

Project Report

CIS 634 / EEC 521 Software Engineering

Project Name : Chatrasahaya

Team Details :

Vikas Sabbi (2836919)

Keshav Singhal (2822290)

Tanmay Rajgor (2827074)

Table of Contents

| SNO | TOPIC | Page No |
|-----|---|---------|
| 1 | Introduction: Purpose and Scope | 5 |
| 2 | Introduction: Structure of the Document | 5 |
| 3 | Project Management Plan | 6 |
| 4 | Risk Analysis | 8 |
| 5 | Hardware and Software Resource Requirements | 9 |
| 6 | Requirement Specifications | 10 |
| 7 | Architecture | 11 |
| 8 | Technology, software, and hardware used | 12 |
| 9 | Design | 13 |
| 10 | Components design | 14 |
| 11 | Database design | 16 |
| 12 | Rationale for your detailed design mode | 17 |
| 13 | A complete list of system test cases | 18 |
| 14 | Traceability of test cases to use cases | 20 |
| 15 | Techniques used for test case generation | 21 |
| 16 | Test results and assessments (how good are your test cases? How good is your software?) | 22 |
| 17 | Defects reports | 23 |
| 18 | Conclusions | 23 |

List of Figures

| FIGURE_NO | NAME | PAGE No |
|-----------|--|---------|
| Fig 1 | Structure of the Document | 5 |
| Fig 2 | Project Organization | 6 |
| Fig 3 | Lifecycle Model | 7 |
| Fig 4 | Graphic use case | 10 |
| Fig 5 | Architectural model | 11 |
| Fig 6 | Request Us Page | 13 |
| Fig 7 | Map and Count View | 13 |
| Fig 8 | Request Us Form Static Model | 14 |
| Fig 9 | Mail Notification Static Model | 14 |
| Fig 10 | Map View Static Model | 14 |
| Fig 11 | Count View Static Model | 15 |
| Fig 12 | Activity diagram for Request Us & Map view , Count View | 15 |
| Fig 13 | Sequence diagram for Request Us & Process Request(Mail Notification),Map view , Count View | 15 |
| Fig 14 | Data structures Model | 16 |

List of Tables

| Table_NO | NAME | PAGE No |
|-------------|---------------------------------|---------|
| Table_NO 1 | Abbreviations | 6 |
| Table_NO 2 | Risk Analysis | 8 |
| Table_NO 3 | Schedule | 9 |
| Table_NO 4 | Database Design | 16 |
| Table_NO 5 | Traceability Requirement Matrix | 17 |
| Table_NO 6 | Test Case Request Us 1 | 18 |
| Table_NO 7 | Test Case Request Us 2 | 18 |
| Table_NO 8 | Test Case Request Us 3 | 19 |
| Table_NO 9 | Test Case Mail Notification | 19 |
| Table_NO 10 | Test Case Maps View 1 | 19 |
| Table_NO 11 | Test Case Maps View 2 | 19 |
| Table_NO 12 | Test Case Count View 1 | 19 |
| Table_NO 13 | Test Case Count View 2 | 19 |
| Table_NO 14 | Traceability of test cases | 20 |
| Table_NO 15 | Test Results | 22 |
| Table_NO 16 | Assessments | 22 |
| Table_NO 17 | Defect Report | 23 |

1. Introduction

1.1. Purpose and Scope

Purpose :

This software will help the Chatrasahaya to be more efficient in submitting of their requests such as money, food and other needs...and tracking of submission on the maps. It also sends the mails to the internal team to check the submissions quickly. The purpose of this project is to computerize all details regarding submissions from request us

Scope :

Users can submit their required needs like education, financial, Health...from the **Request Us for Self /For Others** Page by using some fields such as **Name** -- Full Name of the requester, **Phone Number** -- Phone/mobile Number of the Requester, **Location** -- The place of the request where books, chairs, tables are needed for the schools, **Category**-- Its was type of the request such as education, financial, Health **Description** -- Its was the description/overview of the needs, **Aadhar Number** -Its was unique identity number of the requester, **Documents** --- Proof of the documents needs to be uploaded here.

1.2. Product Overview (including capabilities, scenarios for using the product, etc.)

It is meant to delineate the features of Request Us & Maps Menus for the existing application, so as to serve as a guide to the developers on one hand and as a software validation document for the prospective client on the other.

Request Us & Maps are two enhancements for the existing application called **chatrasahaya** for a web application. Our application is designed for User/Customer for requesting of required categories like education, financial, Health to gain best education for less budget schools or free schools for the existing application. Maps can show the count for the each state based on requests completed by the users/team.

1.3. Structure of the Document

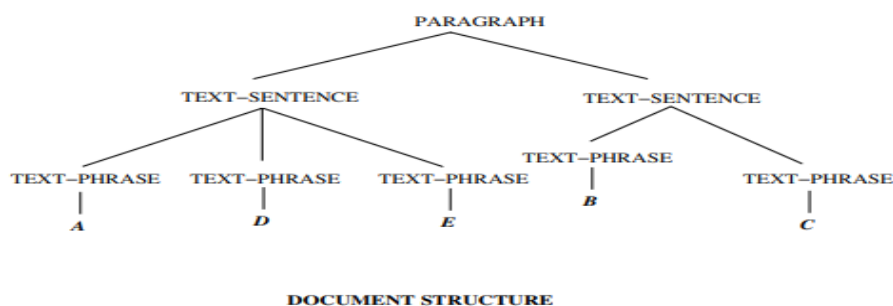


Fig 1: Structure of the Document

1.4. Terms, Acronyms, and Abbreviations

Terms

1. Request Us - Its was a page to request for needs form the chatrasahaya team
2. Maps View-Its was showing of the counts in the state wise maps
3. Request Us(For Self) - Requesting for him/her needs by giving her/him details
4. Request Us(For Others)-Requesting for others people by giving other person details

Abbreviations

| | |
|-----|-------------------------------------|
| SRS | Software Requirement Specification. |
| SDS | Software Design Specification |
| DFD | Data Flow Diagram. |

Table No 1 : Abbreviations

Acronyms

- ✓Request Us Button in the Page– Submitting of the request us page with the details required.
- ✓ Number in the map - Display the no of submissions/requests for the state on the map
- ✓ Aadhar Id- Its was the unique id which was issued by the govt of the india of the person.

2. Project Management Plan

2.1. Project Organization

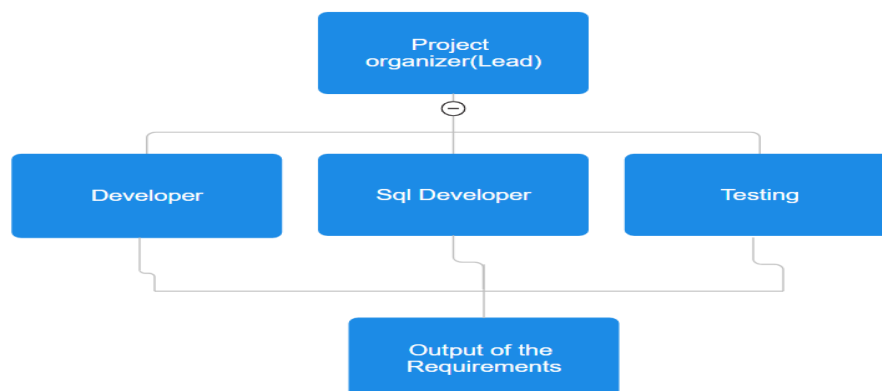


Fig 2: Project Organization

2.2. Lifecycle Model Used

Software Development Life Cycle (also called SDLC Models) is a workflow process which defines the core stages and activities of development cycles or A framework that describes the operations performed at each phase of a software development project.

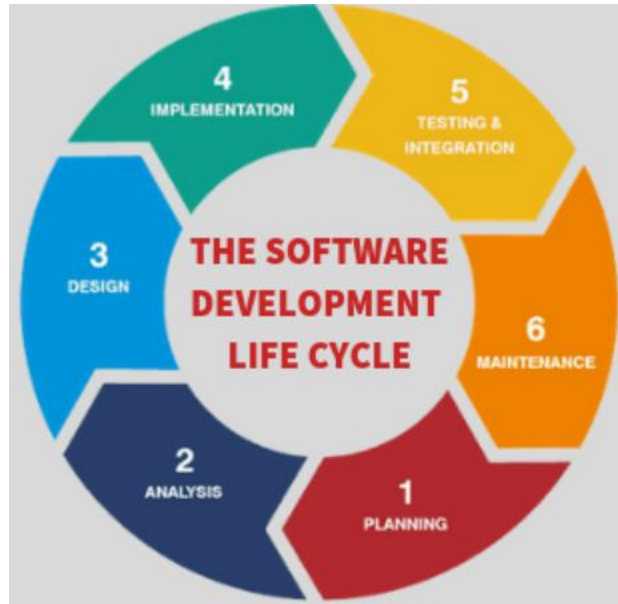


Fig 3: Lifecycle Model

1. **Planning** : Initially , team has planned to do enhancements(Request Us,Map & Count View) for the chatrasahaya website while discussed with the vikas on his own website

2. **Analysis** : Started our analysis by using what are the required fields for the request us page and sending of mails , showing of submissions of request us on the map & count view

3. **Design** : Initially we started designing of the documents and code logic things for the request us page and sending of mails , showing of submissions of request us on the map & count view

4. **Implementation**: Started the code of the request us page and then integrated the google maps api by taking the google platform account officially and sending of the mails after successfully submissions and showing of counts in the map

5. **Testing & Integration** : Testing of the request us page by checking the validations and alerts without proper data and doing of the integration testing once for all the requirements of the website

6. **Maintenance** : Doing the maintenance for the website if any further enhancements required like mobile app,developing of the pages

2.3. Risk Analysis

This section discusses project risks and the approach to managing them.

Project risks are events or circumstances that can potentially have a negative impact on a project's objectives, such as its scope, schedule, budget, or quality. Identifying and assessing project risks is a crucial part of project management.

- ❖ Scope Creep
- ❖ Resource Constraints
- ❖ Schedule Delays
- ❖ Quality Issues
- ❖ Technical Challenges
- ❖ Communication Breakdown

| Name of risk | probability | impact | RM3 pointer |
|-------------------------|-------------|-------------------------------------|-------------|
| Scope Creep | less | Impact on Design of the requirement | Initial |
| Resource Constraints | less | Impact on Schedule Timings | Optimized |
| Schedule Delays | medium | Impact on Timely Deliverable | Defined |
| Quality Issues | medium | Impact on Output of the requirement | Optimized |
| Technical Challenges | medium | Impact on Proper Coding & testing | Initial |
| Communication Breakdown | medium | Impact on Design of the requirement | Initial |

Table No 2 :Risk Analysis

2.4. Hardware and Software Resource Requirements

Hardware Requirements

Laptop/Desktop PC-Purpose for users submitting of the request us page by filling the required fields in the application

Software Resource Requirements

External machine interfaces

- 1.Operating System: Windows 11
- 2.Microsoft Visual Studio
- 3.WordPress PHP on Go-daddy itself
- 4.Microsoft SQL Server Management Studio
- 5.Java/python Development Toolkit.
- 6.Gravity Forms to store the data
- 7.Gradle 3.0 to build from command line

External system interfaces

Users needs to have stable internet connection to access the application and requester details to submit the request

2.5. Deliverables and schedule

Project Plan – 09/21/2023

Software Requirement Specification (SRS) – 10/05/2023

Software Design Specification (SDS) – 10/19/2023

Initial version of software – 11/02/2023

Test plan – 11/09/2023

Project Deliverables – 12/07/2023

Project presentation and demonstration(Final Project) --12/5/2023-12/7/2023

| Feature | Dev Start Date | Dev End Date | QA Start Date | QA End Date |
|-------------------|----------------|--------------|---------------|-------------|
| Request US | 11-01-2023 | 11-01-2023 | 11-02-2023 | 11-02-2023 |
| Mail Notification | 11-01-2023 | 11-01-2023 | 11-03-2023 | 11-03-2023 |
| Map View | 11-20-2023 | 11-22-2023 | 11-24-2023 | 11-24-2023 |
| Count View | 11-22-2023 | 11-24-2023 | 11-25-2023 | 11-25-2023 |

Table No 3 : Schedule

3. Requirement Specifications

3.1. Stakeholders for the system

In a project, there are both internal and external stakeholders. Internal stakeholders may include top management, project team members, your manager, peers, resource manager, and internal customers.

Stakeholders for the system :

- ❖ End Users
- ❖ Internal Team who give the requirements
- ❖ Project Management Team

3.2. Use cases

3.2.1. Graphic use case model

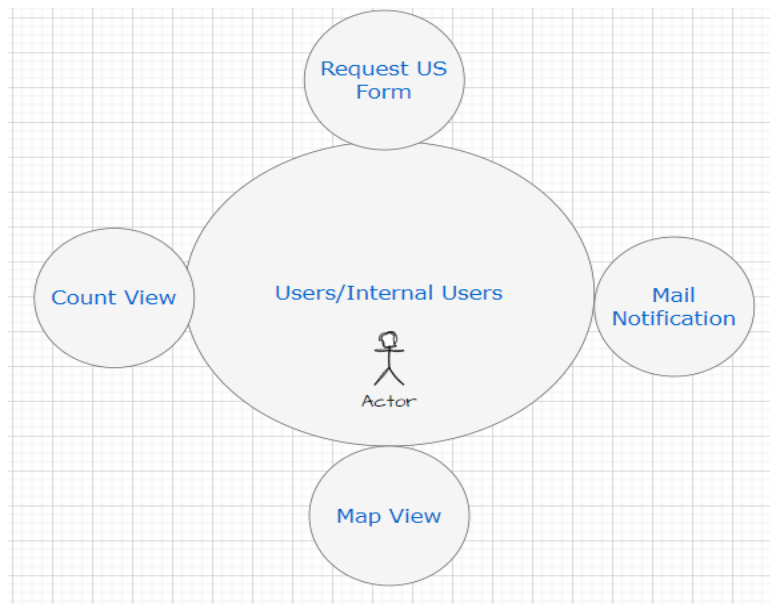


Fig 4 Graphic use case

3.2.2. Textual Description for each use case

1. **Request Us:** If customer wants to request us the required needs to the low budget schools , this page can be useful to submit the details of location,proofs.
2. **Mail Notification:** Our internal team can receive the mail after submitting the request us details
3. **Map View:** User can know the no of requests completed in the map by state level
4. **Count View:** User can know the details of the each count like request location,requester name, request items etc...

3.3. Rationale for your use case model

Graphical models act as an effective tool for requirements management, capturing and organizing functional aspects while validating the completeness of system functionalities. Their role extends beyond mere documentation; they serve as a design blueprint, aiding in architectural planning and offering a reference point for system validation and iteration. Moreover, these models encourage collaboration among team members, allowing for collective contributions, refining system understanding, and facilitating ongoing development. Ultimately, they act as a comprehensive documentation reference, supporting future phases of the project, system maintenance, updates, and knowledge transfer.

3.4. Non-functional requirements

Design of the application should be good and to store the requester details full secure

4. Architecture

4.1. Architectural style(s) used

Component-Based Architecture: It emphasizes the decomposition of systems into reusable, self-contained software components. These components can be combined to create larger applications.

4.2. Architectural model (includes components and their interactions)

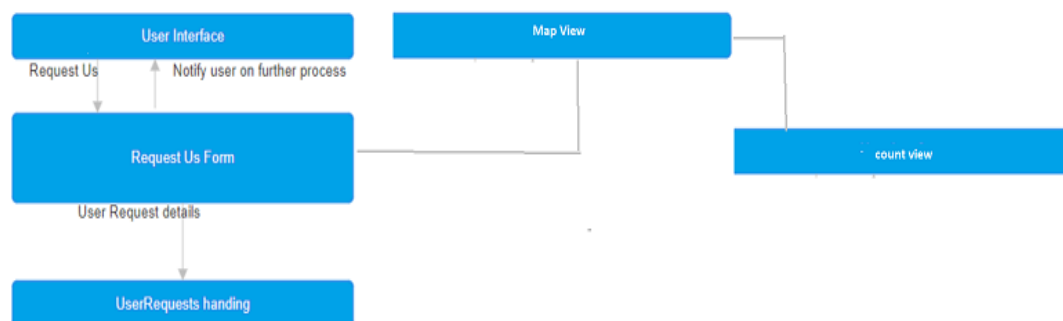


Fig 5 Architectural model

Description for Components

Major enhancement/Components in our web application are:

- Request Us form
- Mail Notification
- Map View
- Count View

Request Us form

This is the main function which allows user to request for the services provided by Chatrasahaya

Mail Notification: After End users clicking on Submit button with the valid data , A mail will be sent to the **chatrasahaya** team mail id to get to know the requester details.

Map View

This is the function where Users/Team can see the counts of the request services provided by Chatrasahaya team on the map view

Count View

This is the function where Users/Team can see the details such as name of the requester, location & category... of the counts on the map view

4.3. Technology, software, and hardware used

Technology, software, and hardware used

Technology: Python and Wordpress

Software Programming Languages: Python and Wordpress

Development Methodologies: Waterfall for project management and development processes.

Databases: Wordpress InBuild Form Tables Data

Networking: HTTP/HTTPS protocols for communication.

Hardware:Desktops/Laptops: Windows/Apple

4.4. Rationale for your architectural style and model

Real-Time Updates: Provides real-time information on incidents like accidents, road closures, traffic jams, and construction, enabling users to plan routes dynamically based on current conditions.

Enhanced Navigation: Offers alternative routes to avoid congested areas, reducing travel time and enhancing the overall navigation experience.

User Contribution: Allows users to report incidents, fostering a community-driven platform that contributes to the accuracy and timeliness of incident updates.

Information Accessibility: Offers accessible information across various devices, enabling users to stay informed about incidents while on the go.

Safety and Efficiency: Helps improve road safety by alerting users to potential hazards, allowing for safer and more efficient travel routes.

5. Design

5.1 Description of the user interface for Request Us

Below is the image for the Request US form where user can request the service Chatrasahaya

| | | | |
|---|-------------------------------|---------------------------------------|-----------------------------|
| Name <input type="text"/> | Phone <input type="text"/> | Aadhar Id <input type="text"/> | Category Food |
| Country <input type="text"/> | State <input type="text"/> | City <input type="text"/> | Zip <input type="text"/> |
| Address 1234 Main St | | | |
| Request Description Description | | | |
| Proof of documents <input type="button" value="Choose File"/> No file chosen | | | |
| <input type="button" value="Request Us"/> | | <input type="button" value="Cancel"/> | |

Fig 6 Request Us Page

5.2 Description of the user interface for Map View & Count View

Below is the image for the Map View & Count View where user can see the counts on the states where services has been completed raised by the Requester , When user clicks on counts such as 5,2 on below screenshot will display the Service details such as Name, Location,Proof of Documents and Description is Count View.



Fig 7 Map and Count View

5.2. Components design (static and dynamic models of each component)

Request Us Form

Form to accept Request : User can request for the services provided by Chatrasahaya and user can see the counts in the map on state view along with requester details in count view

Static models

Here is the below is the Class diagram for the User Request

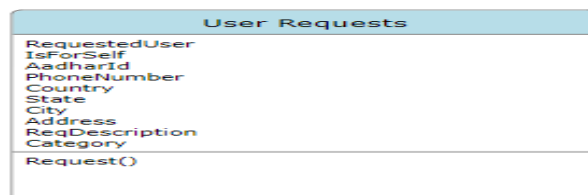


Fig 8 Request Us Form Static Model

Mail Notification

Sending the mail notification to the team after user submitting the request either by self for the others

Static models

Here is the below is the Class diagram for the Mail Notification

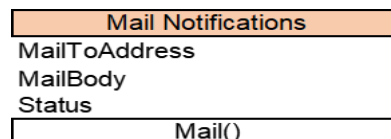


Fig 9 Mail Notification Static Model

Map View

Showing of the Requests successfully completed service as showing the counts in the Map View

Static models

Here is the below is the Class diagram for the Map View

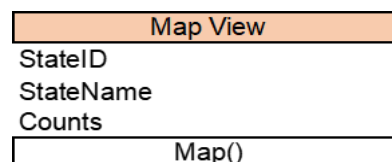


Fig 10 Map View Static Model

Count View

Showing of the Requester details when user clicks on counts on the states of the map view

Static models

Here is the below is the Class diagram for the Count View

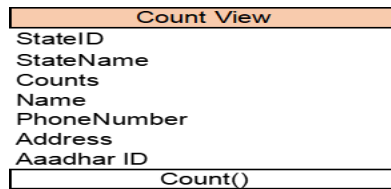


Fig11 Count View Static Model

Dynamic models

Activity diagram:

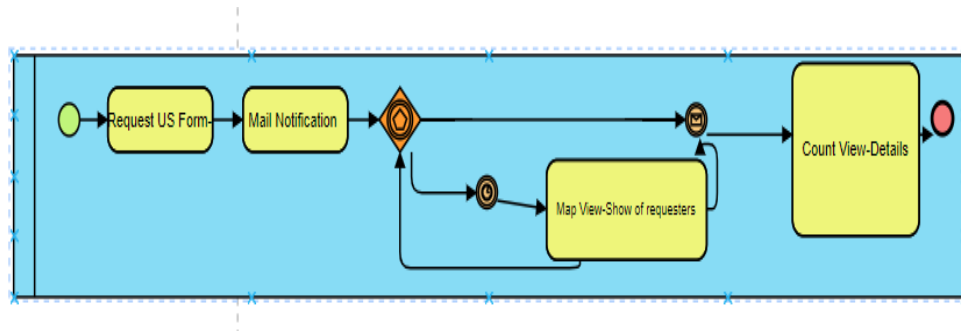


Fig 12 Activity diagram for Request Us & Map view , Count View

Sequence diagram:

The below show the sequence diagram for the Request Us form,Mail Notification,Map View,Count View

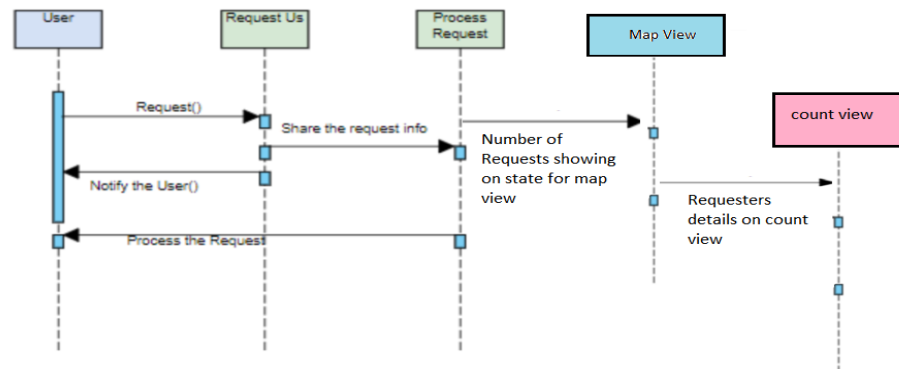


Fig 13 Sequence diagram for Request Us & Process Request(Mail Notification),Map view , Count View

5.3. Database design

Data design

A description of all data structures and databases.

Data structures

Data structured that are available to major portions of the architecture are shown below in table and diagram as well

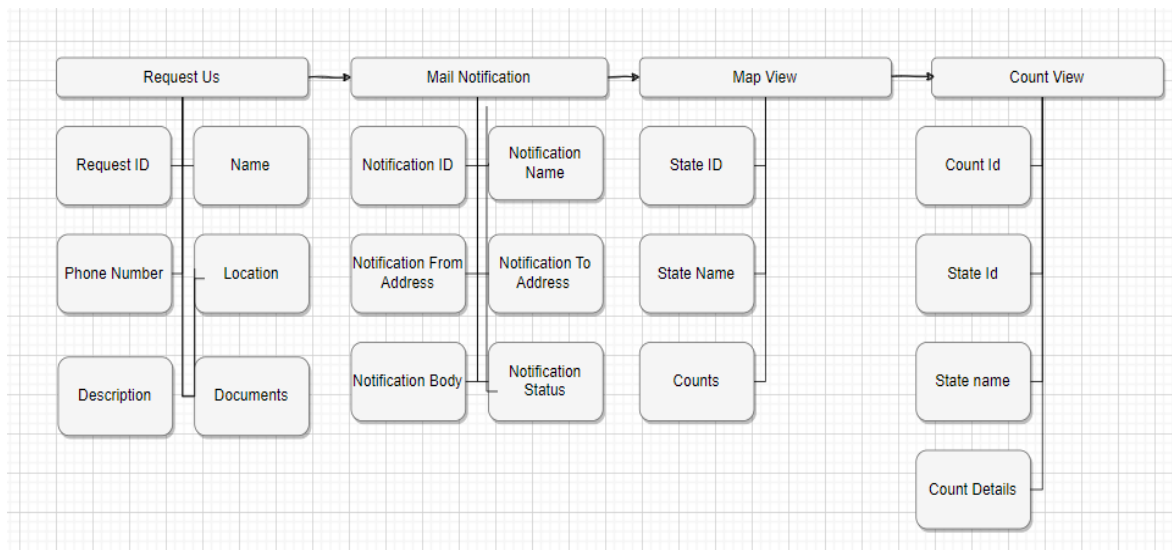


Fig 14 Data structures Model

Database description

Database(s) created as part of the application is(are) described.

Below is the table designed for application.

UserRequests, MailNotification, UserRequests can be used for the Map & Count View

| UserRequests | Mail Notification | Map View | Count View |
|----------------------|----------------------------------|--------------------|--|
| Customer ID: Int | Notification ID : Int | State ID: Int | Count ID: Int |
| Name: String | Notification Name: String | State Name: String | State ID: Int |
| Phone Number: BigInt | Notification To Address : String | Count: Int | State Name:String |
| Location: String | Notification From | | Count Details : Table with Request Name, Request Phone |

| | | | |
|----------------------------------|------------------------------------|--|---|
| | Address : String | | Number,Location,Description,Documents View |
| Description: String | Notification Body: String | | |
| Documents: Img,Pdf Uploads | Notification Status : String | | |

Table No 4: Database Design

5.4. Rationale for your detailed design models

Enhances user engagement and navigation by providing visually appealing and personalized map interfaces.

Offers concise, location-based information, aiding users in easily locating points of interest.

Enables interactive features that improve user interaction and understanding of geographical context within the site.

5.5. Traceability from requirements to detailed design models

| Use Case | Requirement Number | Requirement Description | Test Case | Design Component / Module | Comments |
|-------------------|--------------------|---|-----------|---|----------|
| Request US Form | CS_001 | End Users can request the services from this form either for self or for others by using proof documents,Name,Location and other details | TCS_001 | Request US Form - Name , Location,Phone Number,Aadhar Number , Proof of documents file upload | |
| Mail Notification | CS_002 | After End users clicking on Submit button with the valid data , A mail will be sent to the chatrasahaya team mail id to get to know the requester details. | TCS_002 | Mail Notification with the Requester Details | |

| | | | | | |
|------------|--------|--|---------|--|--|
| Map View | CS_003 | Users/Team can view the Counts on the Map for the State as how many services has been sponered or completed by the Chatrasahaya team | TCS_003 | Map View - Counts on the States | |
| Count View | CS_004 | Users/Team can view the details such as Name , Location,Phone Number,Aaddhar Number , Proof of documents files of the Counts on the Map for the State as how many services has been sponered or completed by the Chatrasahaya team | TCS_004 | Count View - Viewing of the Infromation such as Name , Location,Phone Number,Aaddhar Number , Proof of documents file upload | |

Table No 5: Traceability Requirement Matrix

6. Test Management

6.1. A complete list of system test cases

Test cases for Request Us:

| | |
|-----------------|--|
| ID | Request_1 |
| Test Input | No Data input, just User needs to clicks on Request Us link |
| Expected Output | Request Us Page should be displayed when user clicked on Request Us link |
| Description | User needs to click on request us link to display the request us page |

Table No 6: Test Case Request Us 1

| | |
|-----------------|--|
| ID | Request_2 |
| Test Input | Name,Phone,Aadhar ID,Email, Category, Country,State,City,Zip,Address,Request Description,Proof of Upload |
| Expected Output | User needs to submit the request us page with the valid data input |
| Description | User has to enter his/her or other details to submit the request us page in the application. If these details of the |

| | |
|--|---|
| | data is valid, he. she will be shown the confirmation message or Else, error message should be displayed. |
|--|---|

Table No 7: Test Case Request Us 2

| | |
|-----------------|--|
| ID | Request_3 |
| Test Input | Invalid format of the data fields |
| Expected Output | Error message should be display when user clicked on submit with invalid format of the data fields in the request us page |
| Description | User has to enter his/her or other details to submit the request us page in the application. If these details of the data is valid, he. she will be shown the confirmation message or Else, error message should be displayed. |

Table No 8: Test Case Request Us 3

Test cases for Mail Notification:

| | |
|-----------------|--|
| ID | Mail_1 |
| Test Input | User needs to click on submit the request us page with the valid data |
| Expected Output | User/team should get the email notification with the request details after successfully submission |
| Description | User should be Mail and can be seen in the sent list for the internal email box. |

Table No 9: Test Case Mail Notification

Test cases for Maps View:

| | |
|-----------------|---|
| ID | Map_1 |
| Test Input | No Data input, just User needs to clicks on Request Us Map View link |
| Expected Output | Map view page with states should be display when user clicked on Map view |
| Description | User needs to click on map view to see the map view |

Table No 10: Test Case Maps View 1

| | |
|-----------------|--|
| ID | Map_2 |
| Test Input | No Data input, just User needs to clicks on Counts on the Map View link |
| Expected Output | Count view with the request details should be display when user clicked on Counts on the Map |
| Description | User can click on counts on the map to view the request details by the end users |

Table No 11: Test Case Maps View 2

Test cases for Count View:

| | |
|-----------------|--|
| ID | Count_1 |
| Test Input | No Data input, just User needs to clicks on Counts on Map View |
| Expected Output | Request us details should be display based on counts on the states on map view |
| Description | User can click on counts on the map to view the request details by the end users |

Table No 12: Test Case Count View 1

| | |
|-----------------|--|
| ID | Count_2 |
| Test Input | No Data input |
| Expected Output | Count view with the correct details of the request information and to validate with the count on the maps view |
| Description | User can click on counts on the map to view the request details by the end users |

Table No 13: Test Case Count View 2

6.2. Traceability of test cases to use cases

| Use Case | Requirement Number | Requirement Description | Test Case | Design Component / Module |
|-------------------|--------------------|---|-----------|---|
| Request US Form | CS_001 | End Users can request the services from this form either for self or for others by using proof documents,Name,Location and other details | TCS_001 | Request US Form - Name , Location,Phone Number,Aadhar Number , Proof of documents file upload |
| Mail Notification | CS_002 | After End users clicking on Submit button with the valid data , A mail will be sent to the chatrasahaya team mail id to get to know the requester details. | TCS_002 | Mail Notification with the Requester Details |

| | | | | |
|------------|--------|--|---------|---|
| Map View | CS_003 | Users/Team can view the Counts on the Map for the State as how many services has been sponsored or completed by the Chatrasahaya team | TCS_003 | Map View - Counts on the States |
| Count View | CS_004 | Users/Team can view the details such as Name , Location,Phone Number,Aadhar Number , Proof of documents files of the Counts on the Map for the State as how many services has been sponsored or completed by the Chatrasahaya team | TCS_004 | Count View - Viewing of the Information such as Name , Location,Phone Number,Aadhar Number , Proof of documents file upload |

Table No 14: Traceability of test cases

6.3. Techniques used for test case generation

Equivalence Partitioning:

Divides input data into equivalence classes to reduce the number of test cases while ensuring adequate coverage. It tests one representative from each partition.

Boundary Value Analysis:

Focuses on testing boundaries of input ranges. Test cases are designed at the edges of input domains (minimum, maximum, and just inside/outside boundaries) to uncover potential issues.

Use Case Testing:

Based on use cases or user scenarios, this technique generates test cases to validate the system's behavior from an end-user perspective.

Regression Testing Selection Techniques:

Selects test cases from an existing suite to ensure that changes in the software haven't adversely affected previously tested functionalities.

6.4. Test results and assessments (how good are your test cases? How good is your software?)

Test Results

| Feature Name | Test Results | No of Bugs Reported | Pending Bugs at final |
|-------------------|--------------|---------------------|-----------------------|
| Request US | Pass | 1 | 0 |
| Mail Notification | Pass | 1 | 0 |
| Map View | Pass | 1 | 0 |
| Count View | Pass | 1 | 0 |

Table No 15: Test Results

Assessments

| Assessment Type | Definition | Value/Qty/Status |
|---------------------------|---|------------------|
| Defect Count | Measure the number of defects found in different phases of testing. | 4 |
| Severity | Assess the impact and severity level of each identified defect. | Pass |
| User Feedback | Gather user feedback through surveys or direct interaction. | N/A |
| Usability Testing Results | Evaluate ease of use, user interfaces, and overall user experience | Pass |
| Regression Test Results | Assess if new changes introduced any regression issues by comparing against baseline results. | Pass |
| Security Testing Results | Analyze vulnerabilities and security breaches identified during testing. | Pass |

Table No 16: Assessments

6.5. Defects reports

| Defect ID | Title/Summary | Date Reported | Status | Severity | Priority | Steps to Reproduce | Comments |
|-----------|---|---------------|--------|----------|----------|--|----------|
| CS_1 | No Validation is displayed when user clicked on Request Us without any fields data | 11-02-2023 | Closed | High | High | 1. Click on Request Us link 2. Click on Request Us button in the page without any data in the fields | N/A |
| CS_2 | Mail was not received to the team/user after submitting the request us page details | 11-05-2023 | Closed | Medium | Medium | 1. Click on Request Us link 2. Click on Request Us button in the page with proper data in the fields | N/A |
| CS_3 | Counts was mismatched on the state wise on the map view | 11-25-2023 | Closed | High | High | 1. Click on Request Us link 2. Click on Request Us button in the page with proper data in the fields for certain states 3. Check the counts on the state wise on the map view | N/A |
| CS_4 | Details was not matched in the count view when user clicked on count for the state | 11-29-2023 | Closed | High | High | 1. Click on Request Us link 2. Click on Request Us button in the page with proper data in the fields for certain states 3. Click on count on the state 4. Check the details of the count view displayed | N/A |

Table No 17: Defect Report

7. Conclusions

7.1. Outcomes of the project (are all goals achieved?)

Yes all the goals achieved successfully and the outcomes of the Project are listed below :

Enhancements as completed as per the Plan such as Request Us,Mail Notification,Map & Count view for the chatrasahaya website

7.2. Lessons learned

We need to contact the API team like Google Maps team initially before planning whether they provide the API or not for our services as we faced some difficult getting the API Key for the Maps view

7.3. Future development

We will plan to the develop the Mobile Application for the same website with the same features one by one.

References

<https://chatrasahaya.org/>