# LAB REPORT

*Submitted by*

**SARTHAK KAUSHAL [RA211026010286]**
**TANMAY GUPTA [RA2111026010288]**
**SHREEYA CHAUHAN [RA2111026010276]**

*Under the Guidance of*

**Ms. Sasi Rekha Shankar**

**Assistant Professor, CINTEL**

***In partial satisfaction of the requirements for the degree of***

**BACHELOR OF TECHNOLOGY**
**in**
**COMPUTER SCIENCE ENGINEERING**

**with specialization in <AI/ML>**

**SCHOOL OF COMPUTING**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**KATTANKULATHUR - 603203**

**MAY 2023**

COLLEGE OF ENGINEERING & TECHNOLOGY
SRM INSTITUTE OF SCIENCE & TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUE – 603 203
Chengalpattu District

# BONAFIDE CERTIFICATE

Register No. RA2111026010288, RA2111026010286, RA2111026010276 Certified to be the bonafide work done by SARTHAK KAUSHAL, TANMAY GUPTA, SHREEYA CHAUHAN of II Year/IV Sem B.Tech Degree Course in the **Practical Software Engineering and Project Management 18CSC206J** in **SRM INSTITUTE OF SCIENCE AND TECHNOLOGY,** Kattankulathur during the academic year 2022 – 2023.

**SIGNATURE**

**Sasi Rekha Shankar**
**Assistant Professor**
**Department of CINTEL**
**SRMIST – KTR.**

**SIGNATURE**

**Dr.R.Annie Uthra**
**HOD CINTEL**

**Date :**

# TABLE OF CONTENTS

## LIST OF FIGURES

# LIST OF TABLES

# ABSTRACT

2-Chasma is a comprehensive web application that aims to enhance the student experience by providing a platform for discovering local restaurants and places of interest. The website features real-time location sharing, feedback system, and travel time estimation, all with the goal of ensuring that students can explore safely and conveniently. The development of 2-Chasma followed a structured approach, starting with requirements gathering to identify the needs of users, followed by design, implementation, and testing, to deliver a functional and reliable website.

The project faced various challenges, including technical complexities and timeline pressures. However, with effective project management, communication, and teamwork, these challenges were addressed, and the project progressed smoothly towards completion. The success of the project is attributed to the collective effort and dedication of the development team, as well as the support and guidance of stakeholders and project sponsors.

2-Chasma offers immense value to students by providing a user-friendly interface to explore local restaurants and places of interest. The website provides extensive information on each location, including its description, images, location, and ratings, making it easy for students to make informed decisions. The real-time location sharing feature enables students to share their location with others, facilitating coordination and safety. Moreover, the feedback system provides a mechanism for students to provide their opinions and suggestions, allowing the website to continuously improve and enhance the user experience

# CREATE A BUSINESS CASE

**Aim :** To Frame a project team, analyze and identify a Software project. To create a business case and Arrive at a Problem Statement for the < 2—CHASMA >

## Project Description

The project focuses on the necessities of the new students on the campus providing them with suggestions of places to visit, restaurants, bus stops, rickshaw stands and prices, medical stores, etc.

## THE PROJECT LOGO

Figure 1.1 – 2 – Chasma Logo

## THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

- Focuses on easier mobility for the students according to hostel and rental living experiences.
- Reduces the major confusion of students around the campus and places to visit and explore.
- Helps find basic amenity stops.

## THE HISTORY

In bullet points, describe the current situation.

- Currently students get exploited due to being new here on the campus
- They often lose out on places to explore
- The stores and pharmacy are not well known to the students

## LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack of special training, etc.

- Data security and privacy: we handle  information and must ensure that this information is protected from unauthorized access or theft

## APPROACH

List what is needed to complete the project.

- Database: Lots of data about restaurant and places
- Build the platform: The next step is to build the platform that will be used to access the data by users

## BENEFITS

In bullet points, list the benefits that this project will bring to the organization.

- Efficiency: we can use technology to streamline the process, making it faster and convenient for user and admin
- Accessibility: we can make it easier for people to access the data  that may have been previously out of reach
- Diversification: we can offer a wide range of places and food restaurant , which can help user to diversify their choice

# STAKE HOLDER AND USER DESCRIPTION

**Aim :** To identify the appropriate Process Model for the project and prepare Stakeholder and User Description.

## Project Title :

Selection of Metho2-logy

- For the proposed project, we will use the spiral method to develop it as it is a user website, lot of advancements and features will be introduced based on user experience and real time & life problems.
- Constant research and development will be a regular part of the project.

| Stakeholder Name | Activity/ Area /Phase | Interest | Influence | Priority (High/ Medium/ Low) |
|---|---|---|---|---|
| The developers | Development of the website | Effective functioning of the app and active user experience. | Medium | High |
| Shop – owners | Provide data and service | Increasing their Business | High | Medium |
| Users | Providing feedback | Using a friendly website | High | Medium |

Table No. 2.1 – Priority Table

# FUNCTIONAL AND NON-FUNCTIONAL REQUIRMENTS

**Aim :** To identify the system, functional and non-functional requirements for the project.

## System Requirements

### Hardware

- A modern computer or laptop with a minimum of 4GB of RAM and a dual-core processor

.

- A high-speed internet connection.

### Software

- Operating System: Win2-ws 7 or later, Mac OS X 10.12 (Sierra) or later, or any modern Linux distribution.
- Web browser: Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge (latest version).
- A code editor such as Visual Studio Code, Sublime Text, or Atom.
- A local web server environment such as XAMPP, WAMP, or MAMP.
- A version control system like Git.

### Technical Skills

- Basic knowledge of HTML, CSS, and JavaScript.
- Familiarity with a server-side scripting language like PHP or Node.js.
- Experience with a relational database management system like MySQL or PostgreSQL.
- Knowledge of how to use version control systems such as Git.

## Functional Requirements

User Account Management

- Allow users to create an account and log in to the website.
- Provide users with the ability to reset their password in case they forget it.
- Store and manage user profiles, including personal information, interests, and preferences.

Place Discovery and Exploration

- Provide a search function that allows users to find places to visit based on various criteria such as location, type of activity, or interests.
- Display a list of places to visit with detailed information, including images, description, location, and contact information.
- Allow users to save places they are interested in visiting to a list of favorites.

User Reviews and Ratings

- Provide a platform for users to write reviews and rate places they have visited.
- Display an average rating for each place based on all user reviews.
- Allow users to view reviews from other users for each place.

Map Integration

- Integrate a map to display the location of places to visit and provide directions to each location.
- Allow users to search for places within a specific area.

Content Management

- Provide a backend interface for administrators to manage and add new places to the website.
- Allow administrators to edit existing place information, approve and manage user reviews, and monitor user activity.

Mobile Compatibility

- Ensure that the website is mobile-friendly and can be easily viewed on mobile devices.

- Provide a mobile-optimized version of the site that offers a similar experience to the desktop version.

Performance and Security

- Ensure that the website loads quickly and is responsive to user interactions.
- Implement appropriate security measures to protect user data and prevent unauthorized access to the website.

**Non-Functional Requirements**

Here are the non-functional requirements according to McCall's 11 quality factors:

Usability

- The website should be easy to use and navigate for new students.
- The user interface should be intuitive and visually appealing.
- The search and filter functions should be straightforward and effective.

Reliability

- The website should be stable and function without crashing or error messages.
- The website should be available and accessible 24/7.
- The website should be able to handle high traffic periods.

Performance

- The website should load quickly and be responsive to user interactions.
- The website should be optimized for fast loading times on both desktop and mobile devices.
- The website should be able to handle large amounts of data and search queries efficiently.

Security

- The website should protect user data and prevent unauthorized access.
- The website should use encryption to secure sensitive information, such as passwords.
- The website should be regularly updated to address security vulnerabilities.

Compatibility

- The website should be compatible with modern web browsers, including Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.
- The website should be accessible on both desktop and mobile devices.

Maintainability

- The website should be easy to maintain and update by the development team.
- The website should be built using modular, scalable, and maintainable code.
- The website should have clear 2-cumentation to help developers understand how it works.

Portability

- The website should be built using open-source technologies that are widely supported.
- The website should be easy to transfer to a different hosting environment if needed.

Reusability

- The website should be built using modular components that can be reused in future projects.
- The code should be well-2-cumented and structured to make it easy to reuse.

Testability

- The website should be built with testing in mind, and it should be easy to test and debug.
- The website should have automated tests in place to ensure that it functions as expected.

Interoperability

- The website should be able to integrate with other systems and APIs if needed.
- The website should be able to share data with other systems in a secure and controlled manner.

Scalability

- The website should be able to handle increased traffic and data volume as the user base grows.

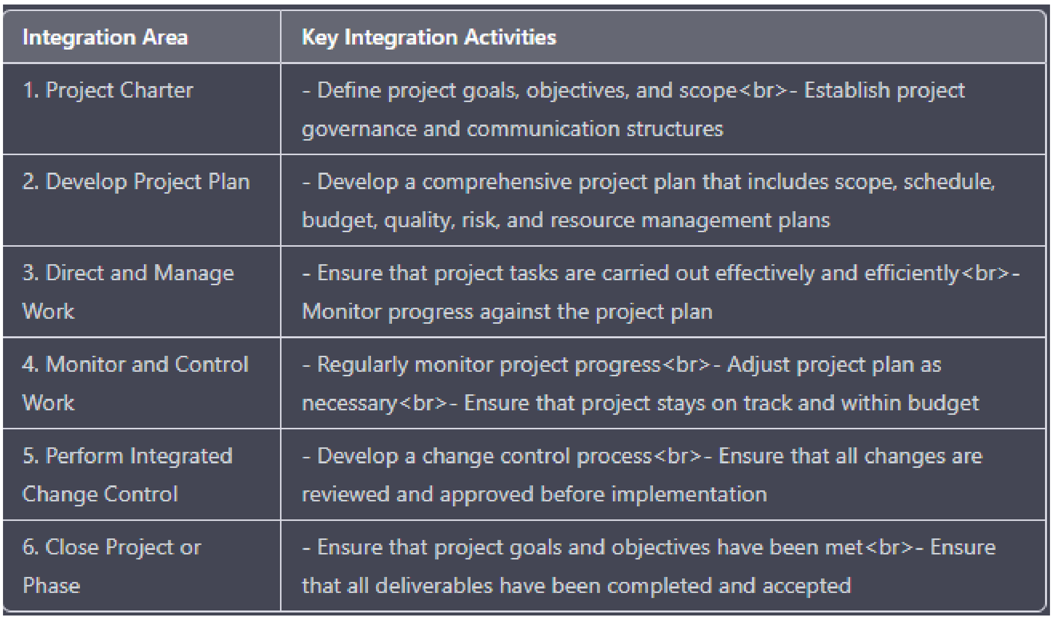
- The website should be designed with scalability in mind, using scalable technologies and infrastructure.

# PROJECT PLAN

**Aim :** To Prepare Project Plan based on scope, Calculate Project effort based on resources, Find Job roles and responsibilities

## Requirements

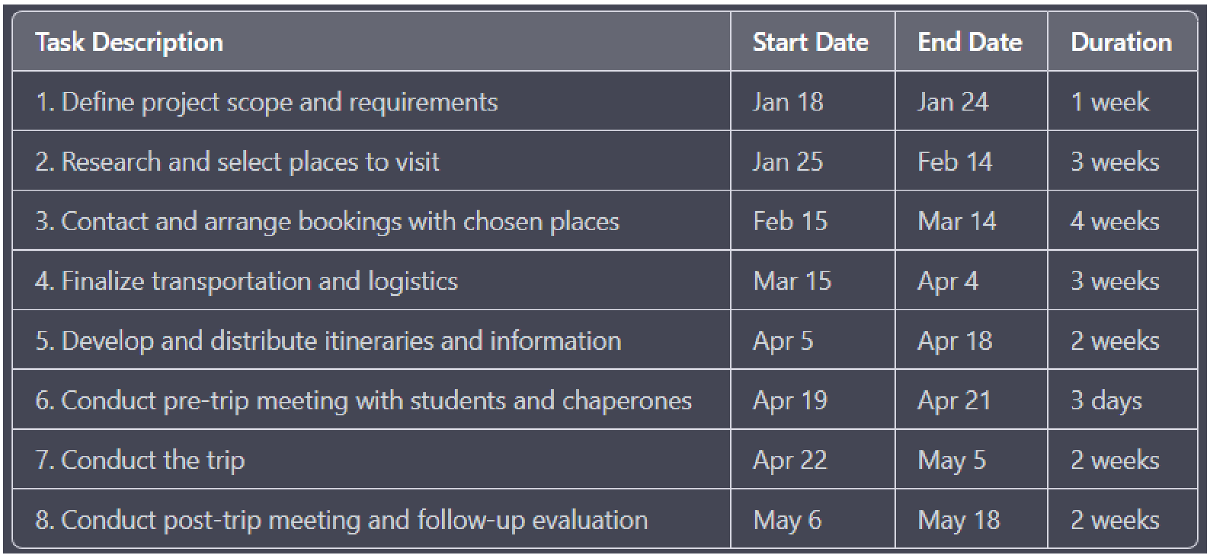
### 1. Integration Management for 2-Chasma



| Integration Area | Key Integration Activities |
|---|---|
| 1. Project Charter | - Define project goals, objectives, and scope<br>- Establish project governance and communication structures |
| 2. Develop Project Plan | - Develop a comprehensive project plan that includes scope, schedule, budget, quality, risk, and resource management plans |
| 3. Direct and Manage Work | - Ensure that project tasks are carried out effectively and efficiently<br>- Monitor progress against the project plan |
| 4. Monitor and Control Work | - Regularly monitor project progress<br>- Adjust project plan as necessary<br>- Ensure that project stays on track and within budget |
| 5. Perform Integrated Change Control | - Develop a change control process<br>- Ensure that all changes are reviewed and approved before implementation |
| 6. Close Project or Phase | - Ensure that project goals and objectives have been met<br>- Ensure that all deliverables have been completed and accepted |

Table no. 4.1 – Intergration Management Table

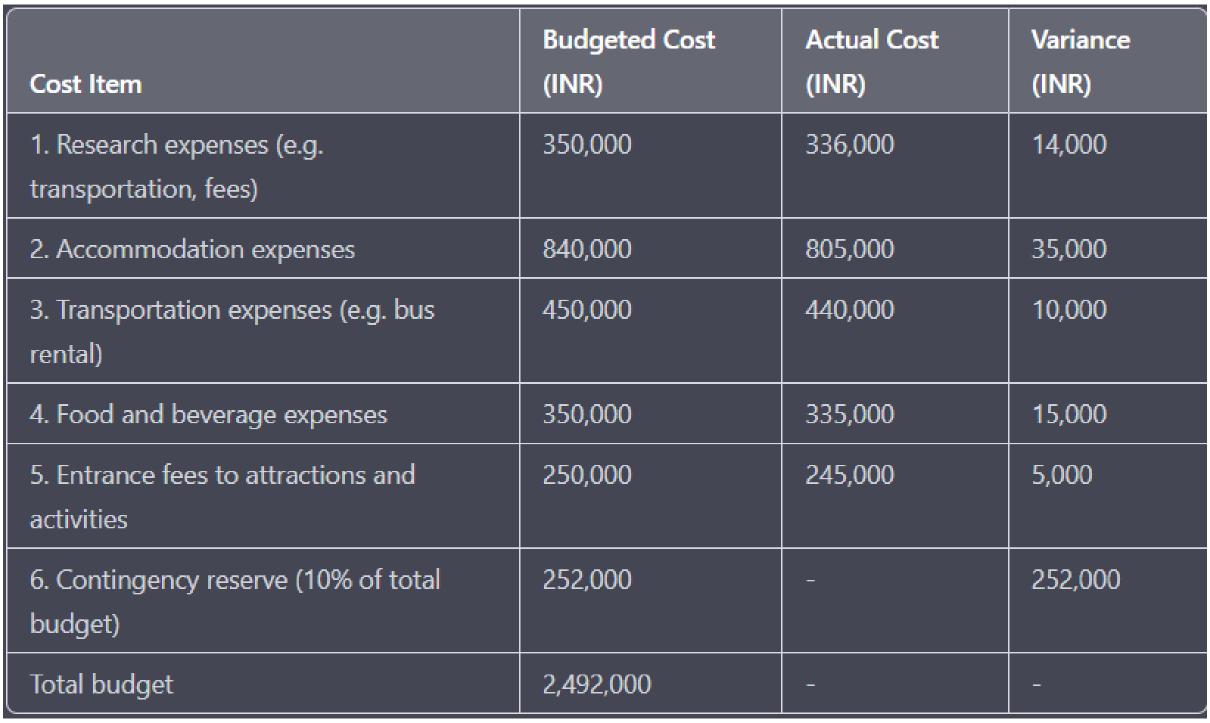
## 2. Schedule Management for 2-Chasma



| Task Description | Start Date | End Date | Duration |
|---|---|---|---|
| 1. Define project scope and requirements | Jan 18 | Jan 24 | 1 week |
| 2. Research and select places to visit | Jan 25 | Feb 14 | 3 weeks |
| 3. Contact and arrange bookings with chosen places | Feb 15 | Mar 14 | 4 weeks |
| 4. Finalize transportation and logistics | Mar 15 | Apr 4 | 3 weeks |
| 5. Develop and distribute itineraries and information | Apr 5 | Apr 18 | 2 weeks |
| 6. Conduct pre-trip meeting with students and chaperones | Apr 19 | Apr 21 | 3 days |
| 7. Conduct the trip | Apr 22 | May 5 | 2 weeks |
| 8. Conduct post-trip meeting and follow-up evaluation | May 6 | May 18 | 2 weeks |

Table No. 4.2 – Schedule Management Table

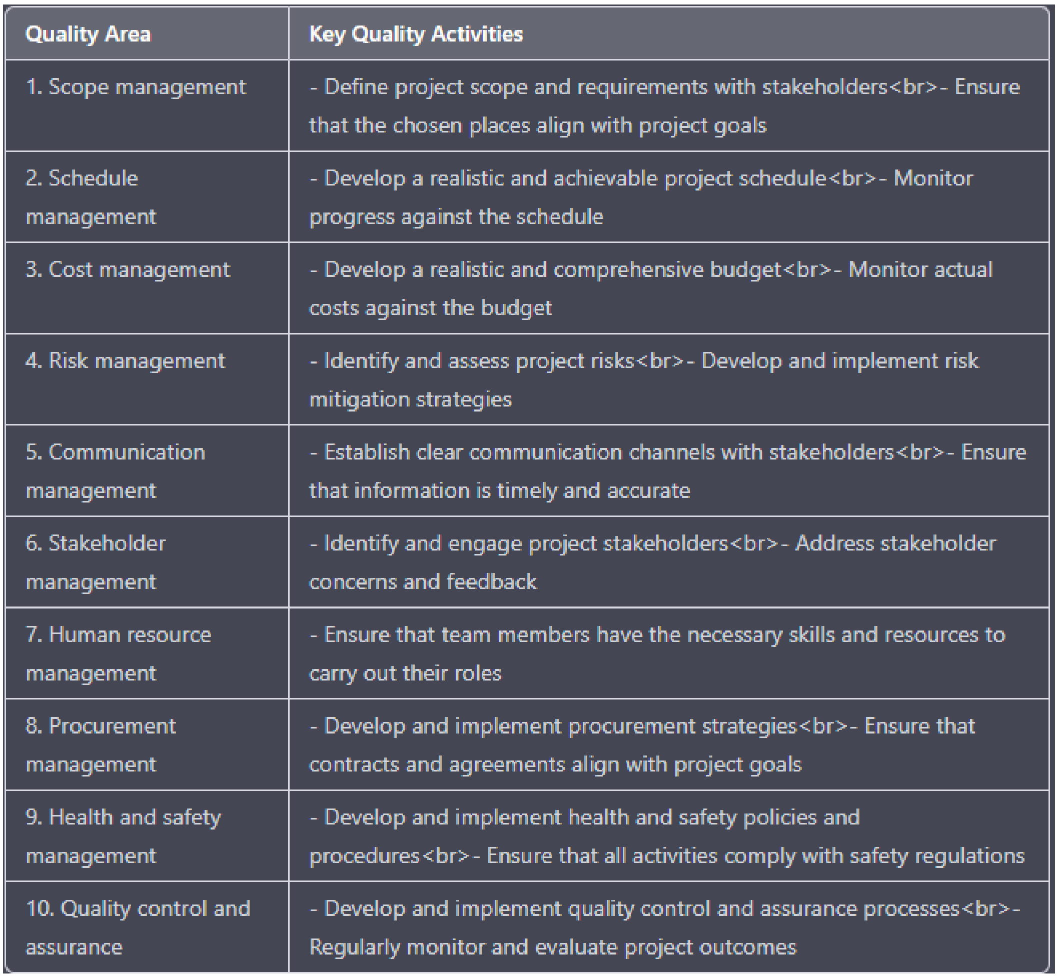
## 3. Cost Management for 2-Chasma



| Cost Item | Budgeted Cost (INR) | Actual Cost (INR) | Variance (INR) |
|---|---|---|---|
| 1. Research expenses (e.g. transportation, fees) | 350,000 | 336,000 | 14,000 |
| 2. Accommodation expenses | 840,000 | 805,000 | 35,000 |
| 3. Transportation expenses (e.g. bus rental) | 450,000 | 440,000 | 10,000 |
| 4. Food and beverage expenses | 350,000 | 335,000 | 15,000 |
| 5. Entrance fees to attractions and activities | 250,000 | 245,000 | 5,000 |
| 6. Contingency reserve (10% of total budget) | 252,000 | - | 252,000 |
| Total budget | 2,492,000 | - | - |

Table 4.3 – Cost Management Table

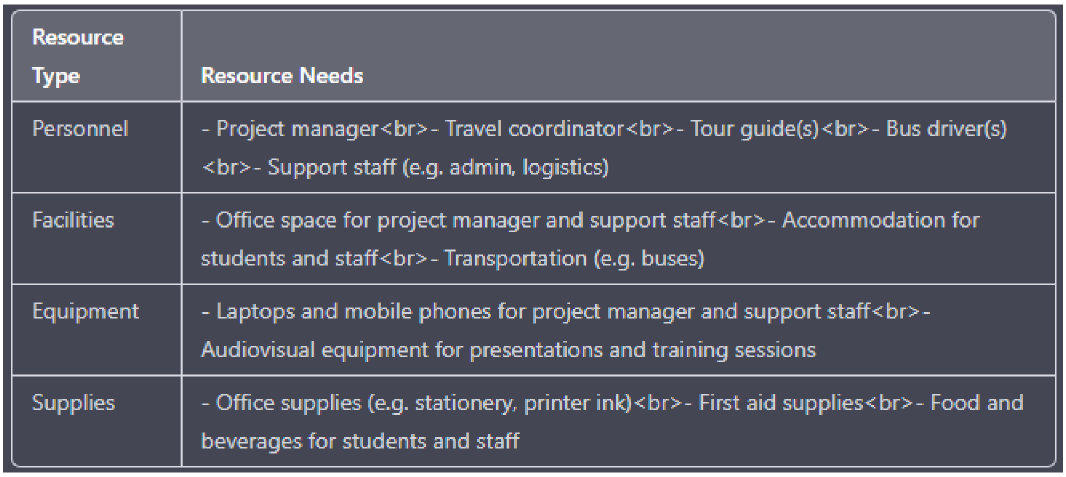
## 4. Quality Management for 2-Chasma



| Quality Area | Key Quality Activities |
|---|---|
| 1. Scope management | - Define project scope and requirements with stakeholders<br>- Ensure that the chosen places align with project goals |
| 2. Schedule management | - Develop a realistic and achievable project schedule<br>- Monitor progress against the schedule |
| 3. Cost management | - Develop a realistic and comprehensive budget<br>- Monitor actual costs against the budget |
| 4. Risk management | - Identify and assess project risks<br>- Develop and implement risk mitigation strategies |
| 5. Communication management | - Establish clear communication channels with stakeholders<br>- Ensure that information is timely and accurate |
| 6. Stakeholder management | - Identify and engage project stakeholders<br>- Address stakeholder concerns and feedback |
| 7. Human resource management | - Ensure that team members have the necessary skills and resources to carry out their roles |
| 8. Procurement management | - Develop and implement procurement strategies<br>- Ensure that contracts and agreements align with project goals |
| 9. Health and safety management | - Develop and implement health and safety policies and procedures<br>- Ensure that all activities comply with safety regulations |
| 10. Quality control and assurance | - Develop and implement quality control and assurance processes<br>- Regularly monitor and evaluate project outcomes |

Table No. 4.4 – Quality Management Table

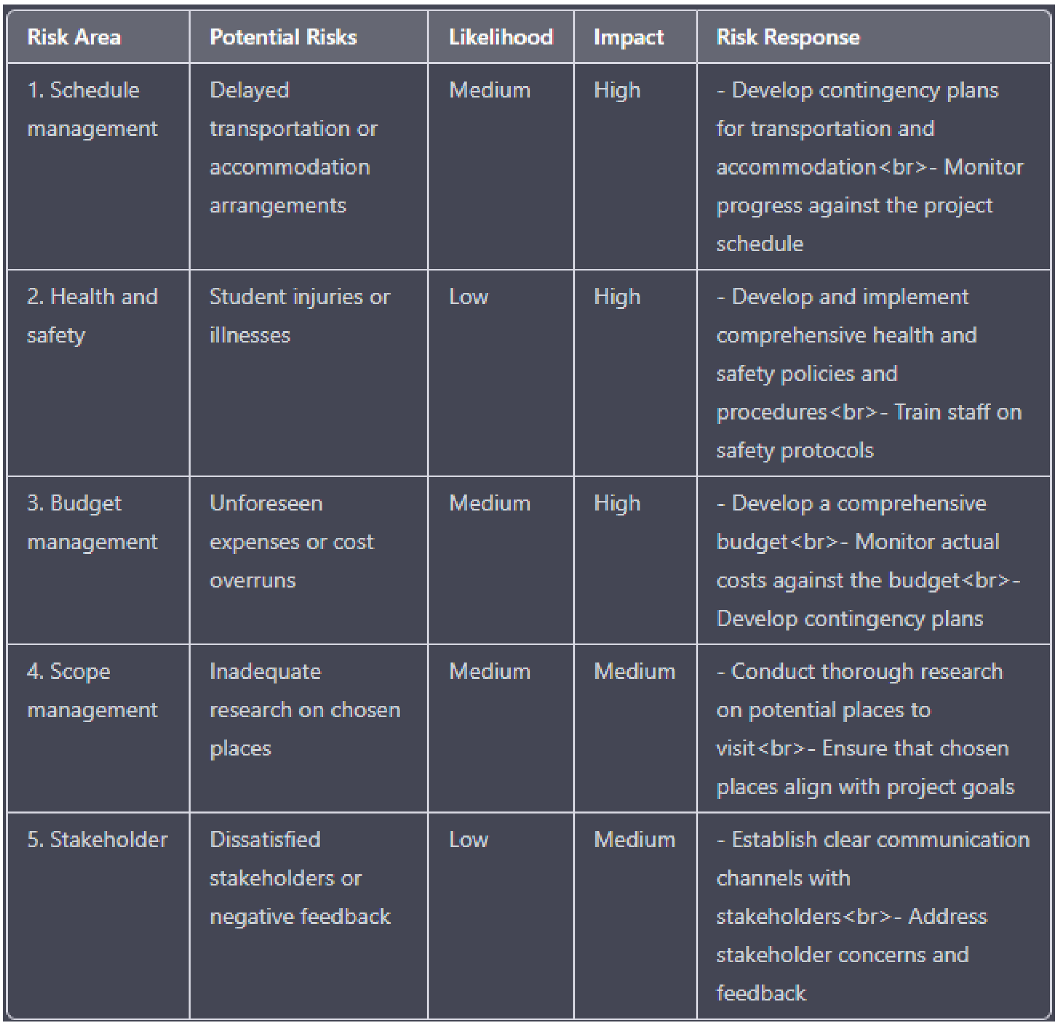
## 5. Resource management for 2-Chasma



| Resource Type | Resource Needs |
|---|---|
| Personnel | - Project manager<br>- Travel coordinator<br>- Tour guide(s)<br>- Bus driver(s)<br>- Support staff (e.g. admin, logistics) |
| Facilities | - Office space for project manager and support staff<br>- Accommodation for students and staff<br>- Transportation (e.g. buses) |
| Equipment | - Laptops and mobile phones for project manager and support staff<br>- Audiovisual equipment for presentations and training sessions |
| Supplies | - Office supplies (e.g. stationery, printer ink)<br>- First aid supplies<br>- Food and beverages for students and staff |

Table 4.5 – Resource Management Table

## 6. Risk Management  for 2-Chasma

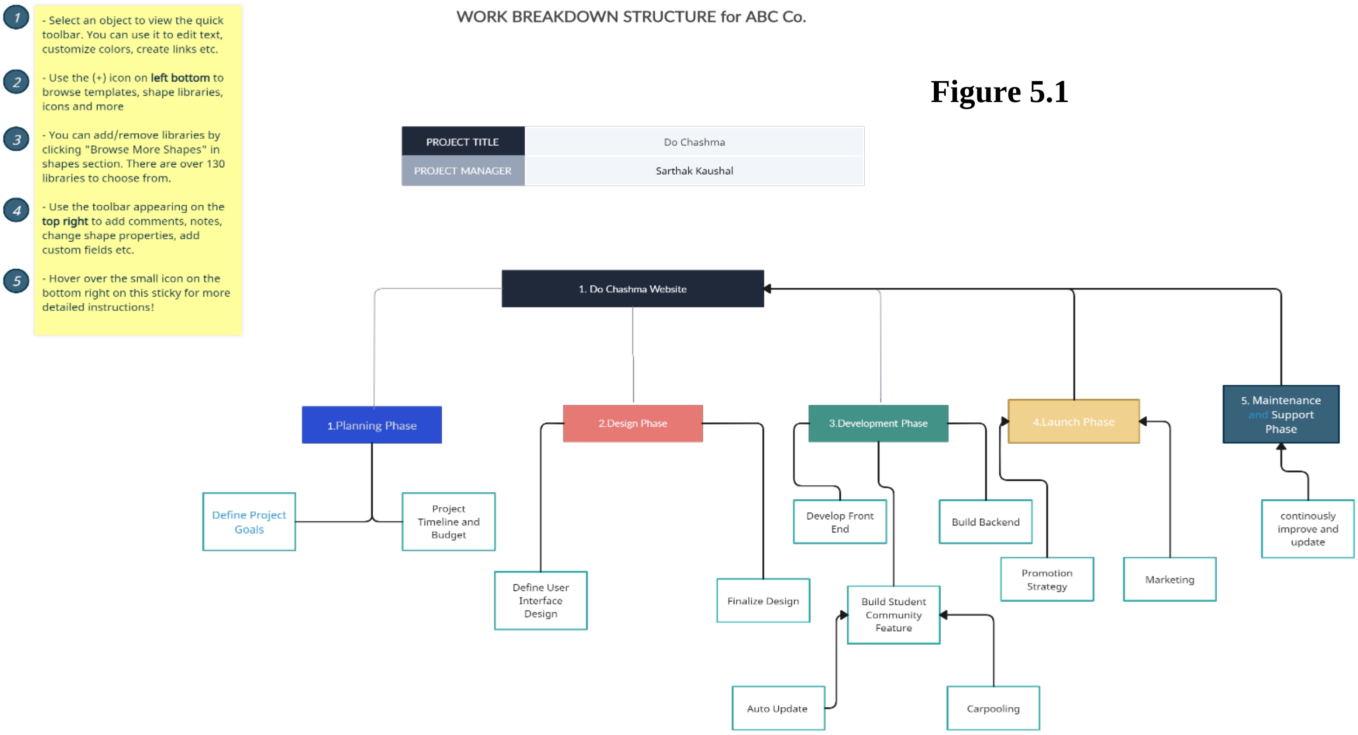


| Risk Area | Potential Risks | Likelihood | Impact | Risk Response |
|---|---|---|---|---|
| 1. Schedule management | Delayed transportation or accommodation arrangements | Medium | High | - Develop contingency plans for transportation and accommodation<br>- Monitor progress against the project schedule |
| 2. Health and safety | Student injuries or illnesses | Low | High | - Develop and implement comprehensive health and safety policies and procedures<br>- Train staff on safety protocols |
| 3. Budget management | Unforeseen expenses or cost overruns | Medium | High | - Develop a comprehensive budget<br>- Monitor actual costs against the budget<br>- Develop contingency plans |
| 4. Scope management | Inadequate research on chosen places | Medium | Medium | - Conduct thorough research on potential places to visit<br>- Ensure that chosen places align with project goals |
| 5. Stakeholder | Dissatisfied stakeholders or negative feedback | Low | Medium | - Establish clear communication channels with stakeholders<br>- Address stakeholder concerns and feedback |

Table 4.6 – Risk Management Table

# BREAKDOWN STRUCTURE,TIMELINE CHARTAND RISK IDENTIFICATION TABLE

**Aim :** To Prepare Work breakdown structure, Timeline chart and Risk identification table



1 - Select an object to view the quick toolbar. You can use it to edit text, customize colors, create links etc.

2 - Use the (+) icon on **left bottom** to browse templates, shape libraries, icons and more

3 - You can add/remove libraries by clicking "Browse More Shapes" in shapes section. There are over 130 libraries to choose from.

4 - Use the toolbar appearing on the **top right** to add comments, notes, change shape properties, add custom fields etc.

5 - Hover over the small icon on the bottom right on this sticky for more detailed instructions!

WORK BREAKDOWN STRUCTURE for ABC Co.

**Figure 5.1**

| | |
|---|---|
| PROJECT TITLE | Do Chashma |
| PROJECT MANAGER | Sarthak Kaushal |

Figure 5.1 – Work Breakdown

It is a hierarchical decomposition of the project into smaller, more manageable parts. It outlines the major phases of the project, including Planning, Design, Development, Launch, and Maintenance and Support, and breaks down each phase into a series of tasks and sub-tasks that must be completed to achieve the project's objectives.

The Planning Phase includes tasks such as defining project goals and objectives, identifying the target audience and user needs, conducting market research and competitive analysis, and developing the project timeline and budget.

The Design Phase includes tasks such as developing user personas and scenarios, defining user interface design and user experience (UI/UX) requirements, creating wireframes and mock ups for each feature, developing branding, logo, and visual design, and finalizing the design and obtaining stakeholder approval.

The Development Phase includes tasks such as developing the front-end interface and user experience, building the back-end database and server infrastructure, developing each of the features (Places to Visit, Student Community, Carpooling, Events and Activities, and Feedback and Reviews), testing and refining the features, integrating all features into a cohesive application, conducting user acceptance testing (UAT), and obtaining feedback.

The Launch Phase includes tasks such as preparing for launch, including beta testing and final bug fixes, launching the app on designated app stores, creating a marketing and promotion strategy, collecting and analysing user feedback to make improvements, and continuing to monitor app performance and user engagement.

Finally, the Maintenance and Support Phase includes tasks such as providing ongoing maintenance and support to ensure the app runs smoothly, continuously improving and updating the app based on user feedback and market trends, providing customer support and responding to user inquiries and complaints, and staying up to date with technology advancements and industry best practices to ensure the app remains relevant and competitive.

Overall, the WBS provides a clear and organized roadmap for completing the project on time, within budget, and to the satisfaction of all stakeholders.

## Time Line and Gantt Chart for 2-Chasma

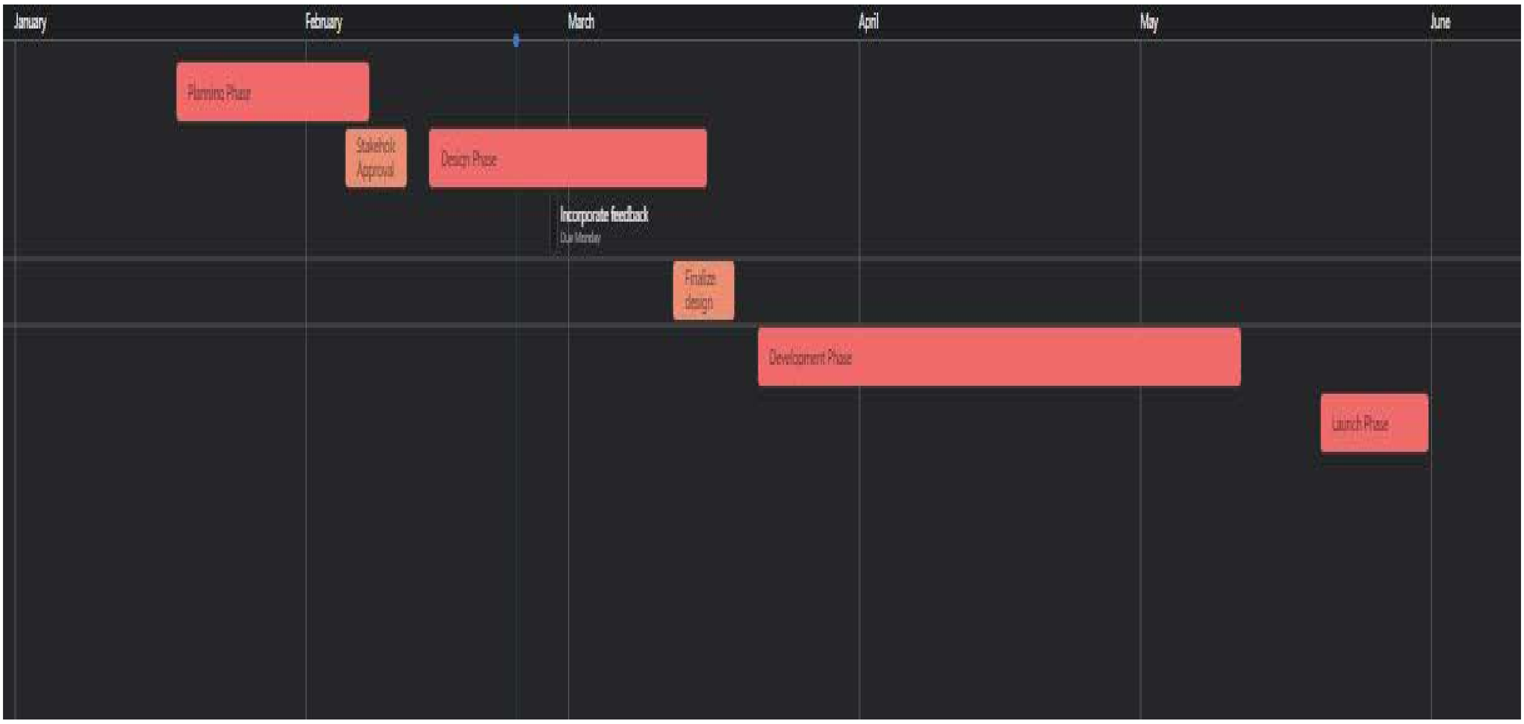


Figure 5.2 – Timeline and Gantt Chart

## RISK ANALYSIS – SWOT & RMMM FOR 2-Chasma

| Strengths | Weaknesses |
|---|---|
| Provides a platform for new university students to find places to visit and connect with other students | Limited budget for development and marketing |
| Offers a solution to the common problem of transportation for students | Limited team size and resources |
| Provides a community for students to discuss carpooling and other issues | Lack of experience and reputation compared to established competitors |
| User-friendly interface and design | Dependency on third-party APIs for some features |
| High potential for growth and expansion | Possible technical issues and bugs during development and launch |

| **Opportunities** | **Threats** |
|---|---|
| Growing demand for carpooling and ride-sharing services | Strong competition from established players in the market |
| Increasing focus on sustainable and eco-friendly transportation options | Potential legal and regulatory issues related to ride-sharing and carpooling |
| Potential for partnerships with universities and local businesses | Technological changes and disruptions that could impact the app's functionality and user experience |
| Ability to expand to other markets and regions | Economic Downturns or market fluctuations that could impact user adoption and engagement |
| Potential for monetization through advertising, partnerships, or premium features | Negative publicity or reviews that could impact the app's reputation and user trust |

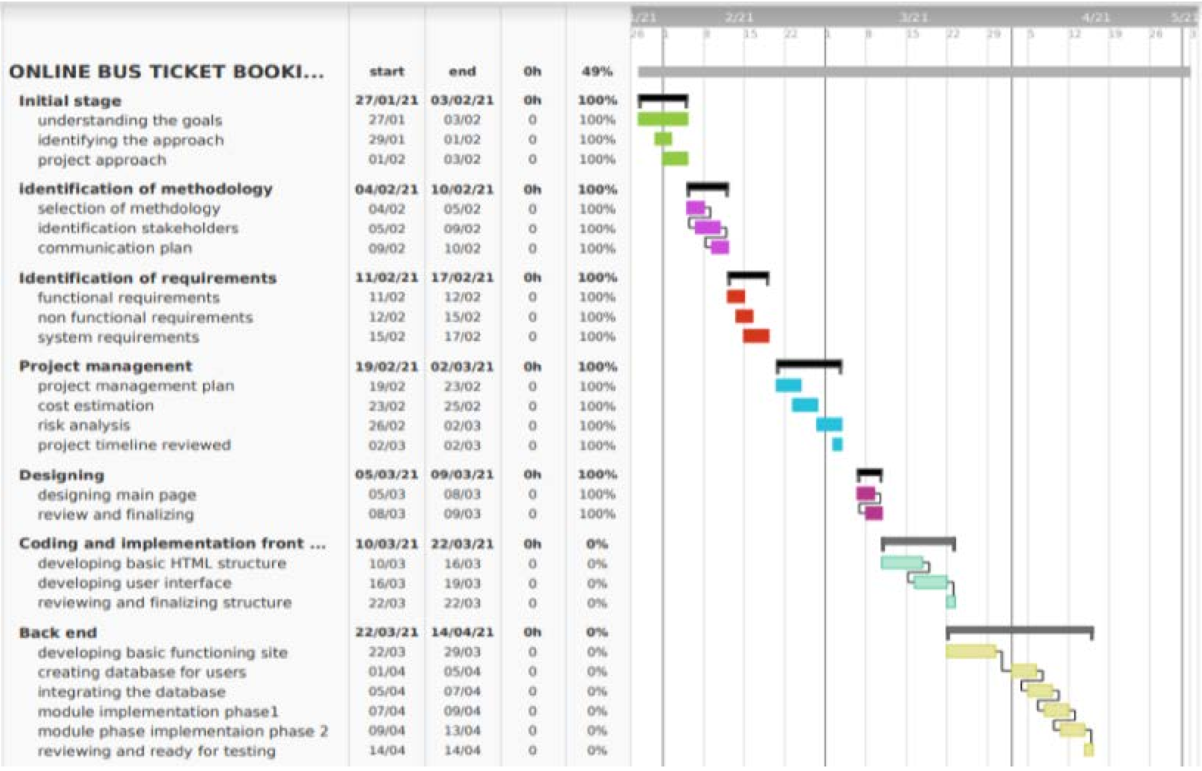
Table 5.1 – SWOT

| Risk | Likelihood | Impact | Mitigation Strategy | Monitoring Strategy |
|---|---|---|---|---|
| Technical issues and bugs during development and launch | High | High | Conduct thorough testing and quality assurance before launch. Develop a feedback system to quickly identify and address issues reported by users. | Monitor user feedback and ratings. Continuously test and optimize the app's functionality and performance. |
| Strong competition from established players in the market | Medium | High | Differentiate the app by focusing on unique features, such as providing a community for students to discuss carpooling and places to visit. Develop a strong marketing strategy to increase brand awareness and user acquisition. | Monitor market trends and competitor activities. Continuously assess the app's unique value proposition and adjust the strategy as needed. |
| Potential legal and regulatory issues related to ride-sharing and carpooling | Medium | High | Conduct thorough research and comply with all relevant laws and regulations. Develop clear policies and guidelines for users to follow. | Monitor changes in laws and regulations related to ride-sharing and carpooling. Continuously review and update policies and guidelines as needed. |

| Risk | Likelihood | Impact | Mitigation Strategy | Monitoring Strategy |
|---|---|---|---|---|
| Limited budget for development and marketing | High | Medium | Prioritize features and marketing activities based on ROI and impact. Consider alternative funding sources, such as grants, sponsorships, or crowdfunding. | Monitor expenses and ROI. Continuously review and adjust the budget and funding strategy as needed. |
| Dependency on third-party APIs for some features | Medium | Medium | Develop contingency plans in case of API downtime or unavailability. Consider building alternative solutions or features. | Monitor the API provider's performance and reliability. Continuously test and optimize the app's integration with third-party APIs. |

Table 5.2 - RMMM

## TIMELINE – GANTT CHART FOR 2-CHASMA



| ONLINE BUS TICKET BOOKI... | start | end | 0h | 49% |
|---|---|---|---|---|
| **Initial stage** | **27/01/21** | **03/02/21** | **0h** | **100%** |
| understanding the goals | 27/01 | 03/02 | 0 | 100% |
| identifying the approach | 29/01 | 01/02 | 0 | 100% |
| project approach | 01/02 | 03/02 | 0 | 100% |
| **identification of methodology** | **04/02/21** | **10/02/21** | **0h** | **100%** |
| selection of methdology | 04/02 | 05/02 | 0 | 100% |
| identification stakeholders | 05/02 | 09/02 | 0 | 100% |
| communication plan | 09/02 | 10/02 | 0 | 100% |
| **Identification of requirements** | **11/02/21** | **17/02/21** | **0h** | **100%** |
| functional requirements | 11/02 | 12/02 | 0 | 100% |
| non functional requirements | 12/02 | 15/02 | 0 | 100% |
| system requirements | 15/02 | 17/02 | 0 | 100% |
| **Project managenent** | **19/02/21** | **02/03/21** | **0h** | **100%** |
| project management plan | 19/02 | 23/02 | 0 | 100% |
| cost estimation | 23/02 | 25/02 | 0 | 100% |
| risk analysis | 26/02 | 02/03 | 0 | 100% |
| project timeline reviewed | 02/03 | 02/03 | 0 | 100% |
| **Designing** | **05/03/21** | **09/03/21** | **0h** | **100%** |
| designing main page | 05/03 | 08/03 | 0 | 100% |
| review and finalizing | 08/03 | 09/03 | 0 | 100% |
| **Coding and implementation front ...** | **10/03/21** | **22/03/21** | **0h** | **0%** |
| developing basic HTML structure | 10/03 | 16/03 | 0 | 0% |
| developing user interface | 16/03 | 19/03 | 0 | 0% |
| reviewing and finalizing structure | 22/03 | 22/03 | 0 | 0% |
| **Back end** | **22/03/21** | **14/04/21** | **0h** | **0%** |
| developing basic functioning site | 22/03 | 29/03 | 0 | 0% |
| creating database for users | 01/04 | 05/04 | 0 | 0% |
| integrating the database | 05/04 | 07/04 | 0 | 0% |
| module implementation phase1 | 07/04 | 09/04 | 0 | 0% |
| module phase implementaion phase 2 | 09/04 | 13/04 | 0 | 0% |
| reviewing and ready for testing | 14/04 | 14/04 | 0 | 0% |

Fig 5.3 – Timeline – Gantt Chart

# USE CASE AND CLASS DIAGRAM

**Aim :** To Design a System Architecture, Use case and Data Flow Diagram

## SYSTEM ARCHITECTURE DESIGN FOR 2-CHASMA

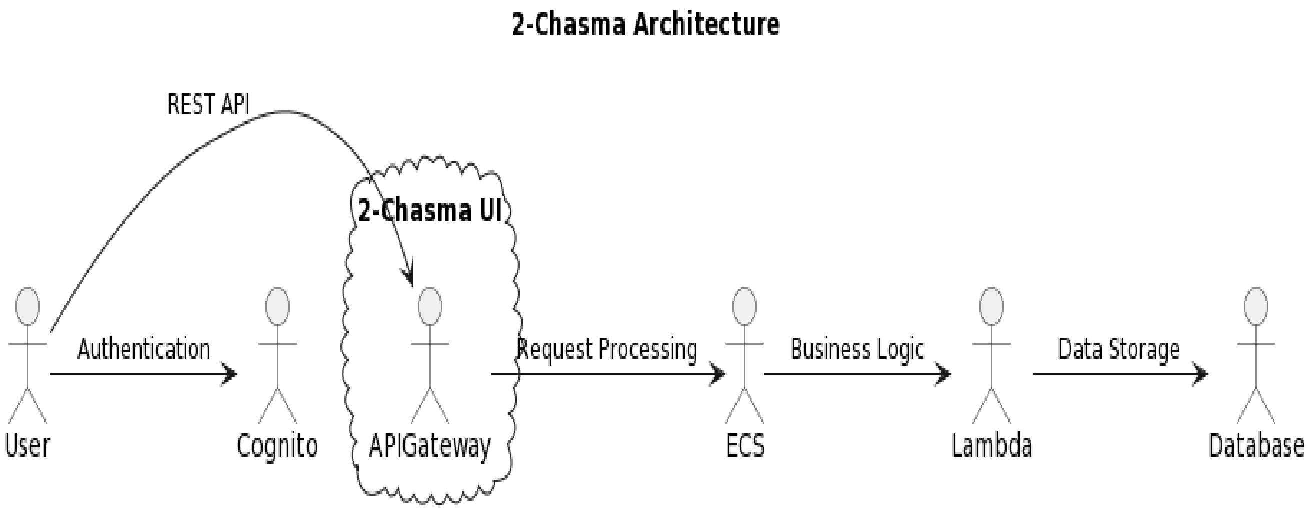


Figure 6.1 – System Architecture Design

In this updated diagram, you can see that the Web Server component has been expanded to include several additional details:

**API Handlers :** This component is responsible for handling incoming API requests and routing them to the appropriate endpoint. It may also be responsible for parsing and validating request parameters and formatting responses.

**Authentication :** This component is responsible for authenticating users and generating access tokens that can be used to authorize subsequent API requests.

**Authorization :**This component is responsible for authorizing API requests based on the access tokens generated by the Authentication component. It may also enforce access control policies to ensure that users can only access data that they are authorized to see.

**Data Access Layer :** This component is responsible for encapsulating the database access logic and providing an API for the Web Application to interact with the database. It may also implement caching or other optimizations to improve performance.

**Database Connection Pool** : This component is responsible for managing a pool of database connections that can be used by the Data Access Layer to interact with the database. This can improve performance and scalability by reducing the overhead of creating and destroying database connections.

## USE CASE DIAGRAM FOR 2-Chasma

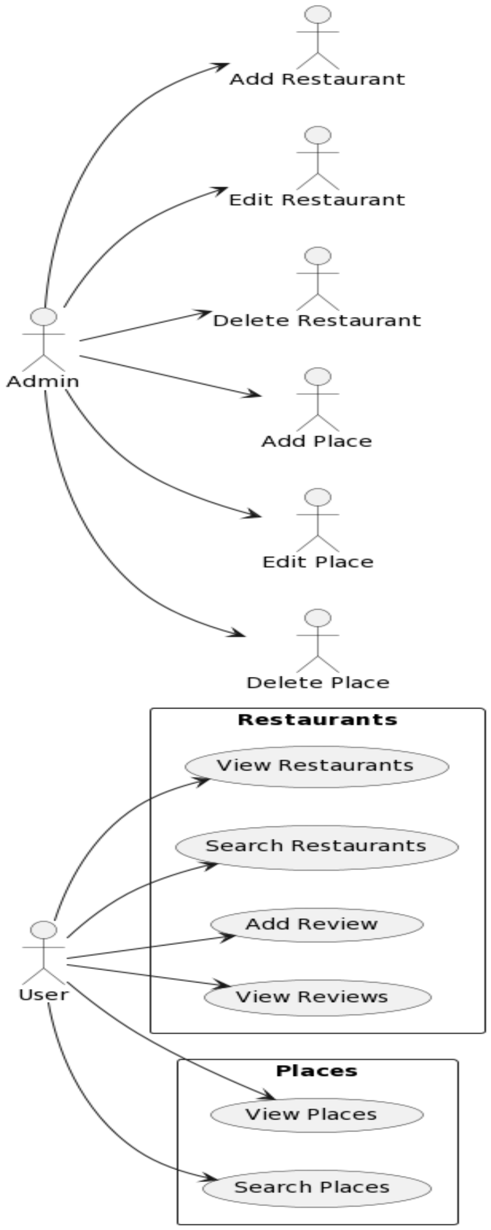


Figure 6.2 – Use Case Diagram

In this example, use case diagram, you can see that there are two main rectangles, Restaurants and Places, representing the different types of entities that can be interacted with in the system.

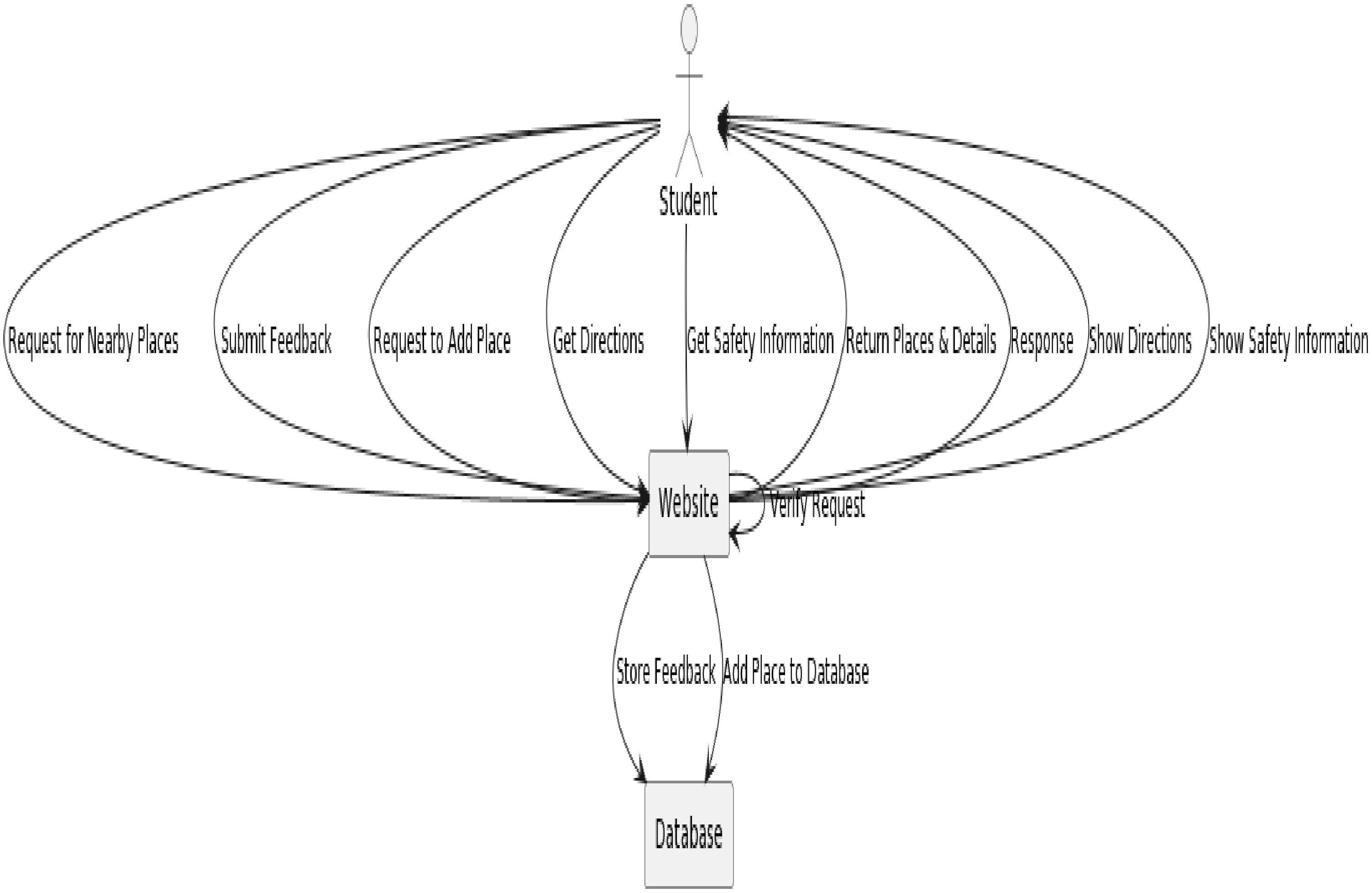
The User actor can perform several use cases, such as viewing and searching for restaurants and places, adding and viewing reviews, and so on.

The Admin actor, on the other hand, has access to additional use cases, such as adding, editing, and deleting restaurants and places. By using actors and use cases, we can model how different users will interact with the system and what functionality they will have access to.

**DATA FLOW DIAGRAM UPTO LEVEL 1 :**

In this diagram, the "User" actor interacts with the "2-Chasma System" through data flows. The data flows are represented by arrows, which indicate the direction of data movement. The data flows pass through three components: "Input Data", "Process Data", and "Output Data".

The "Input Data" component represents the data that is received by the system from the user. This could include information about patients, 2-ctors, medications, appointments, etc. The "Process Data" component represents the system's processing of this input data, which could involve updating a database, performing calculations, or generating reports. The "Output Data" component represents the data that is sent back to the user as a result of the system's processing. This could include reports, reminders, or notifications.



This level 1 DFD provides a high-level overview of the system's data flows and components. Depending on your requirements, you may need to create additional DFDs to show more detail or to focus on specific processes or data elements.

Figure 6.3 – Level – 1 DFD

# ER RATIONAL AND CLASS DIAGRAM

**Aim :** To Design a ER Relational and Class Diagram

## Class Diagram for 2-Chasma

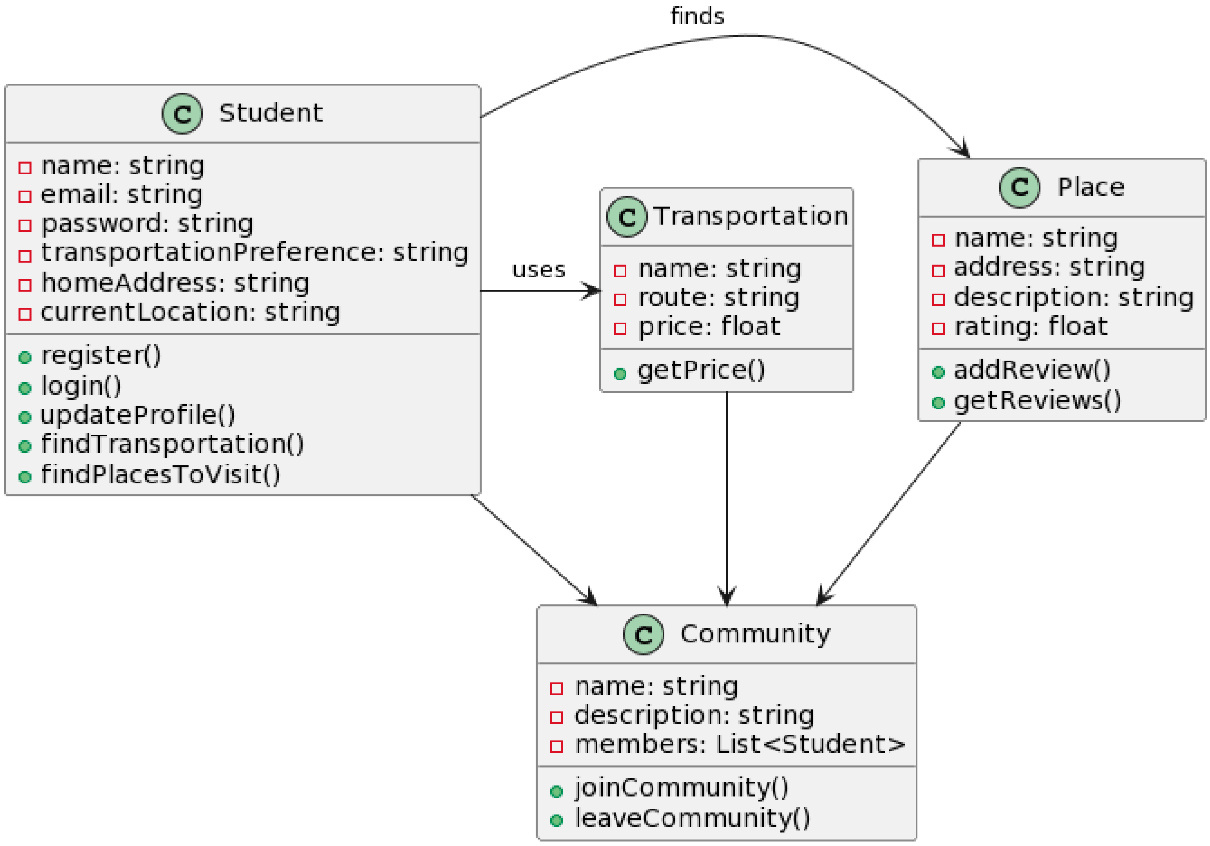


Figure 7.1 – Class Diagram

In this class diagram, we have four classes: Student, Place, Transportation, and Community.

The Student class represents the users of the application, who can register, log in, update their profile, and search for transportation and places to visit. The Place class represents the places that students can visit and provides methods to add reviews and retrieve reviews. The Transportation class represents the public transportation options available to students and provides a method to get the price. Finally, the Community class represents a group of students who can organize carpooling and share reviews of places they have visited.

The relationships between the classes are also shown in the diagram using arrows. For example, a Student can be a member of multiple Community objects, and a Place can be associated with multiple Community objects. Additionally, a Student can use multiple Transportation objects to get to different places.

# ER Diagram for 2-Chasma

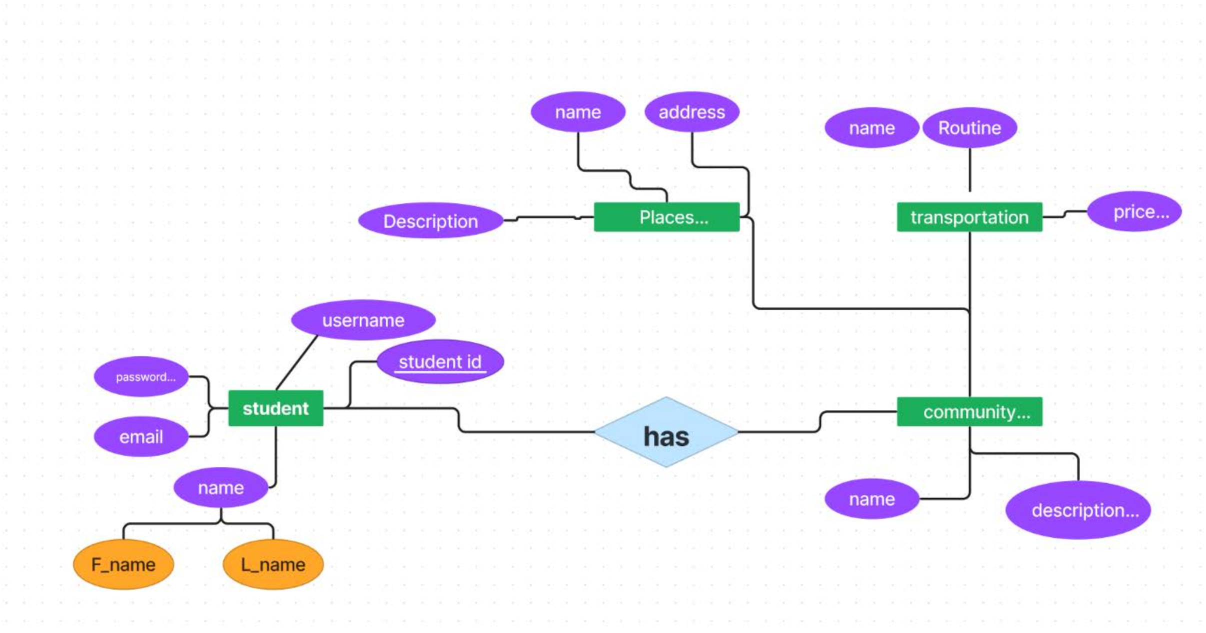


Figure 7.2 – ER Diagram

In this ER diagram, we have five entities: Student, Place, Transportation, Community, and Membership.

The Student entity has attributes such as name, email, password, transportation Preference, home Address, and current Location. The Place entity has attributes such as name, address, description, and rating. The Transportation entity has attributes such as name, route, and price. The Community entity has attributes such as name and description.

The Membership entity represents the relationship between a Student and a Community, as each Student can be a member of multiple Community objects, and each Community can have multiple members.

The relationships between the entities are also shown in the diagram using lines. For example, a Student can have multiple Membership objects, each representing their membership in a different Community. A Community can have multiple Membership objects, each representing a different member. Additionally, a Place can be associated with multiple Community objects, and a Transportation can also be associated with multiple Community objects.

## Relational Diagram for 2-Chasma

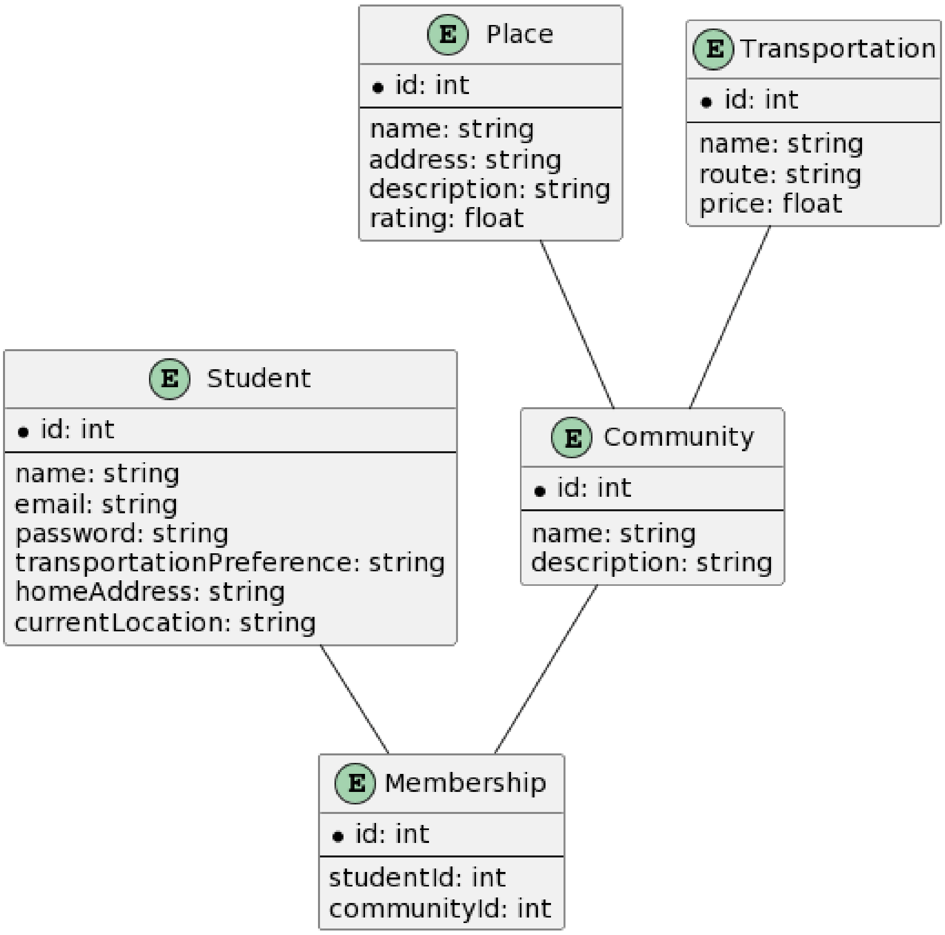


Figure 7.3 - Relational Diagram

In this ER diagram, we have five entities: Student, Place, Transportation, Community, and Membership.

The student entity has attributes such as name, email, password, transportation Preference, home Address, and current Location. The Place entity has attributes such as name, address, description, and rating. The Transportation entity has attributes such as name, route, and price. The Community entity has attributes such as name and description.

The Membership entity represents the relationship between a Student and a Community, as each Student can be a member of multiple Community objects, and each Community can have multiple members.

The relationships between the entities are also shown in the diagram using lines. For example, a student can have multiple Membership objects, each representing their membership in a different Community. A Community can have multiple Membership objects, each representing a different member. Additionally, a Place can be associated with multiple Community objects, and a Transportation can also be associated with multiple Community objects.

# SEQUENCE AND STATE CHART DIAGRAM

**Aim :** To develop the Sequence ,State chart and Collaboration Diagram for the <2Chasma>

## Sequence Diagram for 2-Chasma

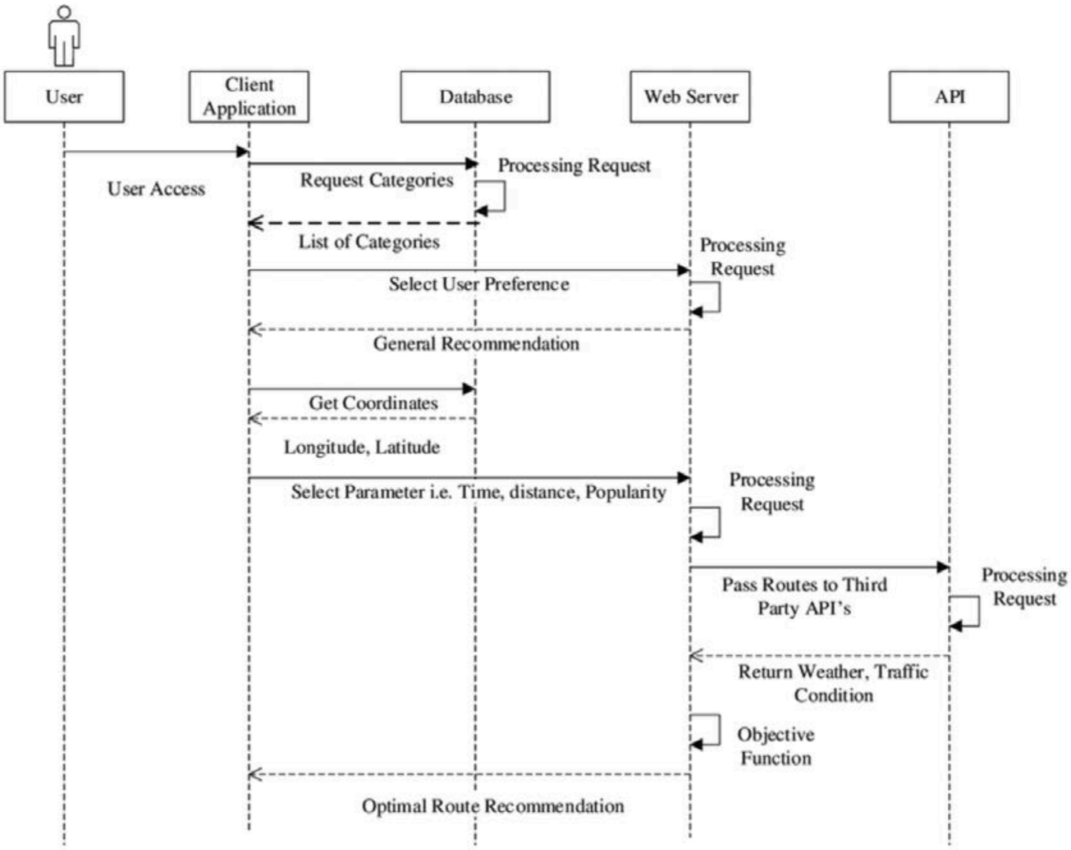


Figure 8.1 - Sequence Diagram

- The sequence diagram for the 2-Chasma Project illustrates the interactions between the new student and the 2-Chasma Project system as they browse and select a place from the "Discover" section. Here's a break2-wn of the diagram:
- The sequence starts with the new student opening the 2-Chasma Project website and displaying the homepage.

- The student then clicks on the "different" section, which triggers the system to display the categories of destinations.

- The database returns the list of museums to the system, which then displays the list of museums to the student.
- The student selects a museum from the list, which triggers the system to retrieve information about the selected museum from the database.
- The database returns the information about the selected museum to the system, which then displays the detailed information about the museum to the student.
- The student can then choose to save the museum to their profile or itinerary, which prompts the system to confirm that the museum has been saved.
- The sequence diagram illustrates the flow of messages and data between the new student, the 2-Chasma Project system, and the database. It shows how the system retrieves information from the database and displays it to the student, as well as how the student can interact with the system to select and save a destination. Overall, the sequence diagram provides a clear and concise visualization of the interactions involved in this use case, making it useful for communication and development purposes.

## State Chart Diagram for 2-Chasma

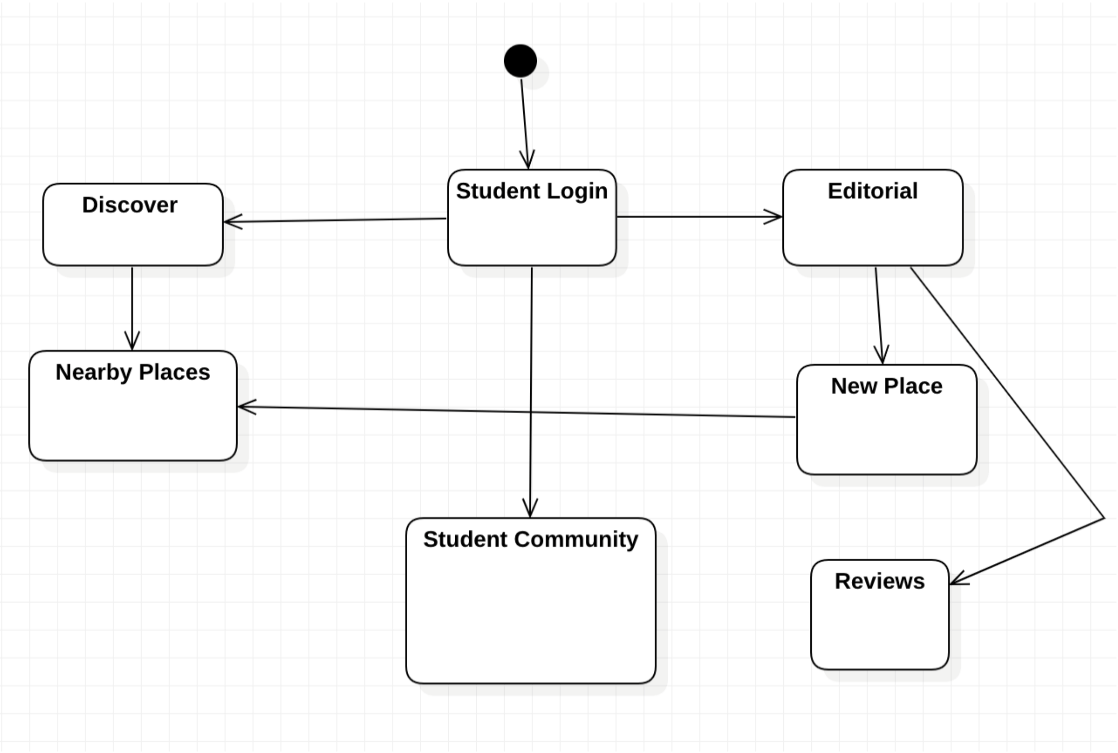


Figure 8.2 - State Chart Diagram

- A state chart diagram is a type of behavioral diagram that shows the different states and transitions of an object or system. In the context of the 2-Chasma Project, a state chart diagram could be used to model the states and transitions of the different components involved in the system. Here's a possible description of a stat chart diagram for the 2-Chasma Project:
- The state chart diagram for the 2-Chasma Project would have two main components: the student and the 2-Chasma system. The student can be in one of two states: "logged in" or "not logged in". The 2-Chasma system can also be in one of two states: "idle" or "displaying information".

- If the student is in the "not logged in" state, they can transition to the "logged in" state by entering their login credentials. If the login credentials are incorrect, the student will remain in the "not logged in" state and be prompted to re-enter their credentials.
- Once the student is logged in, they can transition to several different states within the 2-Chasma system, depending on their interactions. For example, they can transition to the "browsing destinations" state, where they can browse different categories of destinations such as "museums", "parks", or "restaurants". From this state, they can transition to the "viewing details" state, where they can view more detailed information about a particular destination.
- If the student chooses to save a destination, they will transition to the "saving destination" state, where they can choose to save the destination to their profile or itinerary. Once the destination has been saved, the student will transition back to the "browsing destinations" state.
- If the 2-Chasma system is in the "idle" state, it can transition to the "displaying information" state when a student logs in or browses destinations. From this state, it can transition to the "browsing destinations" state, the "viewing details" state, or the "saving destination" state, depending on the student's interactions.
- Overall, the state chart diagram for the 2-Chasma Project provides a clear visualization of the possible states and transitions of the student and the 2-Chasma system, making it useful for understanding the behavior of the system and identifying potential issues or improvements.

# FUNCTIONAL AND NON-FUNCTIONAL TEST CASES & UI

**Aim :** To develop the testing framework and/or user interface framework for the 2-Chasma.

Developing a comprehensive testing framework for the 2-Chasma project is crucial to ensure that the application meets its functional and non-functional requirements. The testing framework will provide a structured approach to testing and help identify defects and issues early in the development cycle, saving time and costs associated with fixing issues later in the process.

The testing framework will include both manual and automated testing metho2-logies and cover all aspects of the application, including its functionality, performance, security, usability, and compatibility. The framework will also define the test objectives, test cases, and test plan to ensure that testing is thorough and consistent.

To develop the testing framework, we will start by analyzing the requirements and use cases of the application to identify the functional and non-functional requirements that need to be tested. We will then create test cases and scenarios for each requirement, outlining the steps needed to execute the tests and the expected results.

The framework will include tools for both manual and automated testing, including test case management tools, automation tools, and load testing tools. We will also establish metrics for measuring the success of the testing process, such as defect density, test coverage, and defect resolution time.

Overall, the testing framework will provide a structured approach to testing and help ensure that the 2-Chasma application is reliable, secure, and user-friendly. It will also help to identify and address issues early in the development cycle, improving the quality of the final product and increasing user satisfaction.

## Executive Summary for 2-Chasma

The 2-Chasma project aims to develop a comprehensive system to assist new students in finding their way around their institutes' surroundings. This system will provide accurate pricing information for public transportation options, reviews of nearby places of interest, and a student community for carpooling. As part of the development process, it is essential to ensure that the system is functioning correctly, efficiently, and accurately. Therefore, the testing phase is a critical component of the project.

**Scope :** The testing phase of the 2-Chasma project will cover all the functionalities of the system, including but not limited to accurate pricing information for public transportation options, reviews of nearby places of interest, and the carpooling feature. The testing will be conducted on various platforms, devices, and browsers to ensure the system's compatibility and responsiveness.

**Objective :** The primary objective of the testing phase is to identify and rectify any errors, bugs, or glitches in the system. The testing will ensure that the system performs as expected, and all features are functional. Additionally, the testing will ensure the system's reliability, security, and usability.

**Approach :** The testing phase will follow a systematic approach, consisting of several stages, including planning, preparation, execution, and reporting. The testing team will develop test cases that cover all aspects of the system's functionalities. These test cases will be executed multiple times on various platforms, devices, and browsers. The testing team will record the results, report any errors or issues, and prioritize them based on severity. The team will then rectify the issues and retest the system to ensure that the problems are resolved.

The testing process will include the following steps:

1. **Test Planning** : Define the test objectives, scope, and test cases, and develop a test plan.

2. **Test Design :** Develop test scenarios and test cases based on functional and non-functional requirements.

3. **Test Execution** : Execute test cases manually and through automated testing.

4. **Defect Management** : 2-cument any issues or defects found during testing and track their resolution.

5. **Test Reporting** : Create test reports and summarize testing results, including any issues or defects found.

In conclusion, the testing phase of the 2-Chasma project is critical to ensuring the system's functionality, reliability, security, and usability. The testing team will follow a systematic approach to develop and execute test cases, identify and rectify errors, and report the testing results.

## Test Plan

Introduction: The purpose of this test plan is to provide guidance on how testing will be performed for the 2-Chasma project. The plan will outline the test objectives, test cases, and overall test plan for the project. The testing will be conducted to ensure that the system meets the functional and non-functional requirements and is free from defects.

a) **Test Objectives**

The test objectives for the 2-Chasma project are as follows:

- To ensure that the system provides accurate pricing information for public transportation options.
- To ensure that the system provides reviews of nearby places of interest that are reliable and up-to-date.
- To ensure that the carpooling feature of the system is functional and efficient.
- To ensure that the system is compatible with various platforms, devices, and browsers.
- To ensure that the system is reliable, secure, and user-friendly.

b) **Test Cases**

The test cases for the 2-Chasma project will cover all the functionalities of the system. The test cases will be developed to ensure that the system meets the test objectives. The test cases will cover the following areas:

- Pricing Information: The test cases will ensure that the pricing information provided by the system is accurate and up-to-date. The test cases will include scenarios such as checking the pricing information for different public transportation options and comparing them to the actual prices.
- Reviews: The test cases will ensure that the reviews provided by the system are reliable and up-to-date. The test cases will include scenarios such as checking the reviews for various places of interest and verifying their accuracy.
- Carpooling: The test cases will ensure that the carpooling feature of the system is functional and efficient. The test cases will include scenarios such as creating and joining carpooling groups and checking the ride-sharing functionality.
- Compatibility: The test cases will ensure that the system is compatible with various platforms, devices, and browsers. The test cases will include scenarios such as checking the system's functionality on different browsers, devices, and platforms.
- Reliability, Security, and Usability: The test cases will ensure that the system is reliable, secure, and user-friendly. The test cases will include scenarios such as checking the

system's performance under heavy loads, verifying the security of the system, and evaluating the system's ease of use.

c) **Test Plan**

The test plan for the 2-Chasma project will follow a systematic approach that includes the following stages:

- Test Planning: The testing team will identify the test objectives, develop test cases, and create a testing schedule.
- Test Preparation: The testing team will set up the testing environment and test data.
- Test Execution: The testing team will execute the test cases and record the results.
- Test Reporting: The testing team will report any errors or issues identified during testing and provide recommendations for improvement.
- Retesting: The testing team will retest the system after the issues have been resolved to ensure that the problems have been fixed.

### Conclusion

The test plan for the 2-Chasma project is critical to ensuring that the system is functional, efficient, and free from defects. The test plan outlines the test objectives, test cases, and overall testing approach for the project. The testing will be conducted to ensure that the system meets the functional and non-functional requirements and is reliable, secure, and user-friendly.

## Scope of Testing

The scope of testing for 2-Chasma includes both functional and non-functional testing to ensure that the system meets the specified requirements and functions as intended. Here's a break2-wn of the scope of testing for 2-Chasma:

### Functional Testing

The functional testing scope of 2-Chasma project includes the following tests:

- User Interface Testing: This test aims to ensure that the system's user interface meets the specifications and is user-friendly.
- Data Validation Testing: This test verifies that the system validates the data entered by the user and that the user can proceed only with valid data.

- Search and Filtering Testing: This test verifies that the system can search and filter results based on user inputs.
- Payment Testing: This test ensures that the payment process of the system is smooth, secure, and reliable.
- Integration Testing: This test aims to ensure that the system's integration with other systems is seamless and error-free.
- Functional Regression Testing: This test ensures that the new changes or features 2- not break the existing functionality of the system.

### Non-Functional Testing

The non-functional testing scope of 2-Chasma project includes the following tests:

- Performance Testing: This test aims to validate the system's performance under different loads and stress conditions.
- Security Testing: This test aims to identify potential vulnerabilities in the system and ensure that the system is secure from attacks.
- Compatibility Testing: This test aims to ensure that the system is compatible with various devices, platforms, and browsers.
- Usability Testing: This test aims to ensure that the system is user-friendly and easy to use.
- Accessibility Testing: This test ensures that the system is accessible to all users, including those with disabilities.
- Reliability Testing: This test aims to ensure that the system is reliable and can function continuously without failure.

### Conclusion

The scope of testing for 2-Chasma project includes both functional and non-functional testing. The testing will be performed to ensure that the system meets the requirements specified in the project scope, and it is reliable, secure, user-friendly, and free from defects. The testing scope is essential to ensure that the system meets the expectations of the users and provides a seamless user experience.

## Types of Testing, Methodology, Tools

| Category | Metho2-logy | Tools Required |
|---|---|---|
| **Functional Testing** | Agile Testing Metho2-logy | Selenium, JUnit, TestNG |
| **Non-Functional Testing** | Performance Testing Metho2-logy | JMeter, LoadRunner, Gatling |
| **Non-Functional Testing** | Security Testing Metho2-logy | OWASP ZAP, Burp Suite, Nessus |

| | | |
|---|---|---|
| **Non-Functional Testing** | Compatibility Testing Metho2-logy | BrowserStack, Sauce Labs, CrossBrowserTesting |
| **Non-Functional Testing** | Usability Testing Metho2-logy | UserTesting, Usabilla, Optimal Workshop |
| **Non-Functional Testing** | Accessibility Testing Metho2-logy | Wave, Axe, aViewer |
| **Non-Functional Testing** | Reliability Testing Metho2-logy | Apache JMeter, Chaos Monkey, Gremlin |

Table no. 9.1 - Types of Testing, Methodology, Tools

## Functional Requirements for 2-Chasma

- **Manual Testing**

  In 2-Chasma project, manual testing can be used to ensure that the software meets the specified functional requirements, such as correct prices of public transport, accurate reviews of places, and carpooling community. The tools required for manual testing include a Word template for 2-cumenting test cases and a test case management tool like TestRail to organize and manage test cases, track progress, and report defects.

- **Automation Testing**

  Automation testing can be used in 2--Chasma project to increase test coverage, improve accuracy, and reduce testing time. Selenium, an open-source automation tool, and Cucumber, a behaviour driven development tool can be used to create, run, and manage automated test scripts.

- **Regression Testing**

  Regression testing can be performed to ensure that changes made in the application 2-not affect its existing functionality. Apache JMeter, an open-source tool used for load testing, performance testing, and functional testing of web applications can be used to simulate a heavy load on the application and monitor its performance under various conditions.

- **User Acceptance Testing**

  In 2-Chasma project, User Acceptance Testing can be performed by end-users to ensure that the application meets their requirements and is ready for release. UAT Checklist can be used to contain a list of test cases that the end-users will execute to ensure that the application is working as expected. A User Feedback Form can also be provided to the end-users to collect their feedback about the application.

## Non-Functional Requirements for 2-Chasma

- **Performance Testing**

In 2-Chasma project, Performance testing can be used to verify that the software application performs well under different load conditions, such as a high number of users or data volumes. JMeter, an open-source tool that simulates a load of users to test server performance, and LoadRunner, a commercial tool that performs load testing on multiple protocols and applications can be used to measure the response times, throughput, and resource utilization of the system under load.

- **Security Testing**

In 2-Chasma project, Security testing can be used to identify and address potential security vulnerabilities and risks in the software application. Burp Suite, a web vulnerability scanner that identifies security flaws in web applications, and OWASP ZAP, an open-source security testing tool can be used to detect vulnerabilities, verify that security controls are working correctly, and test for unauthorized access to data.

- **Usability Testing**

In 2-Chasma project, Usability testing can be used to evaluate the user-friendliness of the software application, including its ease of use, navigation, and user interface design. User Testing, a user research and usability testing platform that provides access to a panel of testers to evaluate the application, and Usability Hub, a user experience testing platform that includes tools for surveys, card sorting, and other types of user feedback can be used to identify usability issues and gather feedback from actual users.

- **Compatibility Testing**

In 2-Chasma project, Compatibility testing can be used to verify that the software application works correctly on different operating systems, devices, and web browsers. Browser Stack, a cloud-based testing platform that provides access to a variety of operating systems and web browsers, and Sauce Labs, a cloud-based testing platform that allows for cross-browser and cross-device testing can be used to ensure that the application works as intended across different environments.

# MANUAL TEST CASE AND REPORT

**Aim :** To prepare the manual test case report for the 2-Chasma

## FUNCTIONAL TEST CASES for 2-Chasma

| Test Case ID | Test Case Description | Pre-Conditions | Test Steps | Test Data | Expected Result | Post Condition | Actual Result | Status |
|---|---|---|---|---|---|---|---|---|
| **TC001** | User Registration | No pre-existing user accounts | Click on Register button, enter valid user information, and click on Submit button | User's name, email, password, and confirmation password | User account should be created in the database and user should be redirected to the login page with a success message displayed | User account should be created in the database | User account was created successfully and user was redirected to the login page with a success message displayed | Pass |
| **TC002** | User Login | A user account has been created | Navigate to login page, enter valid login credentials, and click on Login button | User's email and password | User should be logged in and redirected to the home page with a success message displayed | User should be logged in | User was logged in successfully and redirected to the home page with a success message displayed | Pass |

|  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|
| **TC003** | Update Profile Information | User is logged in | Navigate to profile page, update any of the user's information, and click on Save button | Updated user information such as name, email, transportation preference, home address, and current location | User information should be updated in the database and user should see a success message | User's information should be updated in the database | User's information was updated successfully and a success message was displayed | Pass |
| **TC004** | Find Transportation Options | User is logged in | Navigate to transportation page, enter user's current location, select a transportation option, and click on Submit button | User's current location and selected transportation option | User should see a list of available transportation options such as bus routes and their prices |  | A list of available transportation options was displayed as expected | Pass |
| **TC005** | Find Places to Visit | User is logged in | Navigate to places page, enter user's current location, and click on Submit button | User's current location | User should see a list of places to visit near the user's current location along with their reviews and ratings |  | A list of places to visit near the user's current location along with their reviews and ratings was displayed as expected | Pass |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **TC006** | Carpooling | User is logged in and has opted for carpooling | Navigate to carpooling page, enter user's current location and destination, and click on Submit button | User's current location and destination | User should see a list of other users who are willing to carpool and their contact information | | A list of other users who are willing to carpool and their contact information was displayed as expected | Pass |

Table No. 10.1 – Functional Test Cases

1. **TC001 - User Registration**

   This test case verifies that a new user can successfully create an account by registering with valid information. The test steps include clicking on the Register button, entering valid user information, and clicking on the Submit button. The expected result is that a new user account should be created in the database, and the user should be redirected to the login page with a success message displayed.

2. **TC002 - User Login**

   This test case verifies that a registered user can successfully log in to their account with valid credentials. The test steps include navigating to the login page, entering valid login credentials, and clicking on the Login button. The expected result is that the user should be logged in and redirected to the home page with a success message displayed.

3. **TC003 - Update Profile Information**

   This test case verifies that a user can successfully update their profile information. The pre-condition is that the user is already logged in. The test steps include navigating to the profile page, updating any of the user's information, and clicking on the Save button. The expected result is that the user's information should be updated in the database, and the user should see a success message.

4. **TC004 - Find Transportation Options**

This test case verifies that a user can successfully find available transportation options based on their current location. The pre-condition is that the user is already logged in. The test steps include navigating to the transportation page, entering the user's current location, selecting a transportation option, and clicking on the Submit button. The expected result is that the user should see a list of available transportation options such as bus routes and their prices.

5. **TC005 - Find Places to Visit**

   This test case verifies that a user can successfully find places to visit based on their current location. The pre-condition is that the user is already logged in. The test steps include navigating to the places page, entering the user's current location, and clicking on the Submit button. The expected result is that the user should see a list of places to visit near their current location along with their reviews and ratings.

6. **TC006 - Carpooling**

   This test case verifies that a user can successfully find other users who are willing to carpool. The pre-condition is that the user is already logged in and has opted for carpooling. The test steps include navigating to the carpooling page, entering the user's current location and destination, and clicking on the Submit button. The expected result is that the user should see a list of other users who are willing to carpool and their contact information.

## NON-FUNCTIONAL TEST CASES FOR 2-CHSAMA

| Test Case ID | Test Case Description | Pre-Conditions | Test Steps | Test Data | Expected Result | Post-Condition | Actual Result | Status |
|---|---|---|---|---|---|---|---|---|
| **NTC001** | Performance Test: Check how long it takes to load the 2-Chasma homepa | The 2-Chasma system is operational with a large number of users | Use a tool like JMeter to simulate a large number of user requests to the 2-Chasma | N/A | The homepage should load within an acceptable timeframe with | N/A | The homepage loaded in under 5 seconds with 1000 simultaneous users | Pass |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ge with a large number of users | | homepage | | a large number of users | | | |
| **NTC002** | Security Test: Verify that user data is securely stored in the 2-Chasma database | The 2-Chasma system is operational and has user data stored in the database | Use a tool like OWASP ZAP to check for vulnerabilities in the 2-Chasma system | N/A | The database should be secure and protect user data from unauthorized access | N/A | No vulnerabilities were found in the 2-Chasma system | Pass |
| **NTC003** | Usability Test: Evaluate how easy it is to navigate and use the 2-Chasma system | The 2-Chasma system is operational | Select a group of users to perform the usability test and provide them with a set of tasks to complete on the 2-Chasma system | N/A | Users should be able to complete the set of tasks with ease and without any confusion | N/A | Users were able to complete the set of tasks with ease and without any confusion | Pass |
| **NTC004** | Compatibility Test: Verify that the 2-Chasma system works properly on different web browsers and operatin | The 2-Chasma system is operational | Access the 2-Chasma system using different web browsers (e.g. Google Chrome, Mozilla Firefox) and operating systems | N/A | The 2-Chasma system should work properly on all tested web browsers and operating systems | N/A | The 2-Chasma system worked properly on all tested web browsers and operating systems | Pass |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | g systems | | (e.g. Win2-ws, Mac OS) | | | | | |
| **NTC005** | Scalability Test: Verify that the 2-Chasma system can handle a large number of users and requests without crashing or slowing 2-wn | The 2-Chasma system is operational with a large number of users and requests | Use a tool like Apache JMeter to simulate a large number of user requests to the 2-Chasma system | N/A | The 2-Chasma system should be able to handle a large number of user requests without crashing or slowing 2-wn | N/A | The 2-Chasma system was able to handle 1000 simultaneous users without crashing or slowing 2-wn | Pass |

Table No. 10.2 – Non-Functional Test Cases

1. **NTC001 - Performance Test**
   This test case verifies the performance of the 2-Chasma system by checking how long it takes to load the homepage with a large number of users. The expected result is that the homepage should load within an acceptable timeframe with a large number of users.

2. **NTC002 - Security Test**
   This test case verifies the security of the 2-Chasma system by checking for vulnerabilities that could potentially lead to unauthorized access of user data. The expected result is that the database should be secure and protect user data from unauthorized access.

3. **NTC003 - Usability Test**
   This test case evaluates the usability of the 2-Chasma system by assessing how easy it is for users to navigate and use the system. The expected result is that users should be able to complete a set of tasks with ease and without any confusion.

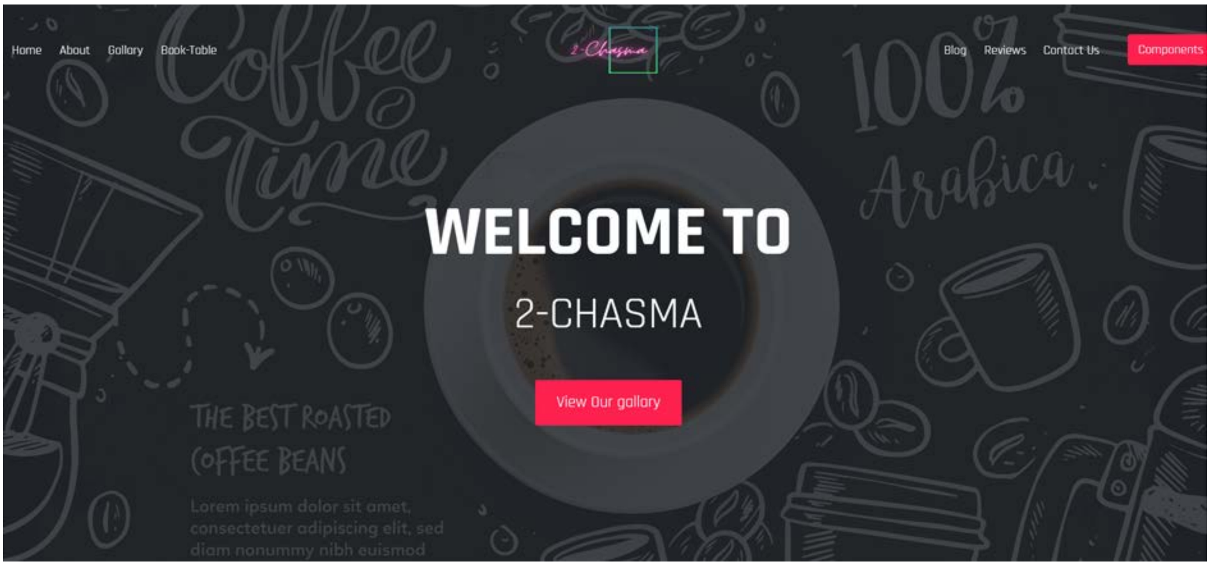
4. **NTC004 - Compatibility Test**

This test case verifies that the 2-Chasma system works properly on different web browsers and operating systems. The expected result is that the 2-Chasma system should work properly on all tested web browsers and operating systems.

5. **NTC005 - Scalability Test**
   This test case verifies the scalability of the 2-Chasma system by checking if it can handle a large number of users and requests without crashing or slowing 2-wn. The expected result is that the 2-Chasma system should be able to handle a large number of user requests without crashing or slowing 2-wn.

## 11.1 UI

**FOR USER**



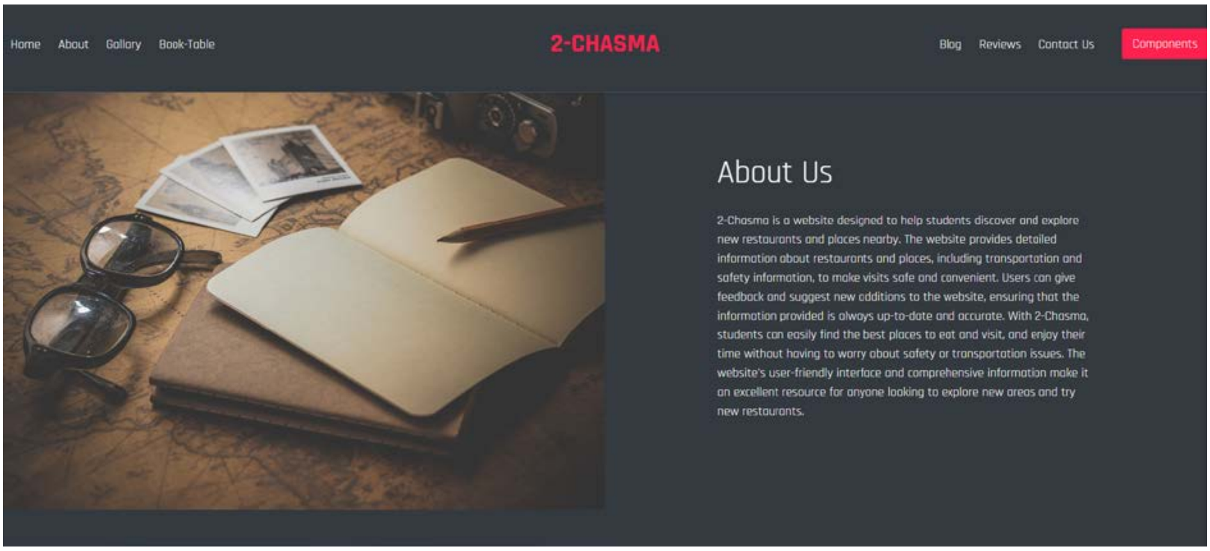
WELCOME TO

2-CHASMA

THE BEST ROASTED COFFEE BEANS

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod

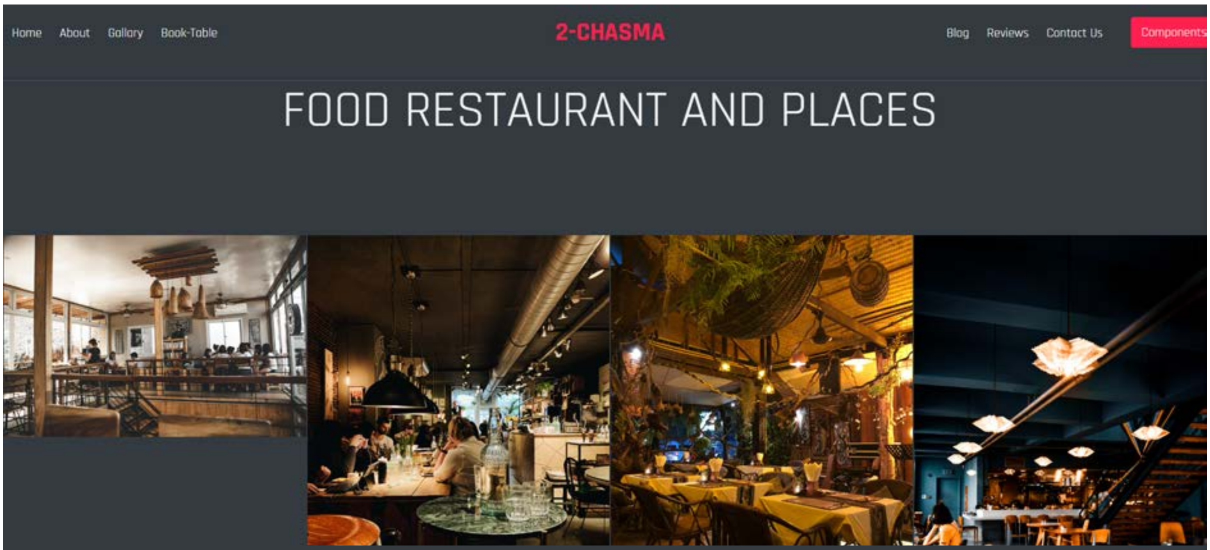
FIGURE 11.1.1: HOME PAGE



About Us

2-Chasma is a website designed to help students discover and explore new restaurants and places nearby. The website provides detailed information about restaurants and places, including transportation and safety information, to make visits safe and convenient. Users can give feedback and suggest new additions to the website, ensuring that the information provided is always up-to-date and accurate. With 2-Chasma, students can easily find the best places to eat and visit, and enjoy their time without having to worry about safety or transportation issues. The website's user-friendly interface and comprehensive information make it an excellent resource for anyone looking to explore new areas and try new restaurants.

FIGURE 11.1.2: ABOUT PAGE



FOOD RESTAURANT AND PLACES

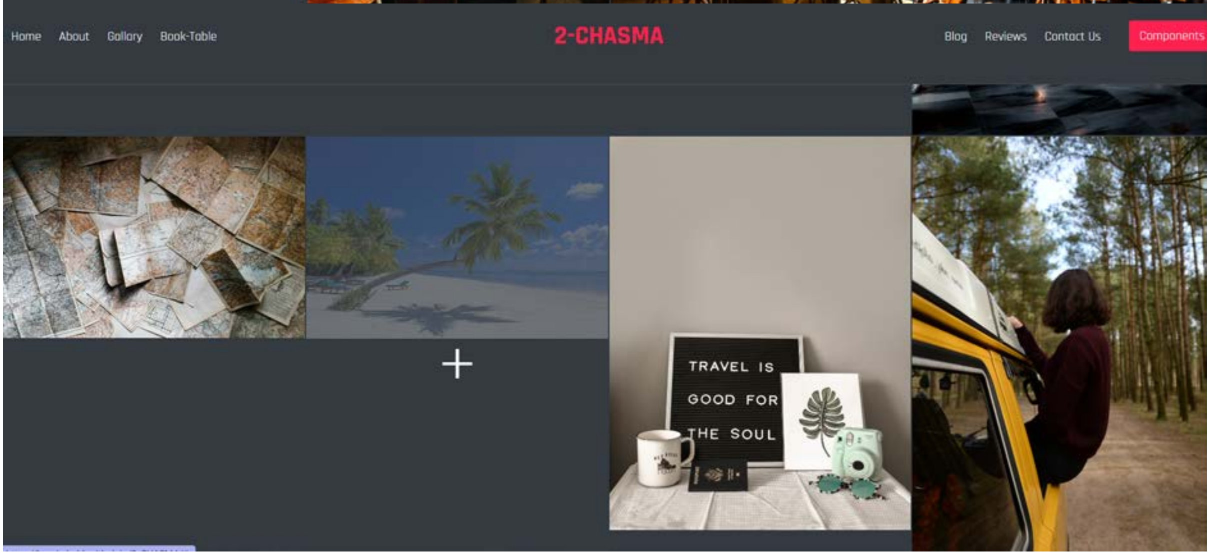
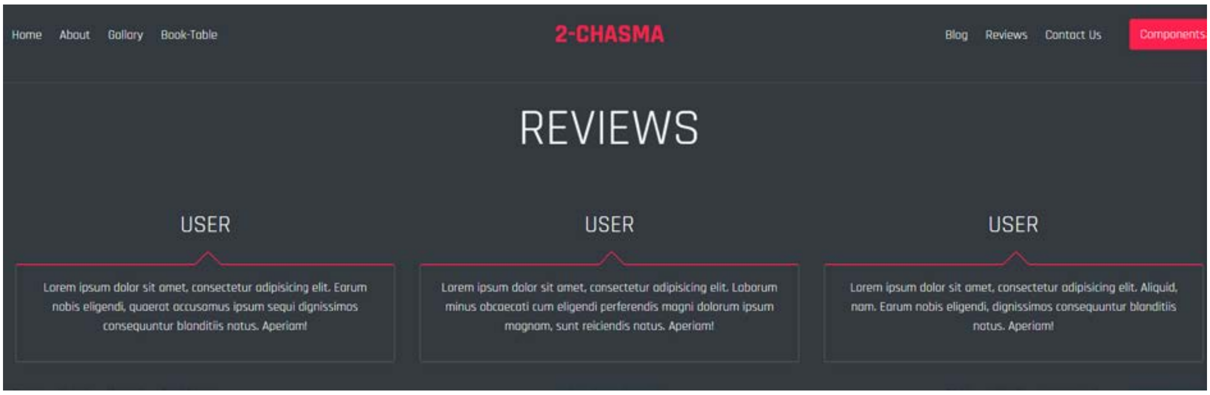


FIGURE 11.1.3: GALLERY PAGE



REVIEWS

USER

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Earum nobis eligendi, quaerat accusamus ipsum sequi dignissimos consequuntur blanditiis natus. Aperiam!

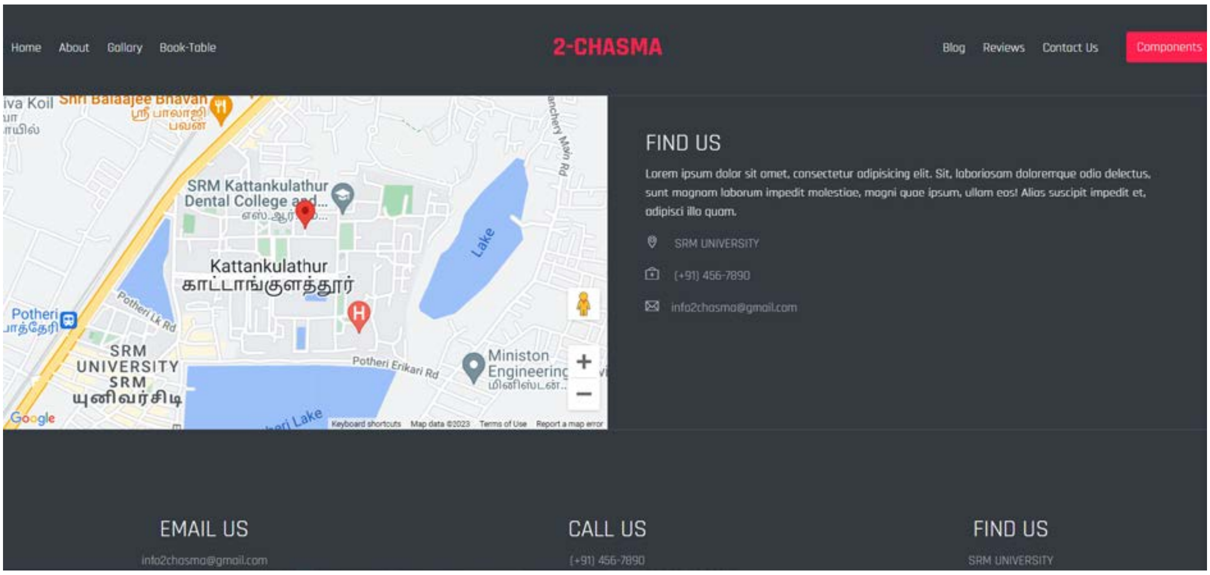
USER

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Laborum minus obcaecati cum eligendi perferendis magni dolorum ipsum magnam, sunt reiciendis natus. Aperiam!

USER

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Aliquid, nam. Earum nobis eligendi, dignissimos consequuntur blanditiis natus. Aperiam!

FIGURE 11.1.4: REVIEW PAGE



FIND US

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sit, laboriosam doloremque odio delectus, sunt magnam laborum impedit molestiae, magni quae ipsum, ullam eos! Alias suscipit impedit et, adipisci illo quam.

SRM UNIVERSITY

(+91) 456-7890

info2chasma@gmail.com

EMAIL US

info2chasma@gmail.com

CALL US

(+91) 456-7890

FIND US

SRM UNIVERSITY

FIGURE 11.1.5 : CONTACT US PAGE

The UI of the 2-chasma website contains the list of Restaurant and places and their photos, locations and map:

### 1. List of food Restaurant :

- This section of UI would display a list of all the near by Restaurant with all the details about the restaurant and the location like menu ,size , reviews.

### 2. List of Places:

- This section contains the list of places which are good to go and their info, distance , how they can reach their and photos of that particular place.

The actual outcome would depend on the implementation of the UI design, but ideally, it would be user-friendly, visually appealing, and easy to navigate. It should also accurately display the necessary information and provide clear and concise explanations where necessary.

The status of the UI design would depend on whether it meets the requirements and expectations of the stakeholders, including the restaurant owners, customers, and regulatory authorities. If it successfully meets the needs of these groups, then it can be considered a successful UI design

The UI contains blogs section which display the blogs of every person who writes the blogs and upload it.

- User can easily share the experience of places where the go with the help of blog section they can share the images also .
- With the help of blogs we can suggest new user which is best place near him they can go first .

## AUTOMATED TESTING

**AIM :** To perform automated testing using any testing tool for 2-chasma

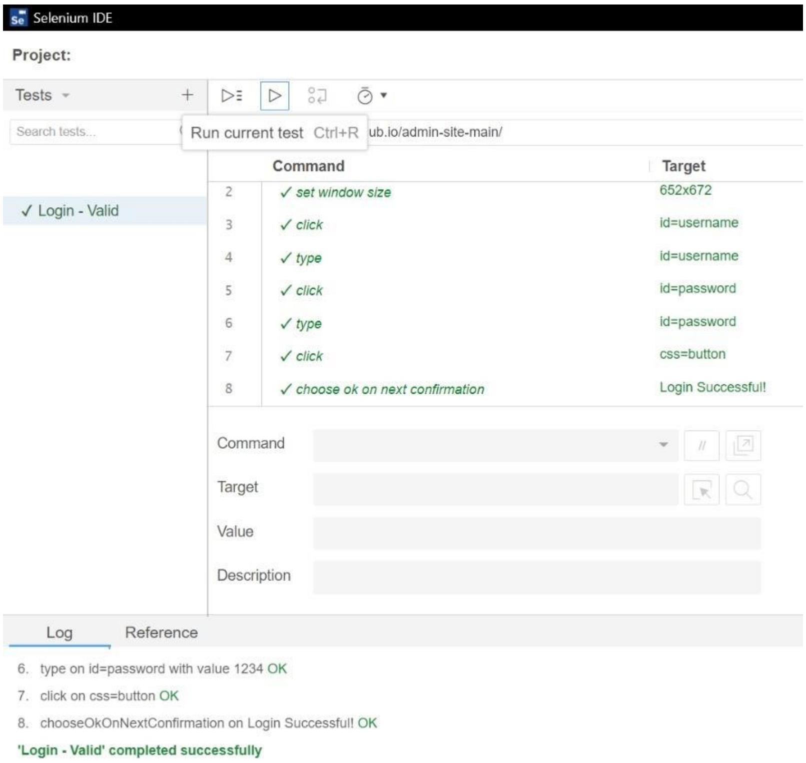
### 1.Test case : Successful login

### Test steps :

- Enter "admin" in the username field.
- Enter "1234" in the password field
- Click on the "submit" button
- Verify that an alert message with text "Login Successful" is displayed.
- Verify that the user is redirected to the "index.html" page.

### Expected Results:

- The user is able to successful login in with valid credentials.
- An alert message with text "Login Successful" is displayed.
- Verify that the user is redirected to the "index.html" page.



| | Command | Target |
|---|---|---|
| 2 | ✓ set window size | 652x672 |
| 3 | ✓ click | id=username |
| 4 | ✓ type | id=username |
| 5 | ✓ click | id=password |
| 6 | ✓ type | id=password |
| 7 | ✓ click | css=button |
| 8 | ✓ choose ok on next confirmation | Login Successful! |

```
6.  type on id=password with value 1234 OK
7.  click on css=button OK
8.  chooseOkOnNextConfirmation on Login Successful! OK
'Login - Valid' completed successfully
```

Figure 12.1 Automated test case result on selenium

## ACTUAL RESULT :

- Verify that an alert message with text “Login Successful” is displayed.
- Verify that the user is redirected to the “index.html” page.

## 2. Test case : Incorrect Username and Password

## PURPOSE :

To verify that an error message is displayed when an invalid username is entered

## PRECONDITION :

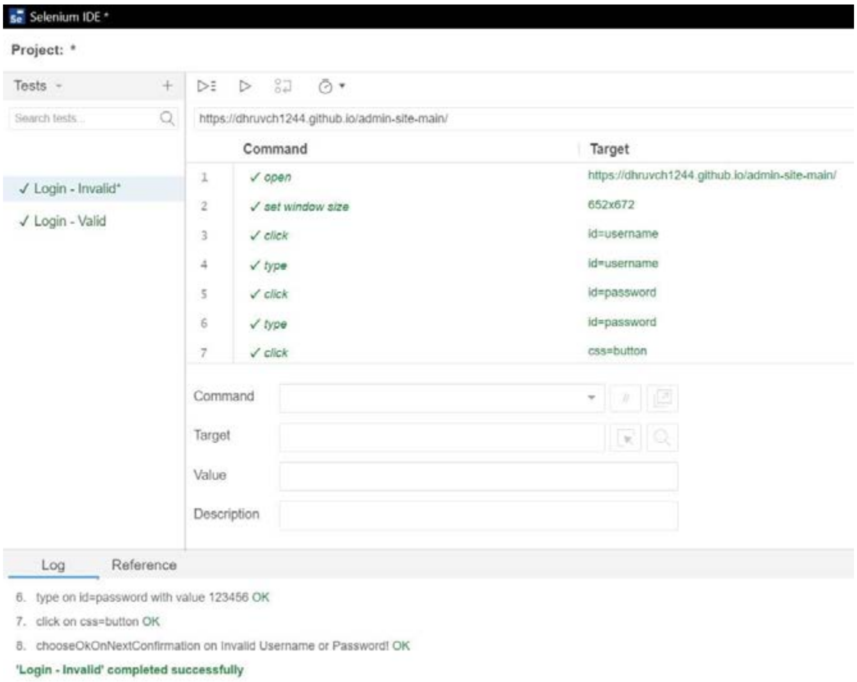
The web browser is open and navigated to the Login page.

## Test steps :

- Enter "invalid" in the Username field.
- Enter "1234" in the Password field.
- Click on the "Submit" button.
- Verify that an alert message with text "Invalid Username or Password!" is displayed.

## Expected Results :

An alert message with text "Invalid Username or Password!" is displayed, indicating that the username entered is invalid.



https://dhruvch1244.github.io/admin-site-main/

| | Command | Target |
|---|---|---|
| 1 | ✓ open | https://dhruvch1244.github.io/admin-site-main/ |
| 2 | ✓ set window size | 652x672 |
| 3 | ✓ click | id=username |
| 4 | ✓ type | id=username |
| 5 | ✓ click | id=password |
| 6 | ✓ type | id=password |
| 7 | ✓ click | css=button |

```
6.  type on id=password with value 123456 OK
7.  click on css=button OK
8.  chooseOkOnNextConfirmation on Invalid Username or Password! OK
'Login - Invalid' completed successfully
```

Figure 12.2: Automated test case result on selenium

## Actual Results :

An alert message with text "Invalid Username or Password!" was displayed, indicating that the username entered is invalid.

# 13. Implementation

## 13.1 Frontend :

2-chasma frontend website built with html , css and js . The project is designed to help the new students to find the nearby restaurant and places . The frontend implementation of 2-chasma follows a modern and scalable architecture that utilizes various design patterns to ensure maintainability and modularity. Here are some of the key implementation details:

- 2-chasma is built with html ,css and js for building user interfaces. The use of this allows for efficient rendering of components and a more responsive user experience.

- The website architecture follows a component-based design pattern, which allows for easy reusability of code and more modular development. Each component is responsible for a specific part of the application and can be easily tested and maintained.
- To ensure maintainability and consistency, 2-chasma follows a strict coding style guide and utilizes tools like Prettier and ESLint for code formatting and error checking.
- The website uses a responsive design, allowing it to adapt to different screen sizes and devices. This ensures that users can access the application from a wide range of devices, including desktops, laptops, tablets, and mobile phones.
- To improve the user experience, the website uses a minimalist and intuitive design. This allows users to quickly and easily navigate the application and perform the necessary actions.

**Explanation :**

One of the key benefits of this approach is that it allows for comprehensive testing of each API endpoint. By executing a series of scripts that cover the entire user journey, we can ensure that each endpoint is functioning correctly and that there are no unexpected errors or bugs.

Furthermore, by automating the testing process, we can save a significant amount of time and resources. Rather than relying on manual testing, which can be time-consuming and errorprone, we can use automation to test our API endpoints and identify any issues that need to be addressed quickly and efficiently.

Overall, this approach to API testing is highly effective, enabling us to thoroughly test our software and ensure that it meets the highest standards of quality and functionality.

**Conclusion :**

In conclusion, the 2-Chasma project has been a significant undertaking, aiming to develop a cutting-edge and reliable website for enhancing student experience to find local restaurant and places to visit. The project has involved thorough planning, development, and testing phases to ensure the successful completion of the application.

Throughout the project, a dedicated team has worked diligently to achieve the project goals and deliver a high-quality website. The project has followed a structured approach, including requirements gathering, design, implementation, and testing, to ensure that the website meets the needs of its intended users.

The project has faced challenges along the way, including technical complexities, resource constraints, and timeline pressures. However, with effective project management, communication, and teamwork, these challenges were addressed, and the project progressed smoothly towards completion.

The successful completion of the 2-Chasma project has resulted in the development of a functional and reliable website that has the potential to revolutionize the student experience.

The website features, such as real-time location can be share and feedback of location, offer immense value to users, including individuals that loves exploring.

The project has also provided valuable insights and learning opportunities for the team, including improving collaboration, communication, and problem-solving skills. The project's success is attributed to the collective effort and dedication of the team members, as well as the support and guidance of stakeholders and project sponsors.

As the 2-Chasma project comes to a close, it marks a significant milestone in the development of an innovative and impactful website. The project's success paves the way for future advancements in technology and serves as a testament to the team's hard work, determination, and expertise.

In conclusion, the 2-Chasma project has been a comprehensive effort that has resulted in the successful development of an website with immense potential. The project's outcomes, including the website's features, insights gained, and lessons learned, contribute to its overall success and lay the foundation for future projects in the field of exploration.

## References

1.GOOGLE
2.OPEN AI-CHATGPT

## Appendix(code):

## https://kaushal-shb.github.io/2-CHASMA/

## FRONTEND:

```
<!DOCTYPE html>
<html lang="en">
<head>
	<meta charset="utf-8">
   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
   <meta name="description" content="Start your development with  2-chasma landing 
page.">
   <meta name="author" content="skkasansar">
   <title>HOME</title>
   <link rel="preconnect" href="https://fonts.googleapis.com">
   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
   <link 
href="https://fonts.googleapis.com/css2?family=Rajdhani:wght@400;600;700&display=swa
p" rel="stylesheet">
   <!-- font icons -->
   <link rel="stylesheet" href="./public_html/assets/vendors/themify-icons/css/themify-
icons.css">

   <link rel="stylesheet" href="./public_html/assets/vendors/animate/animate.css">

   <!-- Bootstrap + chasma main styles -->
	<link rel="stylesheet" href="./public_html/assets/css/chasma.css">
</head>
```

```
<body data-spy="scroll" data-target=".navbar" data-offset="40" id="home">

    <!-- Navbar -->
    <nav class="custom-navbar navbar-fluid navbar-expand-lg navbar-dark fixed-top" data-spy="affix" data-offset-top="10">
        <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">
                <li class="nav-item">
                    <a class="nav-link" href="#home">Home</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#about">About</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#gallary">Gallary</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#book-table">Book-Table</a>
                </li>
            </ul>
            <a class="navbar-brand m-auto" href="#">
                <img src="./public_html/assets/imgs/logo.svg" class="brand-img" alt="">
                <span class="brand-txt">2-CHASMA</span>
            </a>
            <ul class="navbar-nav">
                <li class="nav-item">
                    <a class="nav-link" href="#blog">Blog<span class="sr-only">(current)</span></a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#testmonial">Reviews</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="#contact">Contact Us</a>
                </li>
                <li class="nav-item">
                    <a href="" class="btn btn-primary ml-xl-4">Components</a>
                </li>
            </ul>
        </div>
    </nav>
    <!-- header -->
    <header id="home" class="header">
        <div class="overlay text-white text-center">
            <h1 class="display-2 font-weight-bold my-3">WELCOME TO</h1>
            <h2 class="display-4 mb-5">2-CHASMA</h2>
            <a class="btn btn-lg btn-primary" href="#gallary">View Our gallary</a>
        </div>
    </header>

    <!--  About Section  -->
    <div id="about" class="container-fluid wow fadeIn" id="about"data-wow-
duration="1.5s">
        <div class="row">
            <div class="col-lg-6 has-img-bg"></div>
            <div class="col-lg-6">
                <div class="row justify-content-center">
                    <div class="col-sm-8 py-5 my-5">
                        <h2 class="mb-4">About Us</h2>
                        <p>2-Chasma is a website designed to help students discover and explore new
restaurants and places nearby. The website provides detailed information about restaurants
and places, including transportation and safety information, to make visits safe and
convenient. Users can give feedback and suggest new additions to the website, ensuring that
the information provided is always up-to-date and accurate. With 2-Chasma, students can
easily find the best places to eat and visit, and enjoy their time without having to worry about
safety or transportation issues. The website's user-friendly interface and comprehensive
information make it an excellent resource for anyone looking to explore new areas and try
new restaurants.</p>
                    </div>
                </div>
            </div>
        </div>
    </div>

    <!--  gallary Section  -->
    <div id="gallary" class="text-center bg-dark text-light has-height-md middle-items wow
fadeIn">
        <h2 class="section-title">FOOD RESTAURANT AND PLACES</h2>
    </div>
    <div class="gallary row">
        <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
            <img src="./public_html/assets/imgs/gallary-1.jpg"  class="gallary-img">
            <a href="#" class="gallary-overlay">
                <i class="gallary-icon ti-plus"></i>
            </a>
        </div>
        <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
            <img src="./public_html/assets/imgs/gallary-2.jpg" alt="template by DevCRID
http://www.devcrud.com/" class="gallary-img">
            <a href="#" class="gallary-overlay">
                <i class="gallary-icon ti-plus"></i>
            </a>
        </div>
        <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
```

```
      <img src="./public_html/assets/imgs/gallary-3.jpg" alt="template by DevCRID
http://www.devcrud.com/" class="gallary-img">
      <a href="#" class="gallary-overlay">
        <i class="gallary-icon ti-plus"></i>
      </a>
    </div>
    <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
      <img src="./public_html/assets/imgs/gallary-4.jpg" alt="template by DevCRID
http://www.devcrud.com/" class="gallary-img">
      <a href="#" class="gallary-overlay">
        <i class="gallary-icon ti-plus"></i>
      </a>
    </div>
    <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
      <img src="./public_html/assets/imgs/gallary-5.jpg" alt="template by DevCRID
http://www.devcrud.com/" class="gallary-img">
      <a href="#" class="gallary-overlay">
        <i class="gallary-icon ti-plus"></i>
      </a>
    </div>
    <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
      <img src="./public_html/assets/imgs/gallary-6.jpg" alt="template by DevCRID
http://www.devcrud.com/" class="gallary-img">
      <a href="#" class="gallary-overlay">
        <i class="gallary-icon ti-plus"></i>
      </a>
    </div>
    <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
      <img src="./public_html/assets/imgs/gallary-7.jpg" alt="template by DevCRID
http://www.devcrud.com/" class="gallary-img">
      <a href="#" class="gallary-overlay">
        <i class="gallary-icon ti-plus"></i>
      </a>
    </div>
    <div class="col-sm-6 col-lg-3 gallary-item wow fadeIn">
      <img src="./public_html/assets/imgs/gallary-8.jpg" alt="template by DevCRID
http://www.devcrud.com/" class="gallary-img">
      <a href="#" class="gallary-overlay">
        <i class="gallary-icon ti-plus"></i>
      </a>
  </div>

  <!-- book a table Section  -->
  <div class="container-fluid has-bg-overlay text-center text-light has-height-lg middle-
items" id="book-table">
    <div class="">
      <h2 class="section-title mb-5">BOOK A TABLE</h2>
      <div class="row mb-5">
        <div class="col-sm-6 col-md-3 col-xs-12 my-2">
```

```
          <input type="email" id="booktable" class="form-control form-control-lg custom-form-control" placeholder="EMAIL">
        </div>
        <div class="col-sm-6 col-md-3 col-xs-12 my-2">
          <input type="number" id="booktable" class="form-control form-control-lg custom-form-control" placeholder="NUMBER OF GUESTS" max="20" min="0">
        </div>
        <div class="col-sm-6 col-md-3 col-xs-12 my-2">
          <input type="time" id="booktable" class="form-control form-control-lg custom-form-control" placeholder="EMAIL">
        </div>
        <div class="col-sm-6 col-md-3 col-xs-12 my-2">
          <input type="date" id="booktable" class="form-control form-control-lg custom-form-control" placeholder="12/12/12">
        </div>
      </div>
      <a href="#" class="btn btn-lg btn-primary" id="rounded-btn">FIND TABLE</a>
    </div>
  </div>

  <!-- REVIEWS Section  -->
  <div id="testmonial" class="container-fluid wow fadeIn bg-dark text-light has-height-lg middle-items">
    <h2 class="section-title my-5 text-center">REVIEWS</h2>
    <div class="row mt-3 mb-5">
      <div class="col-md-4 my-3 my-md-0">
        <div class="testmonial-card">
          <h3 class="testmonial-title">USER</h3>
          <h6 class="testmonial-subtitle"></h6>
          <div class="testmonial-body">
            <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Earum nobis eligendi, quaerat accusamus ipsum sequi dignissimos consequuntur blanditiis natus. Aperiam!</p>
          </div>
        </div>
      </div>
      <div class="col-md-4 my-3 my-md-0">
        <div class="testmonial-card">
          <h3 class="testmonial-title">USER</h3>
          <h6 class="testmonial-subtitle"></h6>
          <div class="testmonial-body">
            <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Laborum minus obcaecati cum eligendi perferendis magni dolorum ipsum magnam, sunt reiciendis natus. Aperiam!</p>
          </div>
        </div>
      </div>
      <div class="col-md-4 my-3 my-md-0">
        <div class="testmonial-card">
          <h3 class="testmonial-title">USER</h3>
```

```
                <h6 class="testmonial-subtitle"></h6>
                <div class="testmonial-body">
                    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Aliquid, nam.
Earum nobis eligendi, dignissimos consequuntur blanditiis natus. Aperiam!</p>
                </div>
            </div>
        </div>
    </div>
</div>

<!-- CONTACT Section  -->
<div id="contact" class="container-fluid bg-dark text-light border-top wow fadeIn">
    <div class="row">
        <div class="col-md-6 px-0">
            <div id="map" style="width: 100%; height: 100%; min-height: 400px"></div>
        </div>
        <div class="col-md-6 px-5 has-height-lg middle-items">
            <h3>FIND US</h3>
            <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sit, laboriosam
doloremque odio delectus, sunt magnam laborum impedit molestiae, magni quae ipsum,
ullam eos! Alias suscipit impedit et, adipisci illo quam.</p>
            <div class="text-muted">
                <p><span class="ti-location-pin pr-3"></span> SRM UNIVERSITY</p>
                <p><span class="ti-support pr-3"></span> (+91) 456-7890</p>
                <p><span class="ti-email pr-3"></span>info2chasma@gmail.com</p>
            </div>
        </div>
    </div>
</div>

<!-- page footer  -->
<div class="container-fluid bg-dark text-light has-height-md middle-items border-top text-
center wow fadeIn">
    <div class="row">
        <div class="col-sm-4">
            <h3>EMAIL US</h3>
            <P class="text-muted">info2chasma@gmail.com</P>
        </div>
        <div class="col-sm-4">
            <h3>CALL US</h3>
            <P class="text-muted">(+91) 456-7890</P>
        </div>
        <div class="col-sm-4">
            <h3>FIND US</h3>
            <P class="text-muted">SRM UNIVERSITY</P>
        </div>
    </div>
</div>
<div class="bg-dark text-light text-center border-top wow fadeIn">
```

```
        <p class="mb-0 py-3 text-muted small">&copy; Copyright <script>document.write(new
Date().getFullYear())</script> Made with <i class="ti-heart text-danger"></i> By <a
href="http://skkasansar.com">skkasansar</a></p>
    </div>
    <!-- end of page footer -->

            <!-- core  -->
    <script src="./public_html/assets/vendors/jquery/jquery-3.4.1.js"></script>
    <script src="./public_html/assets/vendors/bootstrap/bootstrap.bundle.js"></script>

    <!-- bootstrap affix -->
    <script src="./public_html/assets/vendors/bootstrap/bootstrap.affix.js"></script>

    <!-- wow.js -->
    <script src="./public_html/assets/vendors/wow/wow.js"></script>

    <!-- google maps -->
    <script async defer
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCtme10pzgKSPeJVJrG1O3tjR6lk
98o4w8&callback=initMap"></script>

    <!-- chasma js -->
    <script src="./public_html/assets/js/chasma.js"></script>

</body>
</html>
```