright py-3">
<div class="container">
<div class="copyrighttop">
<ul>
<li>
<h4>Follow us on:</h4>
</li>
<li>
<a class="facebook" href="#">
<i class="fab fa-facebook-f"></i>
</a>
</li>
<li>
<a class="facebook" href="#">
<i class="fab fa-twitter"></i>
</a>
</li>
<li>
<a class="facebook" href="#">
<i class="fab fa-google-plus-g"></i>

```

```
</a>
</li>
<li>
<a class="facebook" href="#">
<i class="fab fa-pinterest-p"></i>
</a>
</li>
</ul>
</div>
<div class="copyrightbottom">
<p>© 2018 Home Loan. All Rights Reserved | Design by
<a href="http://w3layouts.com/">W3layouts</a>
</p>
</div>
<div class="clearfix"></div>
</div>
<!-- Modal -->
<div class="modal fade" id="myModal" tabindex="-1" role="dialog" aria-
labelledby="exampleModalCenterTitle" aria-hidden="true">
<div class="modal-dialog modal-dialog-centered" role="document">
<div class="modal-content">

<div class="modal-header">
<h4 class="modal-title">Home Loan</h4>
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
<span aria-hidden="true">&times;</span>
</button>
</div>
<div class="modal-body">
<div class="agileits-w3layouts-info">

<p>Duis venenatis, turpis eu bibendum porttitor, sapien quam ultricies tellus, ac rhoncus risus odio
eget nunc. Pellentesque ac fermentum diam. Integer eu facilisis nunc, a iaculis felis. Pellentesque
pellentesque tempor enim, in dapibus turpis porttitor quis. </p>
</div>
</div>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
<button type="button" class="btn btn-primary">Save changes</button>
</div>
</div>
</div>
</div>
<!-- //Modal -->
<!-- js-->
```

```
<script src="{ % static 'js/jquery-2.2.3.min.js' % }"></script>
<!-- js-->
<!-- Banner Responsiveslides -->
<script src="{ % static 'js/responsiveslides.min.js' % }"></script>
<script>
// You can also use "$(window).load(function() {
$(function () {
// Slideshow 4
$("#slider3").responsiveSlides({
  auto: true,
  pager: true,
  nav: false,
  speed: 500,
  namespace: "callbacks",
  before: function () {
    $(' .events').append("<li>before event fired.</li>");
  },
  after: function () {
    $(' .events').append("<li>after event fired.</li>");
  }
});

});
</script>

<!-- // Banner Responsiveslides -->
<!-- stats -->
<script src="{ % static 'js/jquery.waypoints.min.js' % }"></script>
<script src="{ % static 'js/jquery.countup.js' % }"></script>
<script>
$($('.counter').countUp();
</script>
<!-- //stats -->
<!--pop-up-box -->
<script src="{ % static 'js/jquery.magnific-popup.js' % }"></script>
<script>
$(document).ready(function () {
$($('.popup-with-zoom-anim').magnificPopup({
  type: 'inline',
  fixedContentPos: false,
  fixedBgPos: true,
  overflowY: 'auto',
  closeBtnInside: true,
  preloader: false,
  midClick: true,
  removalDelay: 300,
```

```

mainClass: 'my-mfp-zoom-in'
});
});
</script>
<!-- //pop-up-box -->
<!-- Bootstrap Core JavaScript -->
<script src="{ % static 'js/bootstrap.js' }">
</script>
<!-- //Bootstrap Core JavaScript -->
</body>
</html>

```

User Registration

```

% extends 'common/commonbase.html' % }
{ % block content % }
<section class="contact py-5" >
<div class="container py-md-4 mt-md-3" style="background-color: aquamarine;">
<h2 class="heading-agileinfo">Register Now<span></span></h2><br>
<h4 align="center">User Registration Form</h4>
<span class="w3-line black"></span>
<div class="inner-sec-w3layouts-agileinfo mt-md-5 pt-5">

<div class="contact_grid_right mt-5">

<form method="post">
{ % csrf_token % }
<div class="contact_left_grid" style="padding-left: 180px;">
<input type="text" name="name" placeholder="Name" required="">
<input type="email" name="email" placeholder="Email" required="">
<textarea name="address" required="" placeholder="address"></textarea>
<input type="text" name="phoneno" placeholder="Phone No" required="">
<input type="password" name="password" placeholder="password" required="">
<input type="password" name="cpassword" placeholder="Confirm Password" required="">

<input type="submit" value="Register Now">

<div class="clearfix"> </div>
</div>
</form>

```



```
{% if msg %}
<script>
alert('{{ msg }}');
</script>
{% endif %}

{% if word %}
<script>
alert('{{ word }}');
</script>
{% endif %}
</div>
</div>
</div>
</section>
{% endblock %}
```

Service Centre Registration

```
{% extends 'common/commonbase.html' %}
{% block content %}
<section class="contact py-5">
<div class="container py-md-4 mt-md-3" style="background-color: aquamarine;">
<h2 class="heading-agileinfo">Register Now</span></h2><br>
<h4 align="center">Service Hub Registration Form</h4>
<span class="w3-line black"></span>
<div class="inner-sec-w3layouts-agileinfo mt-md-5 pt-5">

<div class="contact_grid_right mt-5">

<form action="#" method="post">
<div class="contact_left_grid" style="padding-left: 200px;">

{% csrf_token %}
<input type="text" name="name" placeholder="Name" required="" title="Please enter your Full
Name" required pattern="[A-Z a-z]+">
<input type="email" name="email" placeholder="Email" required="" title="Please enter your Email
Address">
<textarea name="address" required="" placeholder="address"></textarea>
<select name="district">
<option name="">select your district</option>
```

```
<option name="kasargod">kasargod</option>
<option name="kannur">kannur</option>
<option name="kozhikkode">kozhikkode</option>
<option name="wayanad">wayanad</option>
<option name="palakkad">palakkad</option>
<option name="thrissur">thrissur</option>
<option name="ernakulam">ernakulam</option>
<option name="alappuzha">alappuzha</option>
<option name="kottayam">kottayam</option>
<option name="idukki">idukki</option>
<option name="kollam">kollam</option>
<option name="pathanamthitta">pathanamthitta</option>
<option name="trivandram">trivandram</option>

</select>
<input type="text" name="phoneno" placeholder="Phone No" required="" title="Please enter your
Phone Number" required pattern="[6-9]{1}[0-9]{9}">

<select name="company">
<option name="">
select your company

</option>
{% for i in data %}
<option name="{{i.1}}">
{{i.1}}

</option>
{% endfor %}
</select>

<input type="text" name="product" placeholder="Product type" required="" title="Please enter your
Full Name" required pattern="[A-Z a-z]+">
<input type="text" name="aid" placeholder="Authorization id" required="" title="Please enter your
Authorization id" required maxlength="5">

<input type="password" name="password" placeholder="password" required="">
<input type="password" name="cpassword" placeholder="Confirm Password" required="">
<input type="submit" value="Register Now">

<div class="clearfix"> </div>
</div>
```

```
</form>
</div>
</div>
</div>
</section>
<!--//contact-->
{ % endblock % }
```