**MINI PROJECT-II**

**(2022-23)**

**‘Tripify’**

**Project Report**

**Institute of Engineering & Technology**

**Submitted By-**

Tanmay Goyal (201500738)

Yuvraj Soni (201500837)

Khushal Agarwal (201500340)

Kalyani Agarwal (201500324)

**Under the Mentorship Of-**

Mr. Bhanu Kapoor

Technical Trainer

Department of Computer Engineering & Applications

**Declaration**

I hereby declare that the work which is being presented in the Bachelor of technology. Project ‘Tripify’, in partial fulfillment of the requirements for the award of the Bachelor of Technology in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of our own work carried under the Mentorship of Mr. Bhanu Kapoor, Mentor, Dept. of CEA, GLA University.

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

**Sign:**

**Name of Candidates:** Yuvraj Soni

Kalyani Agrawal

Khushal Agarwal

Tanmay Goyal

**Certificate**

This is to certify that the project entitled ‘Tripify’, carried out in Mini Project - II Lab, is a Bonafede work by Yuvraj Soni, Kalyani Agrawal, Khushal Agarwal and Tanmay Goyal and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

Signature of Mentor:

Name of Mentor: Mr. Bhanu Kapoor

Date:

**ACKNOWLEDGEMENT**

Presenting the ascribed project paper report in this very simple and official

form, we would like to place my deep gratitude to GLA University for providing

us the mentor Mr. Bhanu Kapoor, our Mentor.

He has been helping me since Day 1 in this project. He provided me with the roadmap, the basic guidelines explaining on how to work on the project. He has been conducting regular meetings to check the progress of the project and providing us with the resources related to the project. Without his help, I wouldn't have been able to complete this project.

And at last but not the least we would like to thank our dear parents for helping us to grab this opportunity to get trained and also my colleagues who helped me find resources during the training.

Thanking You

Sign:

Name of Candidates:

Tanmay Goyal (201500738)

Yuvraj Soni (201500837)

Khushal Agrawal (201500340)

Kalyani Agrawal (201500324)

**ABSTRACT**

‘*Tripify*’ will play an essential role in making decisions like choosing a hotel. This system heavily relies on individuals voluntarily submitted reviews to build the reputation for nearby businesses. Unfortunately, the reviews expose user(s) private information such as visited places to the public and adversaries. Even worse, such location information is usually public because it is that the basic information of companies , and adversaries might be anyone starting from advertisement spammer to physical stalker. This website formalizes the privacy preserving problem in hotel review systems. The framework can preserve users’ location privacy in arbitrary local area and may maintain an honest utility for both the system and each user. We evaluate our framework towards real-world data traces. The results validate that the framework are able to do an honest performance.

**Key Words:** Advanced Web technology, JavaScript, NoSQL, Mongoose, Express.JS, EJS.

**INDEX**

| **CONTENTS** | **PAGE NUMBERS** | **SIGNATURES** |
| --- | --- | --- |
| Cover page | 1 |  |
| Declaration | 2 |  |
| Certificate | 3 |  |
| Acknowledgment | 4 |  |
| Abstract | 5 |  |
| Index | 6 |  |
| Chapter 1:Introduction | 7 |  |
| Chapter 2:Software Requirement Analysis | 8-13 |  |
| Chapter 3:Software Design | 13-14 |  |
| Chapter 4:Language, Technology and Tools used | 14-17 |  |
| Chapter 5:Implementation and Interface | 17-18 |  |
| Chapter 6:Testing | 18-22 |  |
| Chapter 7: Conclusion | 22 |  |
| References | 23 |  |

**CHAPTER-1**

**INTRODUCTION**

**1.1 CONTEXT**

*‘Tripify’* may be a website where users can create and review hotels. In order to review or create a hotel, you want to have an account. This project was designed using Node.js, Express, MongoDB, and Bootstrap. Passport.js was used to handle authentication. The Login Feature gives the user right to login to the website after creation of the account successfully for the website. The login process is on high priority.

**1.2 MOTIVATION**

With social media and streaming services easily accessible, we may have a great online presence already. Therefore we need a website to stand out as a hotel review management system.

**1.3 OBJECTIVE**

To create a website for reviewing hotels by using the data of users such as their geographic location to gather information about the hotels they have visited. This data is stored as per each user. The user can post the review for hotels which can be public and can be viewed by registered users. One of the major aspect of the website is to keep the data of the user private. This website is packed with security features to meet the expectations of privacy and security.

**1.4 SOURCES**

The source of our project (including all the project work, documentations and presentations) will is available at the following link <https://github.com/Khushal-ag/tripify>

**CHAPTER -2**

**SOFTWARE REQUIREMENT ANALYSIS**

**2.1 IMPORTANCE OF TRIPIFY**

People interested in travelling get to find hotels for stay by difficulties and find it being over crowed or polluted by other guests. This site uses the feedback of the people already visited a specific hotel to make easier decisions for other peoples who can choose the hotels based on these feedbacks from other people.

**2.2 PROBLEM STATEMENT**

The problem statement for ‘Tripify’ the hotel finding on the basis of reviews management system project is to develop a system that can assist users in finding suitable hotels based on the reviews of past guests. The system should be able to collect, analyze and process customer feedback from various sources, including online review websites, social media platforms, and direct feedback from customers. The system should provide a user-friendly interface for users to search for hotels, filter results based on various criteria, and view detailed information about each hotel, including ratings, reviews, and amenities. The main objective of the project is to help users make informed decisions about hotel bookings, improve the overall customer experience, and promote transparency in the hospitality industry.

**2.3 HARDWARE AND SOFTWARE REQUIREMENTS**

**Hardware Requirement**

**Processor :** Intel

**Operating System :** Any Operating System

**RAM:** 8 GB (or higher)

**Hard disk:** 256GB

**Software Requirement**

**Software used:** Visual Studio

**Language used:** HTML, CSS, JS

**Database:** no database

**User Interface Design:** React JS

**2.4 MODULES AND FUNCTIONALITIES**

**Modules**

Here are some of the main modules that make up the ‘*TripHive*’:

**Authentication :-** This module allows users to register for an account, log in, and log out. It also provides security features to ensure that only authorized users can access certain features.

**Hotel Management :-** This module allows users to add, edit, and delete hotels. Users can also upload photos and add descriptions, ratings, and reviews to the hotels.

**Search & Filtering :-** This module allows users to search for hotels based on various criteria, such as location, amenities, and activities. Users can also filter the search results based on specific requirements.

**Map Integration :-** This module allows users to view the location of each hotel on a map, making it easier to plan their trip and find their way to the hotel.

**Social Features :-** This module allows users to follow each other, view each other's profiles, and leave comments on hotel reviews. This helps create a sense of community and encourages users to share their experiences with each other.

**Functionalities**

**Search:** A search functionality that allows users to search for hotels by location, name, or other criteria.

**Filters:** Filters that allow users to narrow down their search results based on specific criteria such as price range, rating, amenities, etc.

**Sorting:** Sorting options that enable users to sort search results by price, rating, popularity, distance, etc.

**Reviews:** Displaying user-generated reviews of hotels to help users make informed decisions about where to stay.

**Ratings:** Displaying aggregate ratings of hotels to provide users with an at-a-glance view of a hotel's quality.

**Map view:** A map view that shows hotels in a particular area and allows users to see their location and proximity to other landmarks or attractions.

**Photos:** Displaying photos of hotels, rooms, and amenities to give users a better sense of what they can expect.

**Availability:** Real-time availability information that informs users whether a particular hotel has available rooms for their desired dates.

**WEBSITE**

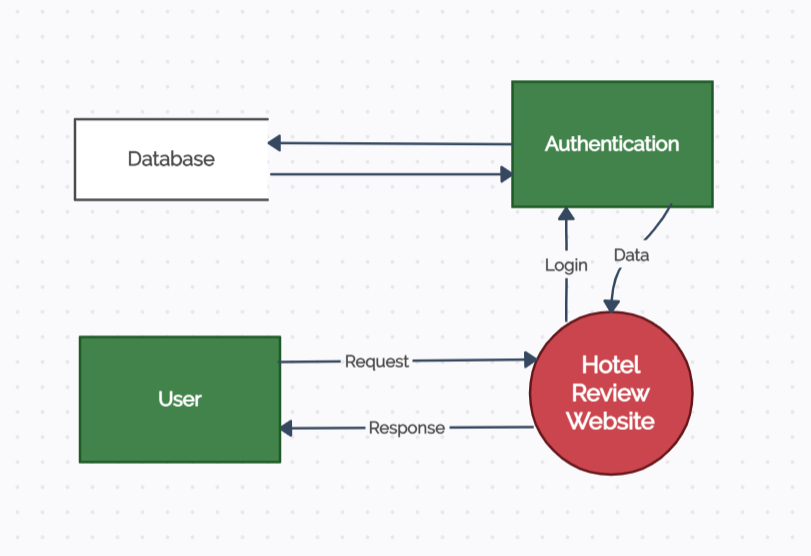
Our project is only a humble venture to satisfy the needs to manage their project work. Several user friendly coding have also adopted. This package shall prove to be a powerful package in satisfying all the requirements of the school. The objective of software planning is to provide a frame work that enables the manger to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

**At the end it is concluded that we have made effort on following points-**

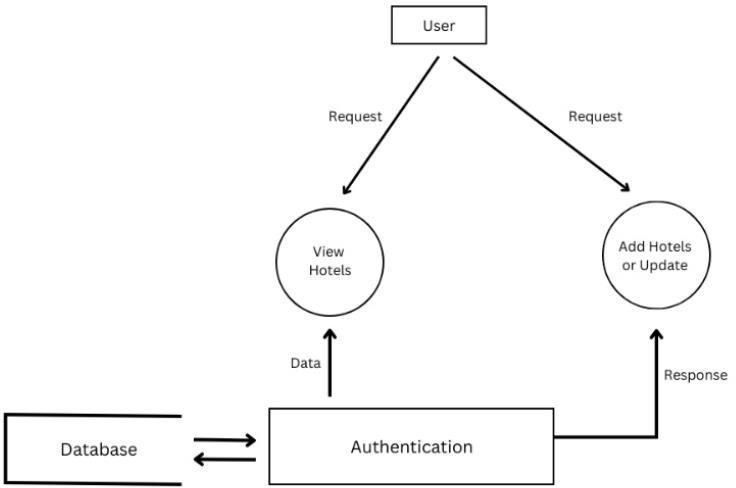
* A description of the background and context of the project and its relation to work already done in the area.
* Made statement of the aims and objectives of the project.
* The description of Purpose, Scope, and applicability.
* We define the problem on which we are working in the project.
* We describe the requirement Specifications of the system and the actions that can be done on these things.
* We understand the problem domain and produce a model of the system, which
* describes operations that can be performed on the system.
* We included features and operations in detail, including screen layouts.
* We designed user interface and security issues related to system.
* Finally the system is implemented and tested according to test cases.

**Data Flow diagram:-**

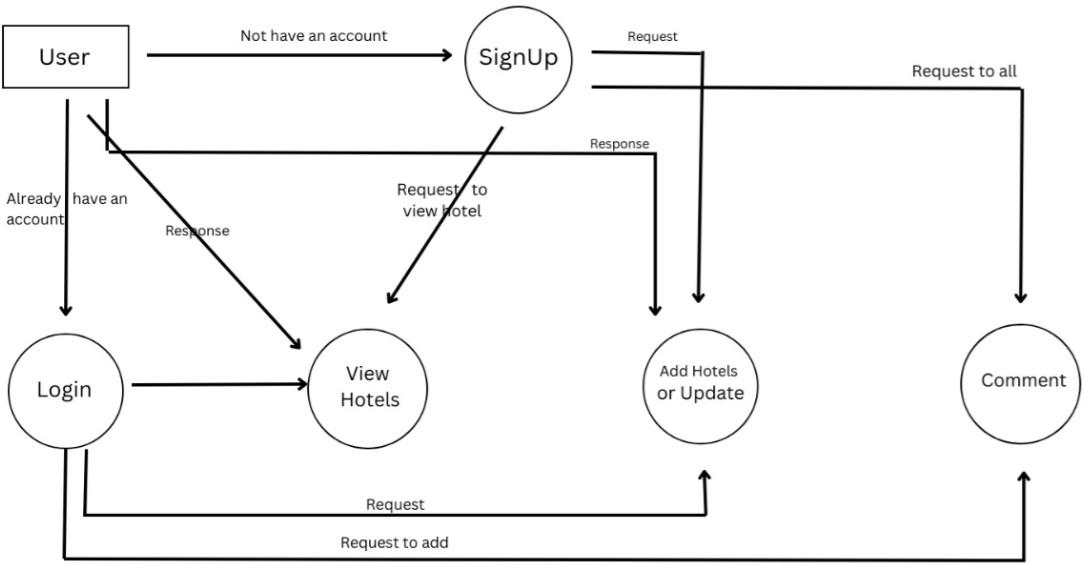
Data flow diagram is the starting point of the design phase that functionally decomposes the requirements specification. A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flows in the system. A DFD describes what data flow rather than how they are processed, so it does not hardware, software and data structure.

DFD Level-0

DFD Level-1



DFD Level-2



**CHAPTER-3**

**SOFTWARE DESIGN**

**3.1 USE-CASE DIAGRAM:**



**CHAPTER-4**

**LANGUAGES, TECHNOLOGY AND TOOLS USED**

**4.1 HTML**

HTML is an acronym which stands for **Hyper Text Markup Language** which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

**Hyper Text:** HyperText simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. HyperText is a way to link two or more web pages (HTML documents) with each other.

**Markup language:** A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

**Web Page:** A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. **With the help of HTML only, we can create static web pages**.

**4.2 CSS**

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

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

**4.3 JAVASCRIPT**

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

**4.5 API**

API is the abbreviation of the term ***Application Programming Interface.*** It is the software responsible for the connection for the communication and information exchange between two apps. API connects two devices or programs in order to facilitate the exchange of information between them. It is the interface that serves the other parts of the software. The API specifications are the standards or documents designed to describe the creation of such connections. If a computer system meets these standards, then it is said to expose an API. The specification or implementation both are known as the API.

**4.6 Tailwind**

Tailwind CSS is a utility-first CSS framework designed to enable users to create applications faster and easier. You can use utility classes to control the layout, color, spacing, typography, shadows, and more to create a completely custom component design — without leaving your HTML or writing a single line of custom CSS.

**4.7 GITHUB**

GitHub is an immense platform for code hosting. It supports version controlling and collaboration and allows developers to work together on projects. It offers both distributed version control and source code management (SCM) functionality of Git. It also facilitates collaboration features such as bug tracking, feature requests, task management for every project.

Essential components of the GitHub are:

* Repositories
* Branches
* Commits
* Pull Requests
* Git (the version control tool GitHub is built on)

**4.8 VSCode**

**Visual Studio Code** (famously known as **VS Code**) is a free open source text editor by Microsoft. VS Code is available for Windows, Linux, and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times.

VS Code supports a wide array of programming languages from Java, C++, and Python to CSS, Go, and Dockerfile. Moreover, VS Code allows you to add on and even creating new extensions including code linters, debuggers, and cloud and web development support.

**4.9 MongoDB**

**MongoDB** is a document-oriented NoSQL database used for high volume data storage. Instead of using tables and rows as in the traditional relational databases, MongoDB makes use of collections and documents. Documents consist of key-value pairs which are the basic unit of data in MongoDB. Collections contain sets of documents and function which is the equivalent of relational database tables. MongoDB is a database which came into light around the mid-2000s.

**CHAPTER-5**

**IMPLEMENTATION AND INTERFACE**

**5.1 IMPLEMENTATION**

Javascript is a scripting language used to enhance the functionality of thebrowser. Java script is integrated with HTML and navigator 2.02. JavaScript facilitates the developer with properties related to document windows, frames, loaded documents, and links.

**5.2 USER INTERFACE DESIGN**

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

The following steps are various quidelines for User Interface Design:

1. The system user should always be aware of what to do next.

2. The screen should be formatted so that various types of information, instructions

and messages always appear in the same general display area.

3. Message, instructions or information should be displayed long enough to allow

the system user to read them.

4. Use display attributes sparingly.

5. Default values for fields and answers to be entered by the user should be specified.

6. A user should not be allowed to proceed without correcting an error.

7. The system user should never get an operating system message or fatal error.

**CHAPTER-6**

**TESTING**

**Implementation and Software Specification Testing**

**Detailed Design of Implementation**

This phase of the systems development life cycle refines hardware and software specifications, establishes programming plans, trains users and implements extensive testing procedures, to evaluate design and operating specifications and/or provide the basis for further modification.

**Technical Design**

**This activity builds upon specifications produced during new system design, adding detailed technical specifications and documentation.**

**Test Specifications and Planning**

This activity prepares detailed test specifications for individual modules and programs, job streams, subsystems, and for the system as a whole**.**

**Programming and Testing**

This activity encompasses actual development, writing, and testing of program units or modules.

**User Training**

This activity encompasses writing user procedure manuals, preparation of user training materials, conducting training programs, and testing procedures.

**Acceptance Test**

A final procedural review to demonstrate a system and secure user approval before a system becomes operational.

**Installation Phase**

In this phase the new Computerized system is installed, the conversion to new procedures is fully implemented, and the potential of the new system is explored.

**System Installation**

The process of starting the actual use of a system and training user personnel in its operation.

**Review Phase**

This phase evaluates the successes and failures during a systems development project, and to measure the results of a new Computerized Transystem in terms of benefits and savings projected at the start of the project.

**Development Recap**

A review of a project immediately after completion to find successes and potential problems in future work.

**Post-Implementation Review**

A review, conducted after a new system has been in operation for some time, to evaluate actual system performance against original expectations and projections for cost-benefit improvements. Also identifies maintenance projects to enhance or improve the system.

**THE STEPS IN THE SOFTWARE TESTING**

The steps involved during Unit testing are as follows:

a. Preparation of the test cases.

b. Preparation of the possible test data with all the validation checks.

c. Complete code review of the module.

d. Actual testing done manually.

e. Modifications done for the errors found during testing.

f. Prepared the test result scripts.

**The unit testing done included the testing of the following items:**

1. Functionality of the entire module/forms.
2. Validations for user input.
3. Checking of the Coding standards to be maintained during coding.
4. Testing the module with all the possible test data.
5. Testing of the functionality involving all type of calculations etc.
6. Commenting standard in the source files.

After completing the Unit testing of all the modules, the whole system is integrated with all its dependencies in that module. While System Integration, We integrated the modules one by one and tested the system at each step. This helped in reduction of errors at the time of the system testing.

**The steps involved during System testing are as follows:**

* Integration of all the modules/forms in the system.
* Preparation of the test cases.
* Preparation of the possible test data with all the validation checks.
* . Actual testing done manually.
* Recording of all the reproduced errors. Modifications done for the errors found during testing.
* Prepared the test result scripts after rectification of the errors.

**The System Testing done included the testing of the following items:**

1. Functionality of the entire system as a whole.

2. User Interface of the system.

3. Testing the dependent modules together with all the possible test data scripts.

4. Verification and Validation testing.

5. Testing the reports with all its functionality.

After the completion of system testing, the next following phase was the Acceptance Testing. Clients at their end did this and accepted the system with application. Thus, we reached the final phase of the project.

**There are other six tests, which fall under special category. They are described below:**

* Peak Load Test: it determines whether the system will handle the volume of handles of activities that occur when the system is at the peak of its processing demand. For example, test the system by activating all terminals at the same time.
* Storage Testing: It determines the capacity of the system to store transaction data on a disk
* or in other files.
* Performance Time Testing: it determines the length of time system used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response.
* Recovery Testing: This testing determines the ability of user to recover data or re-start system after failure. For example, load backup copy of data and resume processing without data or integrity loss.
* Procedure Testing: It determines the clarity of documentation on operation and uses of system by having users do exactly what manuals request. For example, powering down
* system at the end of week or responding to paper-out light on printer. Human Factors Testing: It determines how users will use the system when processing data or preparing reports.

**CHAPTER-7**

**CONCLUSION**

Our project is only a humble venture to satisfy the needs to manage their project work. Several user friendly coding have also adopted. This package shall prove to be a powerful package in satisfying all the requirements of the school. The objective of software planning is to provide a frame work that enables the manger to make reasonable estimates made within limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

**At the end it is concluded that we have made effort on following points...**

* + A description of the background and context of the project and its relation to work already done in the area. Made statement of the aims and objectives of the project.
  + The description of Purpose, Scope, and applicability.
  + We define the problem on which we are working in the project.
  + We describe the requirement Specifications of the system and the actions that
  + can be done on these things.
  + We understand the problem domain and produce a model of the system, which
  + describes operations that can be performed on the system.
  + We included features and operations in detail, including screen layouts.
  + We designed user interface and security issues related to system.
  + Finally the system is implemented and tested according to test cases.

**REFERENCES**

* Project Link: https://github.com/Khushal-ag/tripify
* MDN Web Docs : <https://developer.mozilla.org/en-US/docs/Web/>
* Tailwind
* W3School : https://www.w3schools.in/