1. INTRODUCTION TO PROJECT

Social Networking - It's the way the 21st century communicates now. Social media sites have become an integral part of people's lives, as they provide various benefits such as communication, entertainment, education, and commerce. Social networking is the grouping of individuals into specific groups, like small rural communities or a neighborhood subdivision these websites allow users to interact with each other, create and share content, and form online communities. Although social networking is possible in person, especially in the workplace, universities, and high schools, it is most popular online. This is because unlike most high schools, colleges, or workplaces, the internet is filled with millions of individuals who are looking to meet other people.

Social network is the mapping and measuring of relationships and flows between people, groups, organizations, computers, URLs, and other connected information/knowledge entities. The nodes in the network are the people and groups while the links show relationships or flows between the nodes. Social network provides both a visual and a mathematical analysis of human relationships. However, developing a social media site is not an easy task, as it requires careful planning, design, implementation, and maintenance.

Social Networking Website project itself is a huge project comprising various features like profile updating, friend's list organization and various other application to enhance the overall look and feel of the website.

However, in this project I am basically working on naming "Online Social Site [Globe Trotters]" have two essential feature or module (PROFILE MANAGEMENT & CONNECTIONS MANAGEMENT). PROFILE MANAGEMENT module maintains the profile of a user like name, password, liked posts, etc. CONNECTIONS MANAGEMENT module maintains the created connections list, handles and sends request to the other user. Profiles and Connections lists are two key features on social network sites. The third is a public commenting feature ('Testimonials', 'Comments', 'The Wall', 'Liking Posts'). This feature allows individuals to comment on other users' posts. These comments are displayed prominently and visible for all other users.

The aim of this project is to create a social media site that meets the needs and expectations of the travelers and wonderers.

1.1 Problem Statement

We define online social network sites as web-based services that allow individuals to construct a public or semi-public profile within a bounded system, articulate a list of other users with whom they share a connection, and view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site. Since their introduction, social network sites (SNSs) such as MySpace, Facebook, Cyworld and Hi5 have attracted millions of users, many of whom have integrated these sites into their daily practices. As of this writing, there are hundreds of SNSs, with various technological affordances, supporting a wide range of interests and practices. While their key technological features are fairly consistent, the cultures that emerge around SNSS are varied. Most sites support the maintenance of pre- existing social networks, but others help strangers connect based on shared interests, political views, or activities. Some sites cater to diverse audiences, while others attract people based on common language or shared racial, sexual, religious, or nationality- based identities. Sites also vary in the extent to which they incorporate new information and communication tools, such as mobile connectivity, blogging, and photo/ video-sharing.

Social networking sites are not only for you to communicate or interact with other people globally but, this is also one effective way for business promotion. A lot of business minded people these days are now doing business online and use these social networking sites to respond to customer queries. It isn't just a social media site used to socialize with your friends but also, represents a huge pool of information from day to day living. A social networking service is an online service, platform, or site that focuses on facilitating the building of social networks or social relations among people who, for example, share interests, activities, backgrounds, or real-life connections. A social network service consists of a representation of each user (often a profile), his/her social links, and a variety of additional services. Most social network services are web- based and provide means for users to interact over the Internet, such as e-mail and instant messaging. Online community services are sometimes considered as a social network service, though in a broader sense, social network service usually means an individual- centered service

whereas online community services are group-centered. Social networking sites allow users to share ideas, activities, events, and interests within their individual networks.

1.2 Domain Study

Today, social networking site use is a major activity for internet users from a wide range of demographic groups. Younger adults are especially avid adopters, but social networking continues to grow in popularity for older adults as well. Six out of ten internet users ages 50-64 are social networking site users, as are 43% of those ages 65 and older. Although online seniors are less likely than other age groups to use social networking sites, adoption rates for those 65 and older have tripled in the last four years (from 13% in the spring of 2009 to 43% now).

The main types of social networking services are those that contain category places (such as former school year or classmates), means to connect with friends (usually with self-description pages), and a recommendation system linked to trust. Popular methods now combine of Facebook, Google+, YouTube, LinkedIn, Instagram, Pinterest, Tumblr and Twitter widely used worldwide.

Many of these early communities focused on bringing people together to interact with each other through chat rooms, and encouraged users to share personal information and ideas via personal web pages by providing easy-to-use publishing tools and free or inexpensive web space. Some communities - such as Classmates.com - took a different approach by simply having people link to each other via email addresses. In the late 1990s, user profiles became a central feature of social networking sites, allowing users to compile lists of "friends" and search for other users with similar interests. New social networking methods were developed by the end of the 1990s, and many sites began to develop more advanced features for users to find and manage friends. This newer generation of social networking sites began to flourish with the emergence of Six Degrees.com in 1997 followed by Makeoutclub in 2000, HubCulture and Friendster in 2002 and soon became part of the Internet mainstream. Friendster was followed by MySpace and LinkedIn a year later, and eventually Bebo. Friendster became very popular in the Pacific Island. Orkut became the first social networking in Brazil and then also grow fast in India (Madhavan,

2007). Attesting to the rapid increase in social networking sites' popularity, by 2005, it was reported that MySpace was getting more page views than Google. Facebook, launched in 2004, became the largest social networking site in the world in early 2009. Facebook was first introduced (in 2004) as a Harvard social networking (Cassidy, 2006).

1.3 Project Scope

The main purpose of our project is to allow travelers to connect with each other, share travel experiences, and plan trips together. We tried to design a online social networking website for travelers/wonderers making the user experience and design elements that will make the website easy to use and visually appealing. In other words, our Globe Trotters Website has, following objectives:

- **Connecting Traveller's**: The website can provide a platform for travellers to connect with each other, share their travel experiences, and plan trips together. This can help traveller's find like-minded people to travel with and make new friends.
- **Sharing Travel Experience**: This website allow upload the photos and write their own travel experience in details.
- **Sharing Travel Information**: The website can allow travellers to share information about their travel destinations, including reviews of hotels, restaurants, and tourist attractions. This can help other traveller's plan their trips better and make informed decisions.
- **Providing Travel Tips**: The website can provide travel tips and advice to help traveller's plan their trips better. This can include information on visa requirements, local customs, and safety tips.

2. SYSTEM REQUIREMENTS

2.1 Literature Survey

The Web-based social networking services make it possible to connect people who share interests and activities across political, economic, and geographic borders. Through e-mail and instant messaging, online communities are created where a gift economy and reciprocal altruism are encouraged through cooperation. Information is suited to a gift economy, as information is a non-rival good and can be gifted at practically no cost.

Facebook and other social networking tools are increasingly the object of scholarly research. Scholars in many fields have begun to investigate the impact of social-networking sites, investigating how such sites may play into issues of identity, privacy, social capital, youth culture, and education.

Several websites are beginning to tap into the power of the social networking model for philanthropy. Such models provide a means for connecting otherwise fragmented industries and small organizations without the resources to reach a broader audience with interested users. Social networks are providing a different way for individuals to communicate digitally. These communities of hypertexts allow for the sharing of information and ideas, an old concept placed in a digital environment.

2.2 Functional Specifications

- **DacprojectContext Object:**

  The DacprojectContext class acts as a wrapper for all server functions for our social networking site. It essentially acts as a link between all of the information such as user, post, comments, trips, etc. to our database. When any other model object such as a post is pulled from the server, a temporary copy is made. If that temporary copy is changed in any way. The new version must be sent to the server in order to update the permanent copy. The reason behind local copies is that all the necessary information for the object is sent over in one easy-to-use package. Then the update to the database can be done all at once

by sending back that single object. There is no need for multiple functions or a function that takes a large number of parameters.

- **User Object:**

  Each user who wants to use the site must create an account. This class object contains all the information that shows up in the user's profile. The only time the User class needs to be updated is when the user changes his/her username and/or password. This class is called any time a user tries to visit website.

- **Post Like Object, Post Comment Object, Trip Object, Connection Object:**

  These objects contain unique information for a particular type of action a user performs. All of these contain a reference to the user account that owns them. Each object is a "working- copy" of an object in the Server. Anytime one of these objects is created on the Server, an entry of its creation is added to the news feed database.

- **Post Object:**

  As people use their social networking account, they will want to be able to upload interesting images from their journey/visits to different location to share on their wall. The media upload section will be located at the ‘Share image’ button as well as provide a short description of the file to be uploaded. In order to prevent users from uploading potentially malicious files such as executables, only certain file extensions are supported. These allowed extensions cover popular image extensions such as png, jpg, jpeg.

  A Post object is created when a user composes a new post which is viewable to all the users. Post object can hold comments and likes from other users. If a user creates a post to his or her own account it immediately starts showing on his or her homepage and profile page as well.

  The inner workings of this uploading process are surprisingly simple, since the HTTP server takes care of requesting the file from the client automatically. After the user clicks upload, the file is sent to the server and stored in a temporary location. Through ASP.net Core server all information about these temporary files can be accessed through the $ Form File variable. The temporary location is stored in that variable and can be used to pull the

file name and file extension. That extension is then checked against an array of allowed extensions. If the extension is in the list, then the file is named uniquely using Guid class from System.Guid package, the file is then moved to a permanent location on server named Images. If the extension is not on the list, server will return Bad Request response with message “Invalid File Format”.

- **Profile Pic Object:**

  A Profile Pic object works very similar to a post object. The only difference is that it won’t allow you to add any description while uploading your profile picture. Another difference is that it doesn’t hold any like and comment from other users.

- **Message Object:**

  A Message object is created when a user composes a new message to be sent to other connection. After it is confirmed that the connection is located in the database, the Message object adds its information to the database. Message is then entered in the database and will be visible the recipient user.

- **Connections Object:**

  The most important feature is being able to follow and unfollow other users. In our Online Social Networking site, making connections is a fairly straightforward process. Users can type in the name of another user in the search bar at the top of their home page. The database is queried for an account that has the search term contained in the full name.

  For example, User A could search for User B in the search bar. After clicking on User B's profile, User A will see a button that says Follow If not following already. Clicking on it will establish the new connection [starts following the B] or break the connection [unfollow the user B].

- **User Registration/Account Creation:**

  When a user accesses the site for the first time, he/she must create an account before using any of the site features. The account creation process is broken into two sections. The first section deals with the login information and is required for the user to fill out. This includes

the email, password, and password confirmation. The purpose behind the password confirmation is to ensure that the user didn't accidentally mistype when creating a password. The second section deals with information about who you are such as full name, Date of Birth, unique user name, mobile and gender. Most of these fields are optional except for your unique user's name. It wouldn't be much of a social network if everyone was named anonymous. Once the user clicks create account, a new account, account details are added to the server and the user is brought back to the log in page.

2.3 Non-Functional Specifications

- Secure access of confidential data by user name and password. This application is secure users' password by encrypting it using non-reversible SHA512 cryptographic hash function. The HMAC-SHA512 class object is initialized using byte array of the secret Key which, computes the hash value of the input password data using the ComputeHash method, and returns the hash value as a string to database.
- Better component design to get better performance at peak time.
- The database used here is robust, reliable & fast. So users will have to wait for the output very short time.
- With ASP.NET Core as web framework [Backend] the web application can be run on Windows, Mac or Linux operating system.
- There is no case of redundancy in the database so it will not take extra memory space.

2.4 Software Tools Specifications

- **MySQL:**

  MySQL is a relational database management system (RDBMS) that is widely used for building web applications. It is an open-source software that is free to use and can be downloaded from the official MySQL website. The simplest and recommended method is to download MySQL Installer for Windows, though further configuration is often required to adjust security and optimization settings.

  MySQL is known for its scalability, performance, and ease of use. It can handle large amounts of data and can be used to build complex web applications that require fast and

efficient data processing . MySQL is also highly customizable and can be configured to meet the specific needs of different applications .

Though MySQL began as a low-end alternative to more powerful proprietary databases, it has gradually evolved to support higher-scale needs as well. It is still most commonly used in small to medium scale single-server deployments.

- **React js:**

  React.js is a popular JavaScript library used for building user interfaces. It was developed by Facebook and is now maintained by Facebook and a community of individual developers and companies. React.js allows developers to build reusable UI components and manage the state of their applications more efficiently. It uses a declarative syntax that makes it easier to understand and debug code. React.js is often used in conjunction with other libraries and frameworks such as Redux for state management, React Router for routing, and Next.js for server-side rendering. It is widely used by developers to build web applications, mobile applications, and desktop applications.

  Overall, React is a powerful tool for building user interfaces that are fast, efficient, and easy to maintain. Its flexibility, declarative syntax, and component-based architecture make it a popular choice for developers who want to build complex web applications with ease.

- **Back End: ASP.NET Core:**

  ASP.NET Core is a web application framework that is used to build back-end web APIs. It is an open-source framework that is free to use and can be downloaded from the official ASP.NET Core website[1].

  ASP.NET Core is designed to be cross-platform, meaning that it can run on Windows, macOS, and Linux operating systems [2]. It is also designed to be modular, which means that developers can choose which components they want to use in their applications [3].

  ASP.NET Core provides a number of features that make it an ideal choice for building back-end web APIs. These features include:

- **Routing:** ASP.NET Core provides a powerful routing system that allows developers to map URLs to actions in their controllers [4].
- **Model binding:** ASP.NET Core can automatically bind data from HTTP requests to objects in the application.
- **Dependency injection:** ASP.NET Core provides a built-in dependency injection system that makes it easy to manage dependencies between components in the application.
- **Middleware**: ASP.NET Core uses middleware components to handle requests and responses in the application pipeline.
- **Authentication and authorization**: ASP.NET Core provides built-in support for authentication and authorization, making it easy to secure web APIs.

ASP.NET Core can be used with a variety of front-end frameworks, such as React, Angular, and Vue.js, to build full-stack web applications.

2.5 Hardware Requirements

Processor	:	Intel Pentium IV 2.0 GHz and above
RAM	:	512 MB and above
Hard disk	:	80GB and above
Monitor	:	CRT or LCD monitor
Keyboard	:	Normal or Multimedia
Mouse	:	Compatible mouse

2.6 Software Requirements

Front End	:	React js
Back End	:	Visual Studio 2022 with ASP.NET Core and web development workload
Operating System	:	Windows XP or above
Browser	:	Any latest browser

3. DESIGN SPECIFICATIONS

3.1 Modular Design

The application comprises following major modules:

Register to be a member Module

This module provides functionalities for those people who wants to open an account. Applicants can post their views and share their travel experiences with personal and professional details. They can also update the profile as frequently as required. The members can also browse through the connections profile available. Members can also get message when their friends message them.

Profile Module

This module provides functionalities related to member's profile. Logged users can see their details and if they wish to change any of their information, they can edit it.

3.2 System Design

3.2.1 Data Flow Diagram

Data flow diagrams model the flow of data into, though, and out of an information system:

- show the processes that change or transform data
- show the movement of data between processes
- represent a system as a network of processes which transform data flowing between them

The user screen flow shows what a user of the community will see. After successfully logging on, the user will be given various links (such as search users, create post, update profile, etc.), and be able to select options from there, or go back to their home.

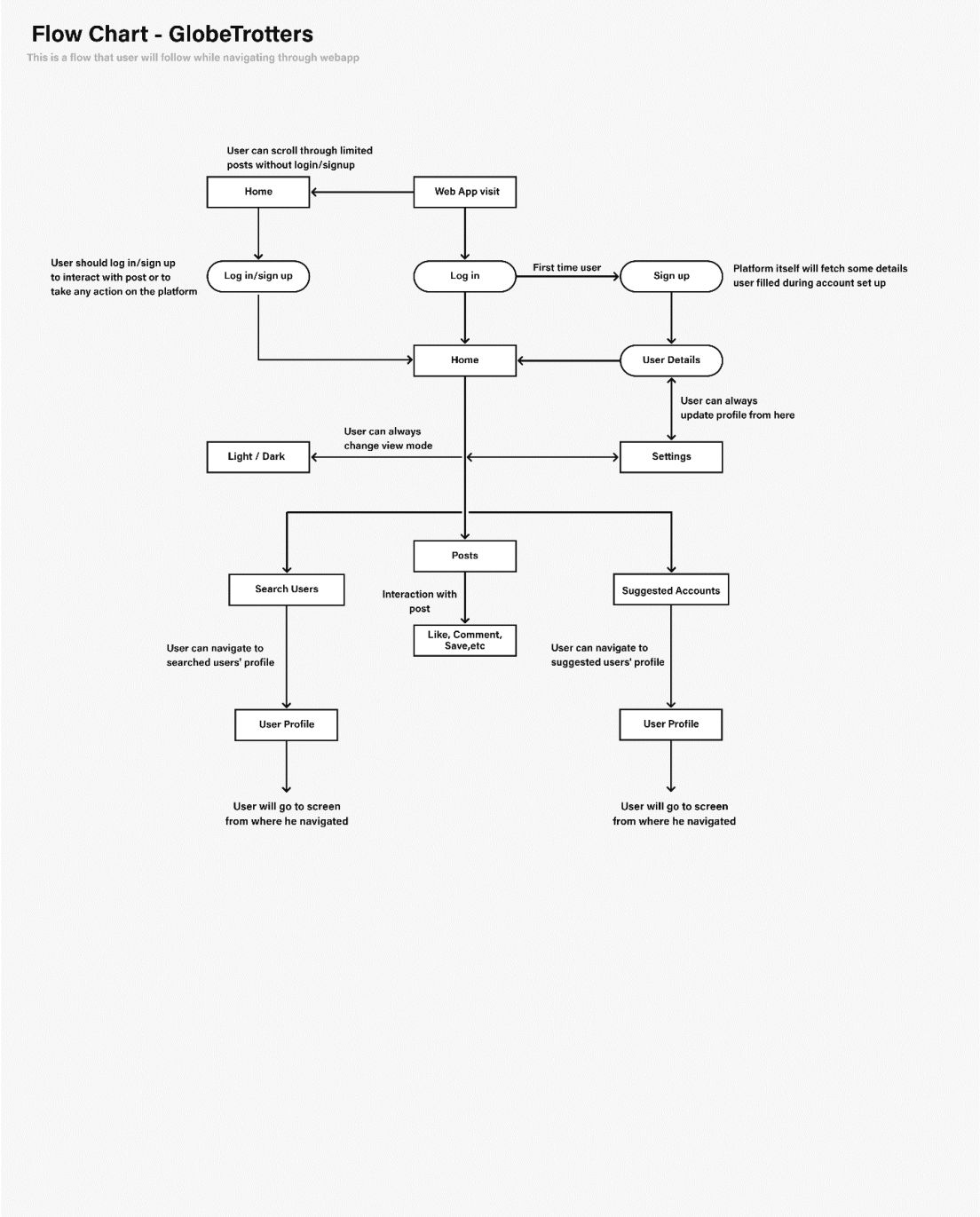


Figure 3.1 Client Screen Flow

Use case diagrams.

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modelling Language), a standard notation for the modelling of real-world objects and systems.

System objectives can include planning overall requirements, validating

a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components:

- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which are the specific roles played by the actors within and around the system.
- The relationships between and among the actors and the use cases.

\

- **Login/Registration**

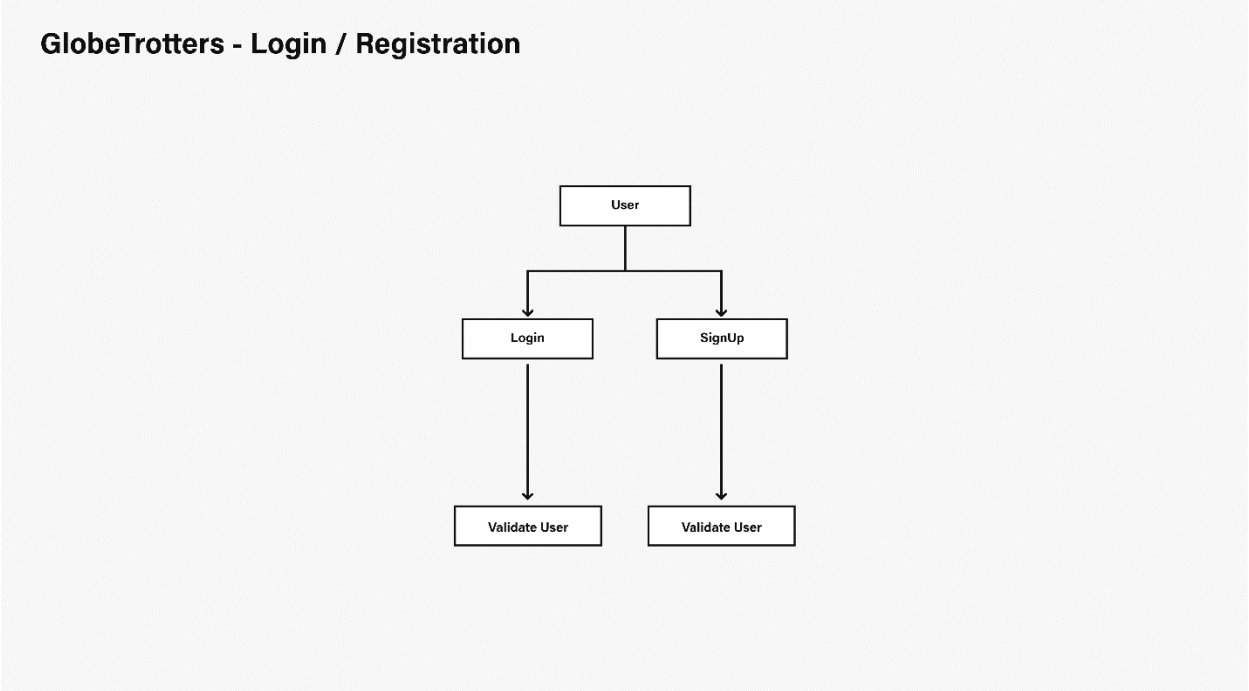


Figure 3.2 Login/Registration

3.2.2 Context Diagram

The highest-level data flow diagram is the context diagram.

- The context diagram shows the interaction of the system with its environment in terms of data flows
- The context diagram defines the boundary of the system (the scope of the system)
- Only the data flows which leave the system and the data flows which come from outside the system are shown.

0 Level DFD:

A level 0 DFD, also called a fundamental system model or context diagram represents the entire software element as a single bubble with input and output data indicated by incoming and outgoing arrows, respectively.

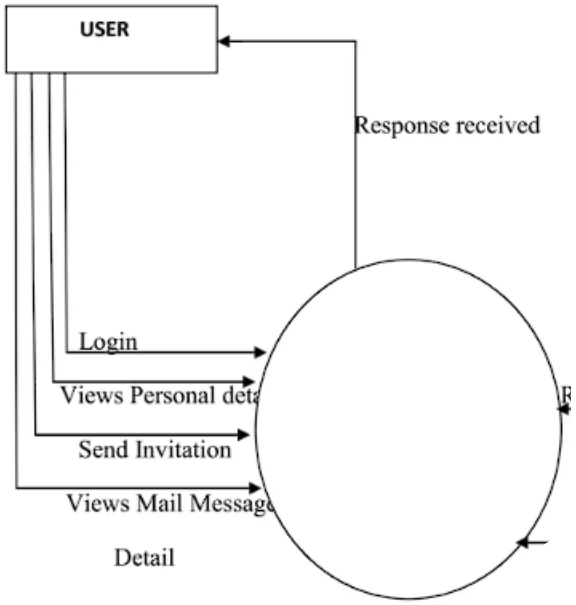


Figure 3.3 0 level DFD

1 Level DFD:

This level of DFD provide more detailed structure. It provides a detailed view of requirements and flow of data from 1 bubble to another.

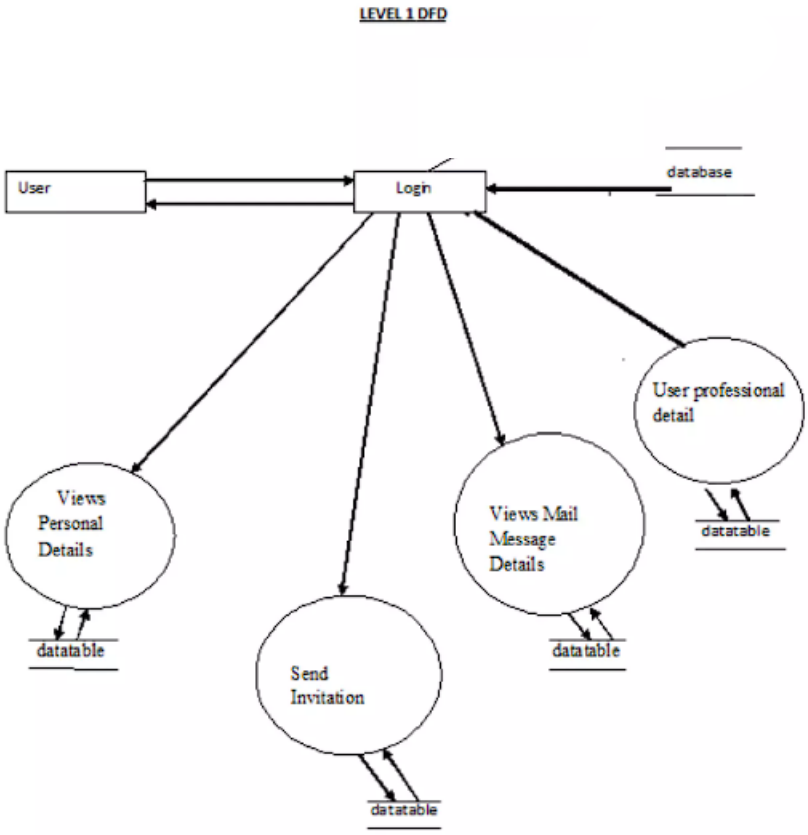


Figure 3.4 1 level DFD

3.2.3 Entity Relationship Diagram

The entity relationship model is a high-level data model. It is based on a perception of a real world that consists of a collection of basic objects, called entities, and of relationship among these objects. It was developed to facilitate database design by allowing specification of an enterprise schema, which represent the overall logical structure of a database.

Entity: An entity is an object that has its existence in the real world. It includes all those "things" about which data is collected. An entity may be a tangible object such as a student, a place or a part. It may also be non-tangible such as an event, a job title or a customer account. For example, if we say that a customer buys goods, it means customer and goods are entities.

Diagrammatically, entities are represented in rectangles.

An Entity Set: It is a set of entities of the same type that share the same properties, or attributes. The set of all persons who are customers at a given bank, example, can be defined as the entity set customer.

Attributes: Attributes are units that describe the characteristics or properties of entities. In a database, entities are represented by tables and attributes by columns. For example, a customer entity might have numerous attributes such as code, name and addresses. Similarly, the goods entity may have attributes like code and price. They are drawn in elliptical shapes along with the entity rectangles.

The entity relationship diagram of mailing system is drawn on the next page:

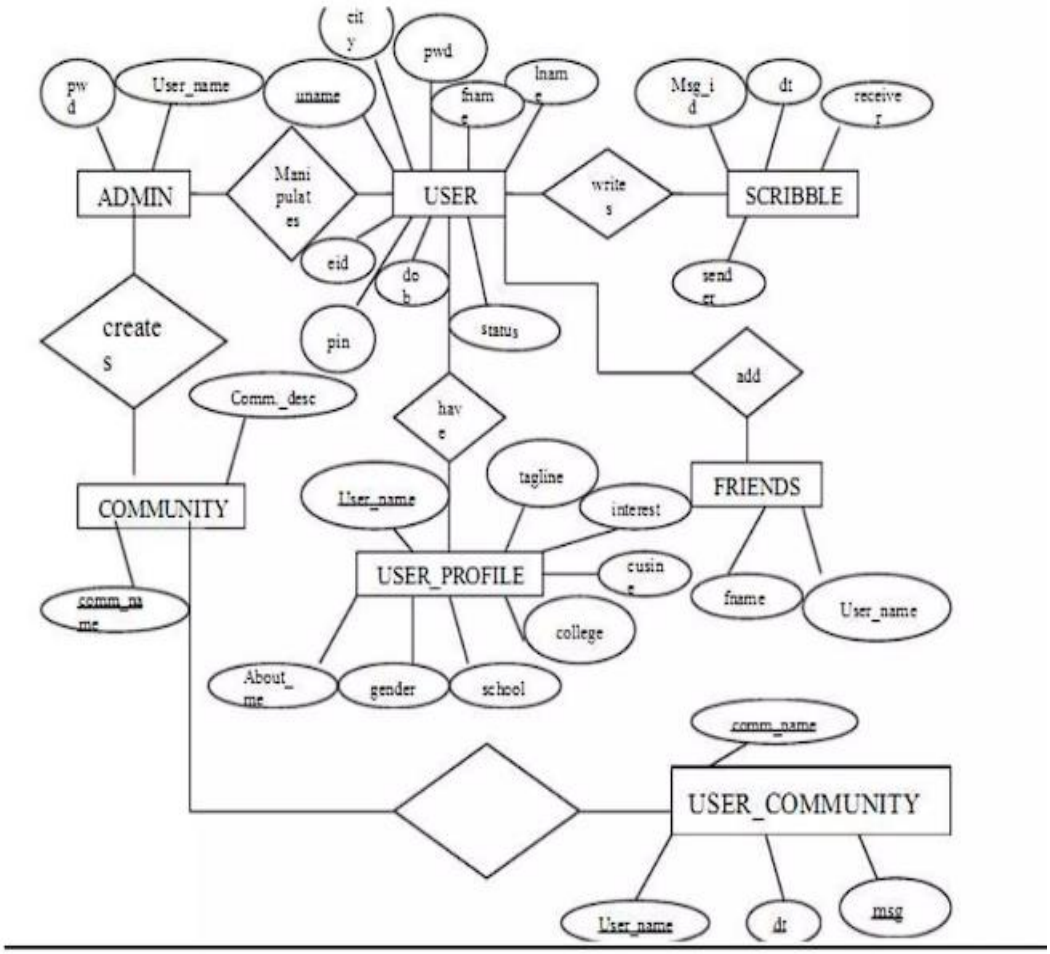


Figure 3.5 ERD

3.2.4 Database

Data base is used to store the relevant information of the individuals. A database is a collection of rows and columns in which rows indicates the tuple and column indicates the domain of table. Database design is the process of producing a detailed data model of a

database. This logical data model contains all the needed logical and physical design choices and physical storage parameters. Need to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of the logical design of the relation of the base data structures used to store the data. In the relational model these are the classes and named relationships. However, the term database design could also be used to apply to overall process of designing, not just the base data structure, but also the forms and queries used as part of the overall database application within the database management system (DBMS).

Field	Type	Null	Key	Default	Extra
user_id	int	NO	PRI	NULL	auto_increment
email	varchar(50)	NO	UNI	NULL	
password	varchar(100)	NO		NULL	
user_name	varchar(50)	NO	UNI	NULL	
name	varchar(50)	YES		NULL	
dob	date	YES		NULL	
gender	varchar(10)	YES		NULL	
mobile	varchar(12)	YES		NULL	
profile_photo	varchar(500)	YES		NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	
extra3	varchar(0)	YES		NULL	
extra4	varchar(0)	YES		NULL	

Figure 3.6 List of Tables

Field	Type	Null	Key	Default	Extra
post_id	int	NO	PRI	NULL	auto_increment
user_id	int	YES	MUL	NULL	
post_url	varchar(500)	YES		NULL	
latitude	double	YES		NULL	
longitude	double	YES		NULL	
location_name	varchar(200)	YES		NULL	
created_datetime	datetime	YES		NULL	
caption	varchar(200)	YES		NULL	
likes_count	int	YES		NULL	
comments_count	int	YES		NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.7 post

Field	Type	Null	Key	Default	Extra
like_id	int	NO	PRI	NULL	auto_increment
post_id	int	YES	MUL	NULL	
user_id	int	YES	MUL	NULL	
created_datetime	datetime	YES		NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.8 post like

Field	Type	Null	Key	Default	Extra
comment_id	int	NO	PRI	NULL	auto_increment
post_id	int	YES	MUL	NULL	
user_id	int	YES	MUL	NULL	
created_datetime	datetime	YES		NULL	
comment_content	varchar(200)	YES		NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.9 post comment

Field	Type	Null	Key	Default	Extra
connection_id	int	NO	PRI	NULL	auto_increment
following_to	int	NO	MUL	NULL	
followed_by	int	NO	MUL	NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.10 connection

Field	Type	Null	Key	Default	Extra
trip_id	int	NO	PRI	NULL	auto_increment
user_id	int	YES	MUL	NULL	
latitude	double	YES		NULL	
longitude	double	YES		NULL	
location_name	varchar(200)	YES		NULL	
itinerary	varchar(500)	YES		NULL	
interested_count	int	YES		NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.11 trip

Field	Type	Null	Key	Default	Extra
interested_id	int	NO	PRI	NULL	auto_increment
user_id	int	YES	MUL	NULL	
trip_id	int	YES	MUL	NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.12 interested

Field	Type	Null	Key	Default	Extra
community_id	int	NO	PRI	NULL	auto_increment
admin_user_id	int	YES	MUL	NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.13 community

Field	Type	Null	Key	Default	Extra
admin_id	int	NO	PRI	NULL	auto_increment
user_id	int	YES	MUL	NULL	
extra1	varchar(0)	YES		NULL	
extra2	varchar(0)	YES		NULL	

Figure 3.14 admin

4. IMPLEMENTATION DETAILS [Screenshots]

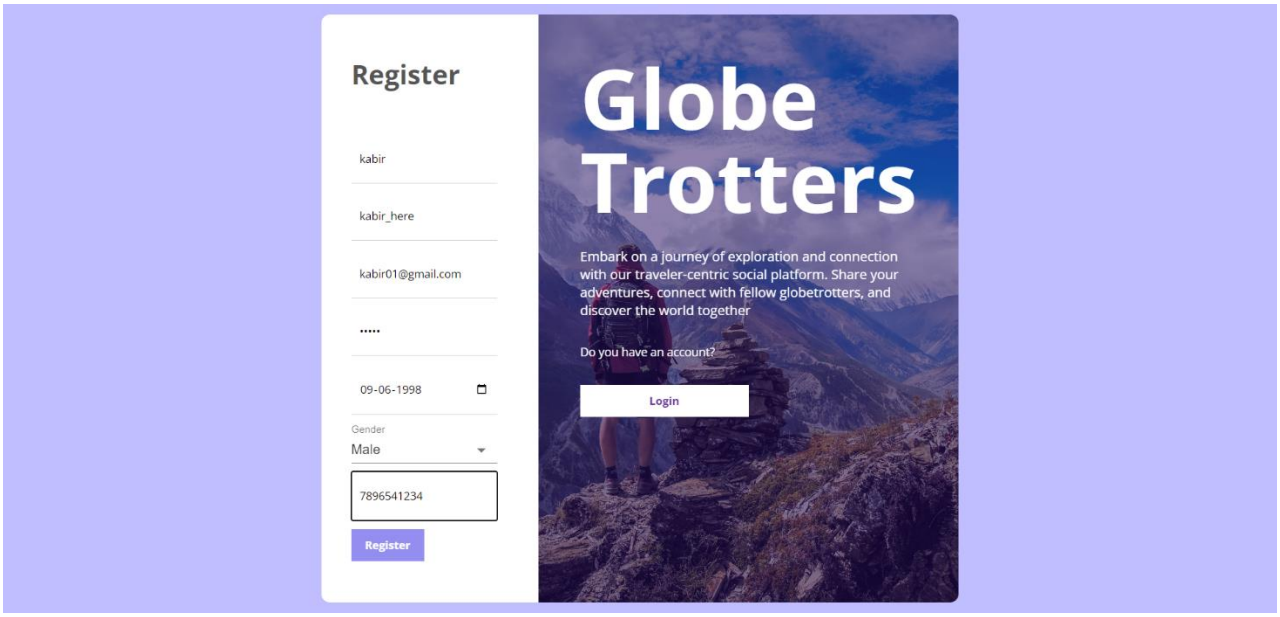


Figure 4.1 Register

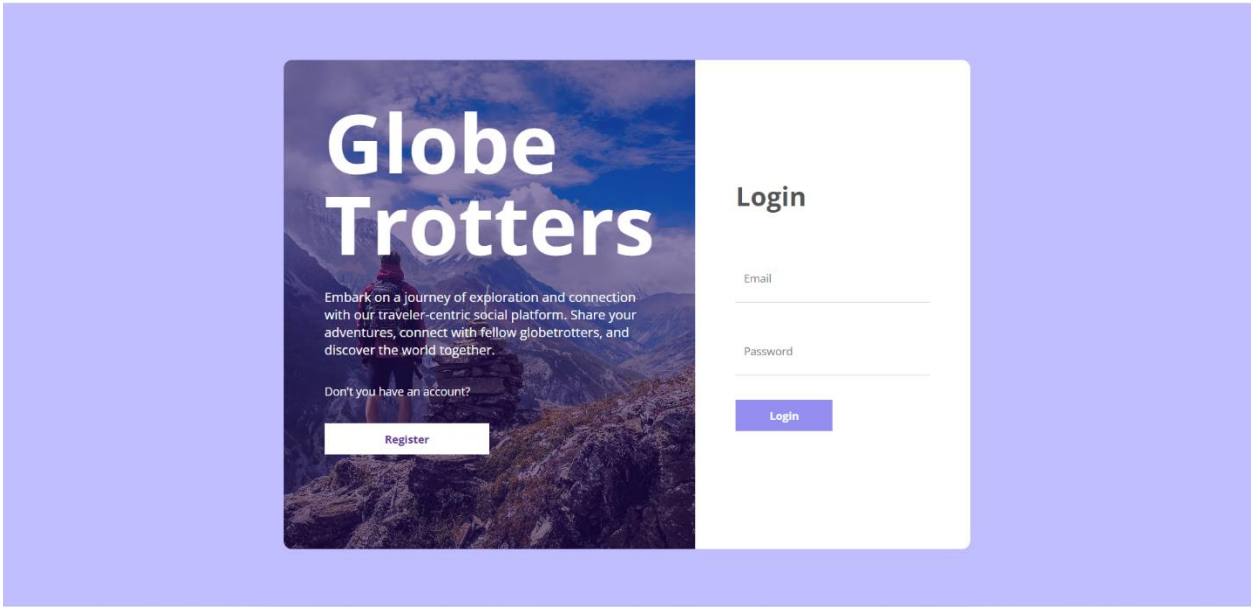


Figure 4.2 Login

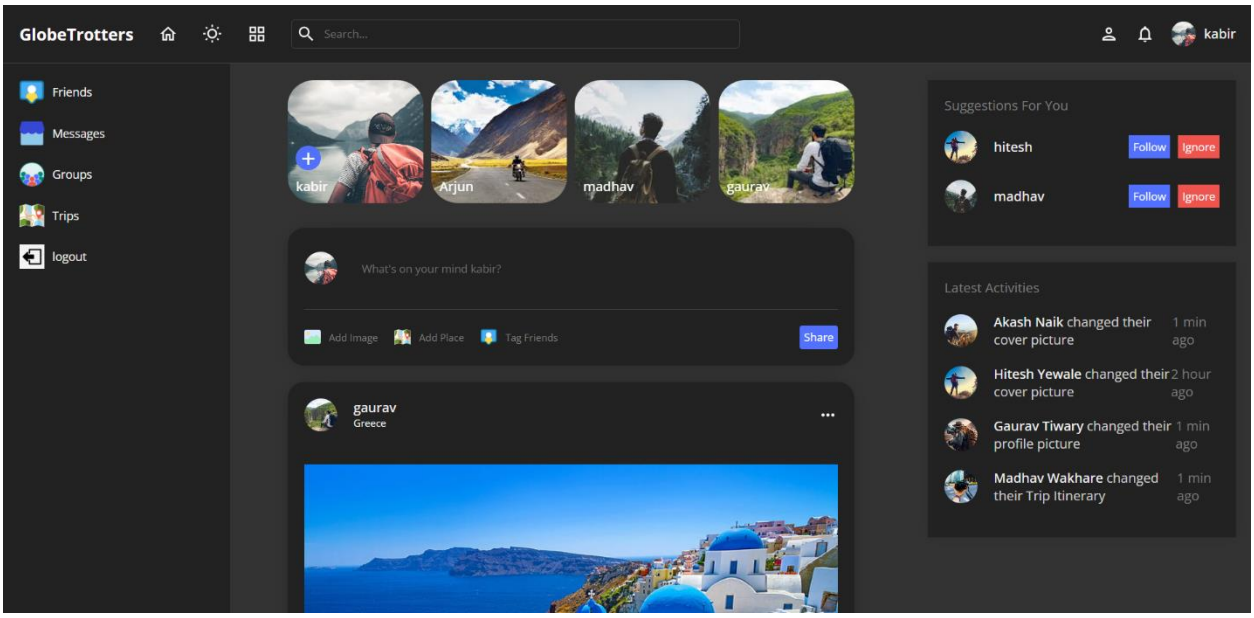


Figure 4.3 Home

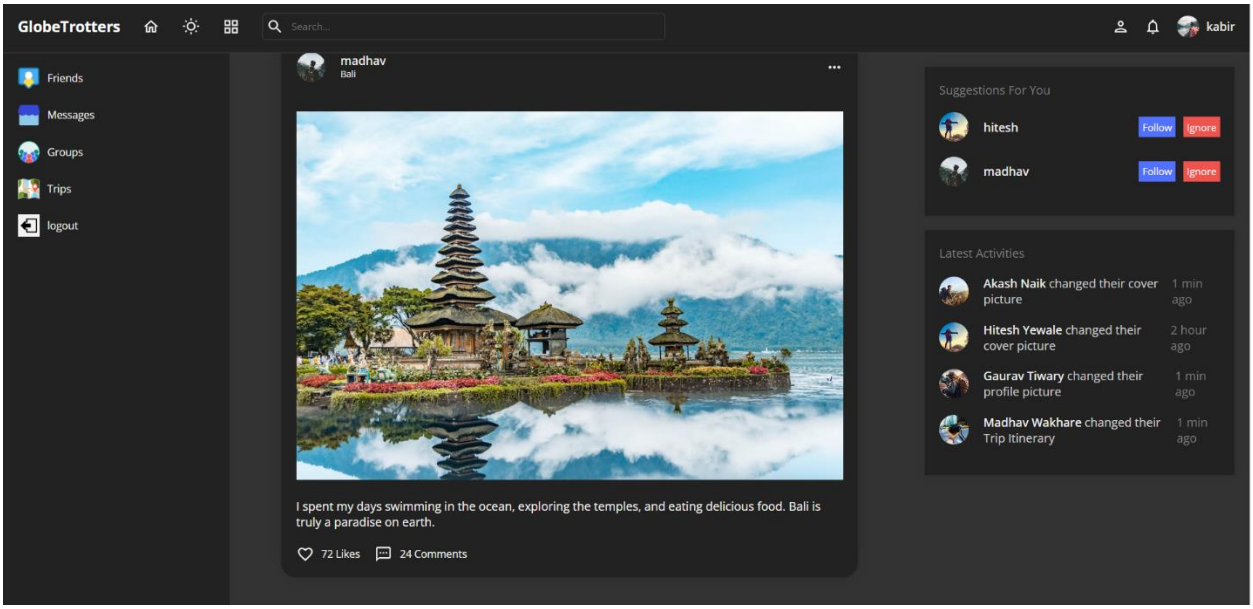


Figure 4.4 Post

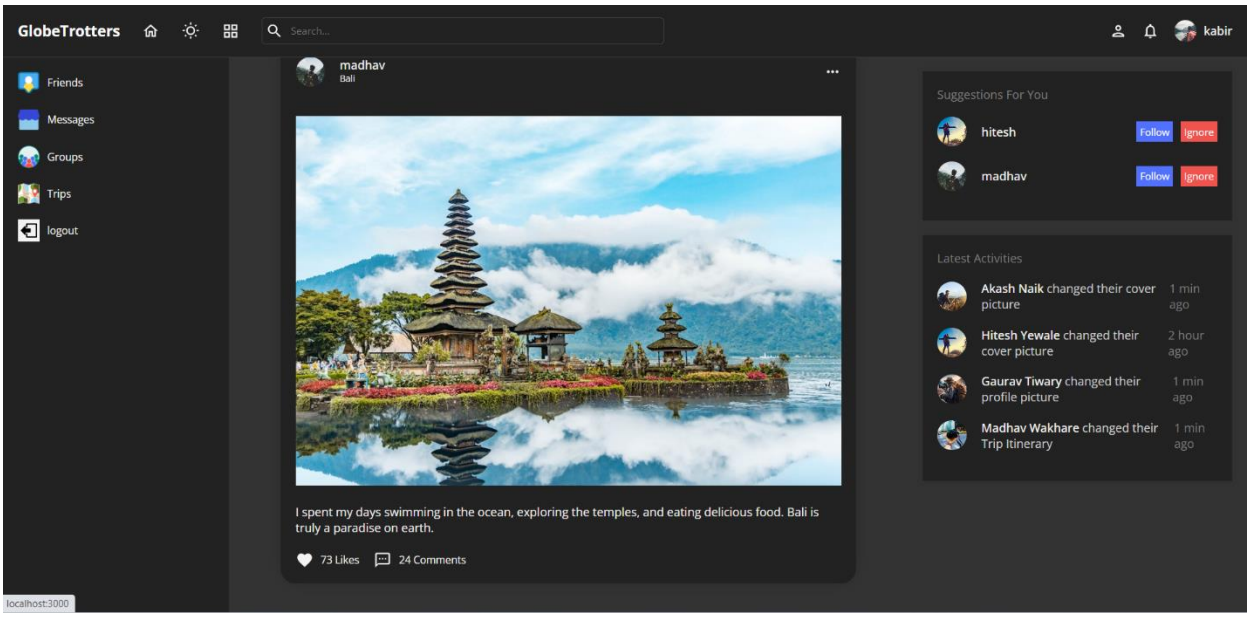


Figure 4.5 post like

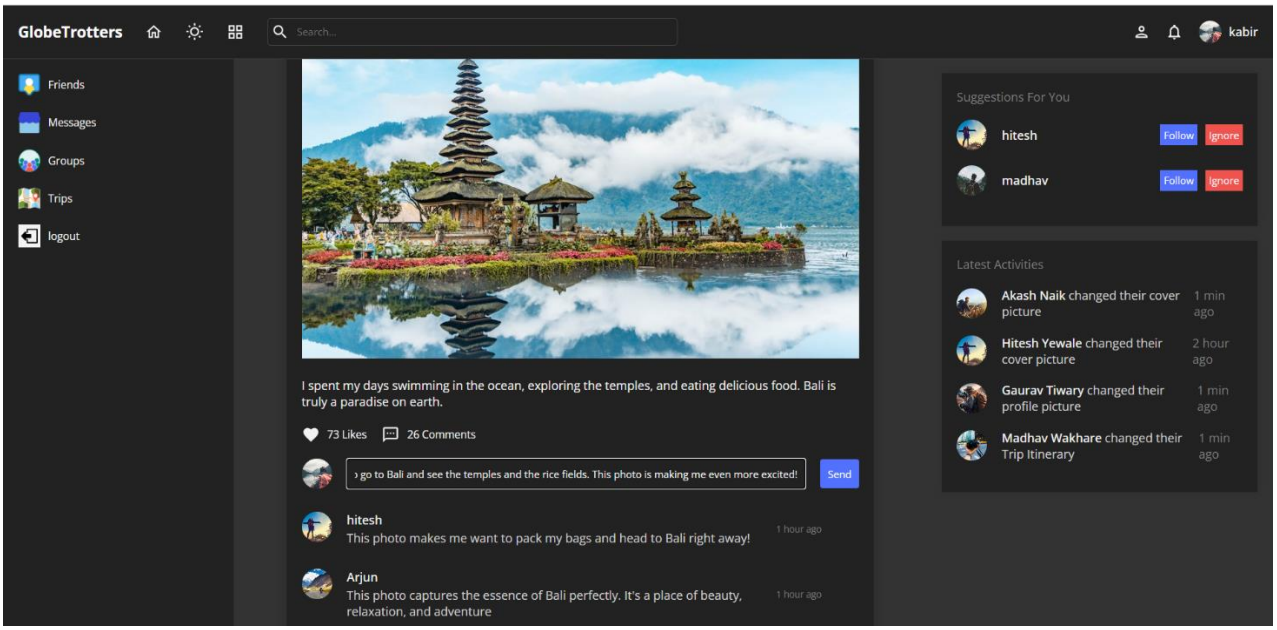


Figure 4.6 comment

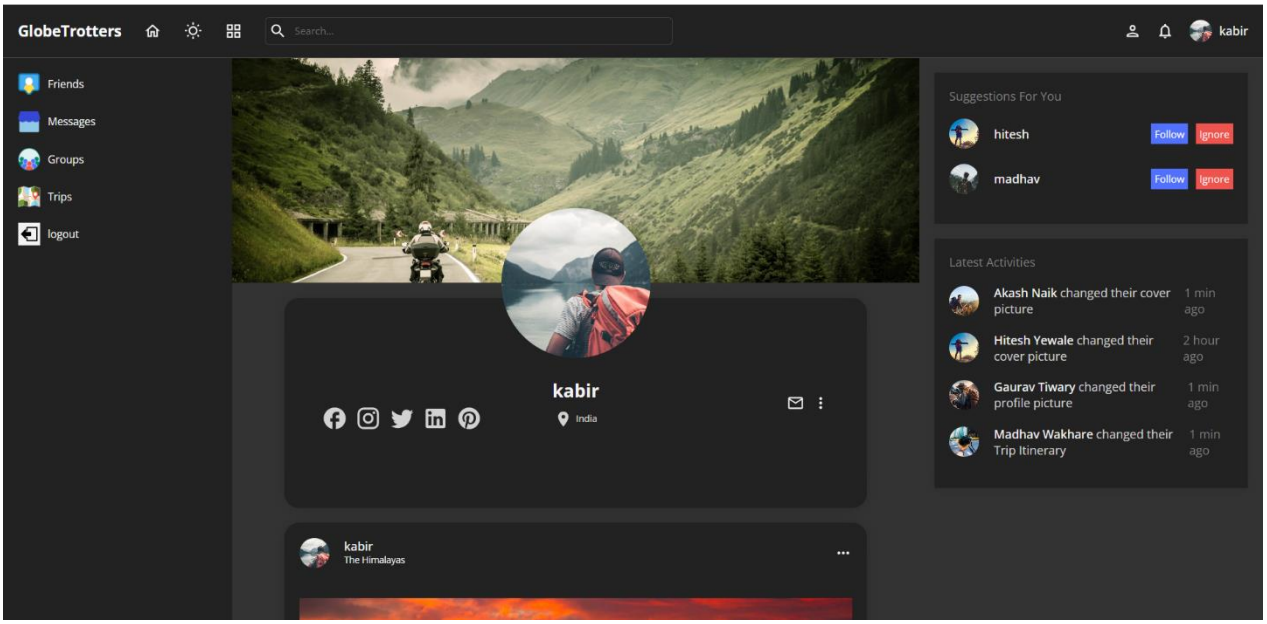


Figure 4.7 our profile

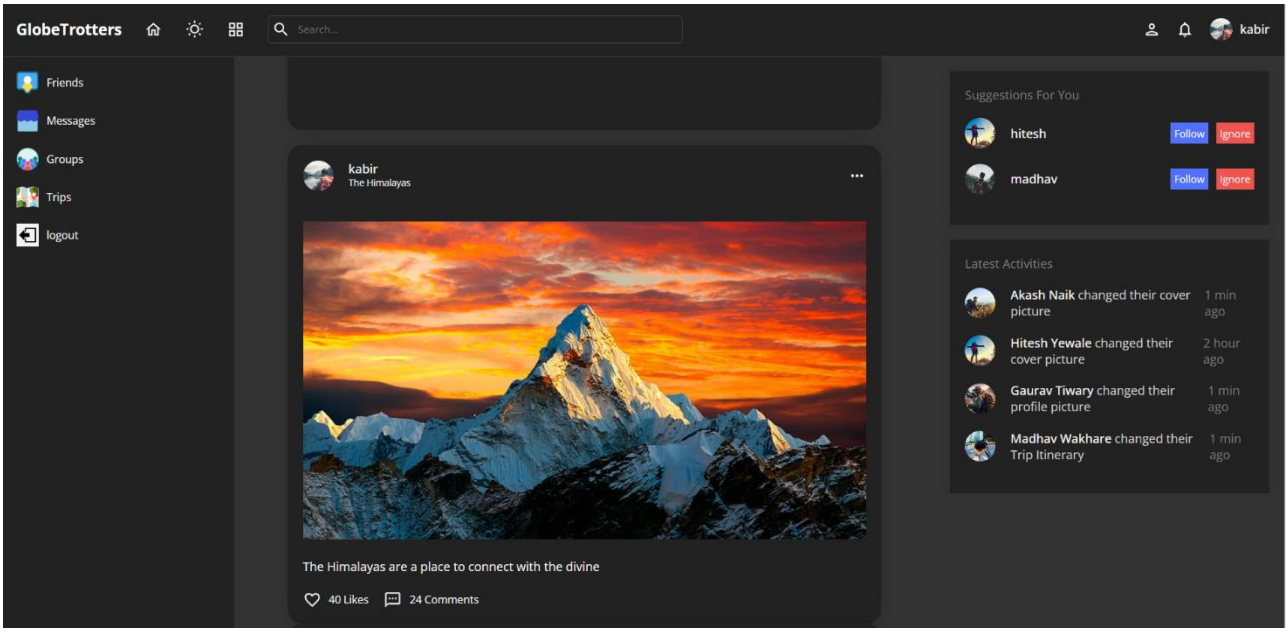


Figure 4.8 our profile field

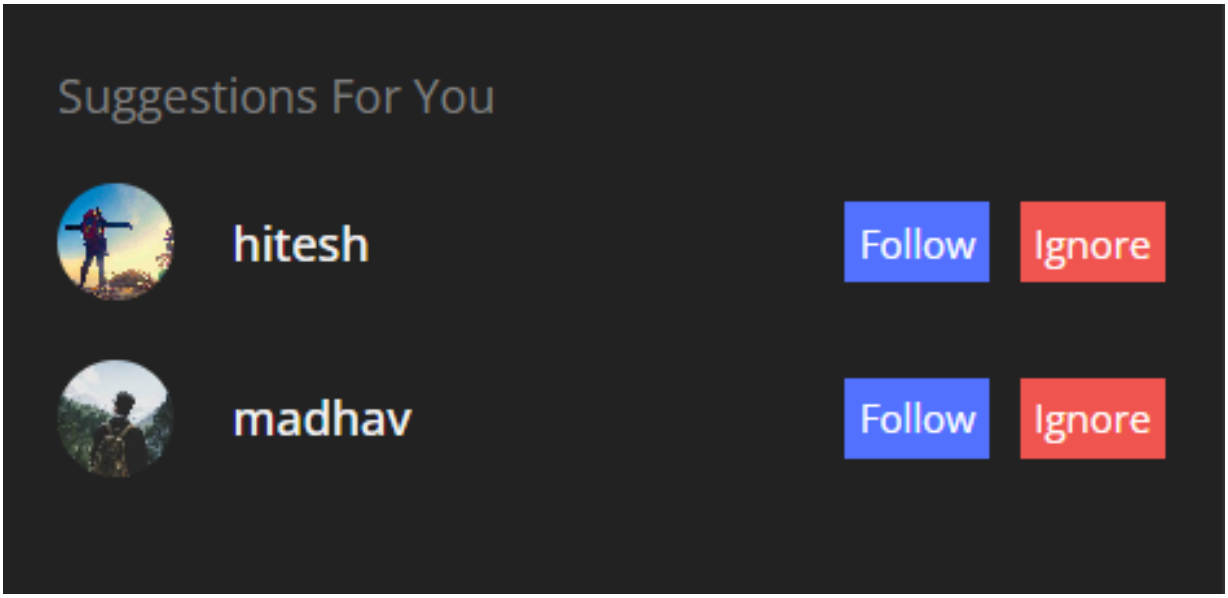


Figure 4.9 Suggestion

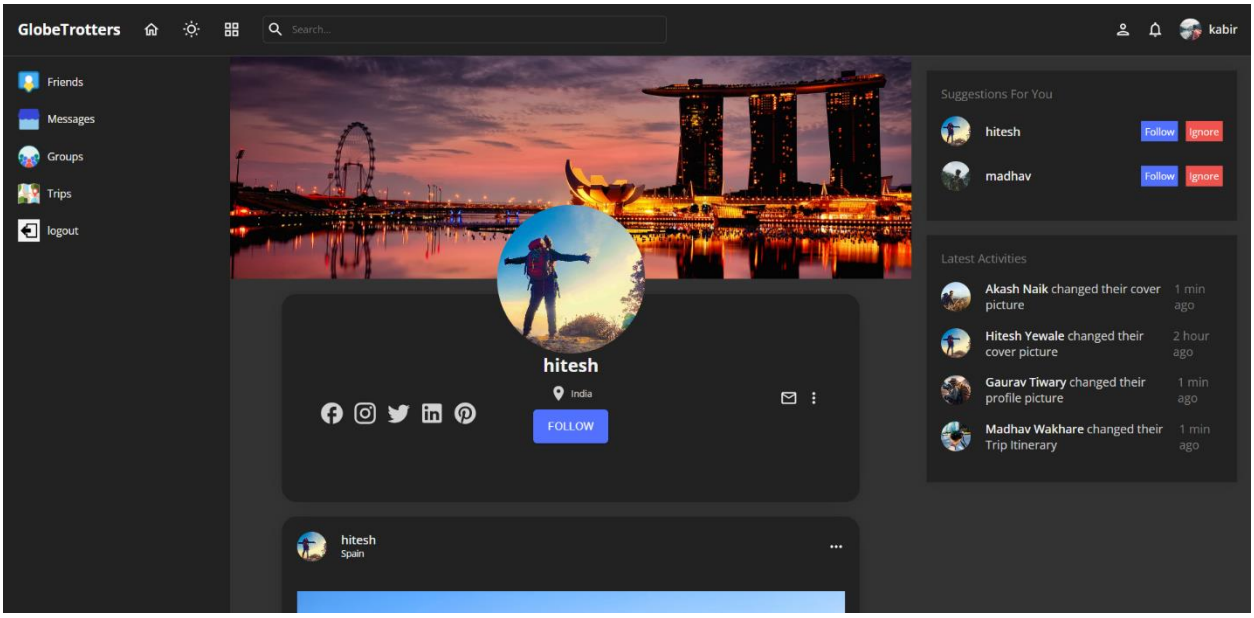


Figure 4.10 Other Users Profile

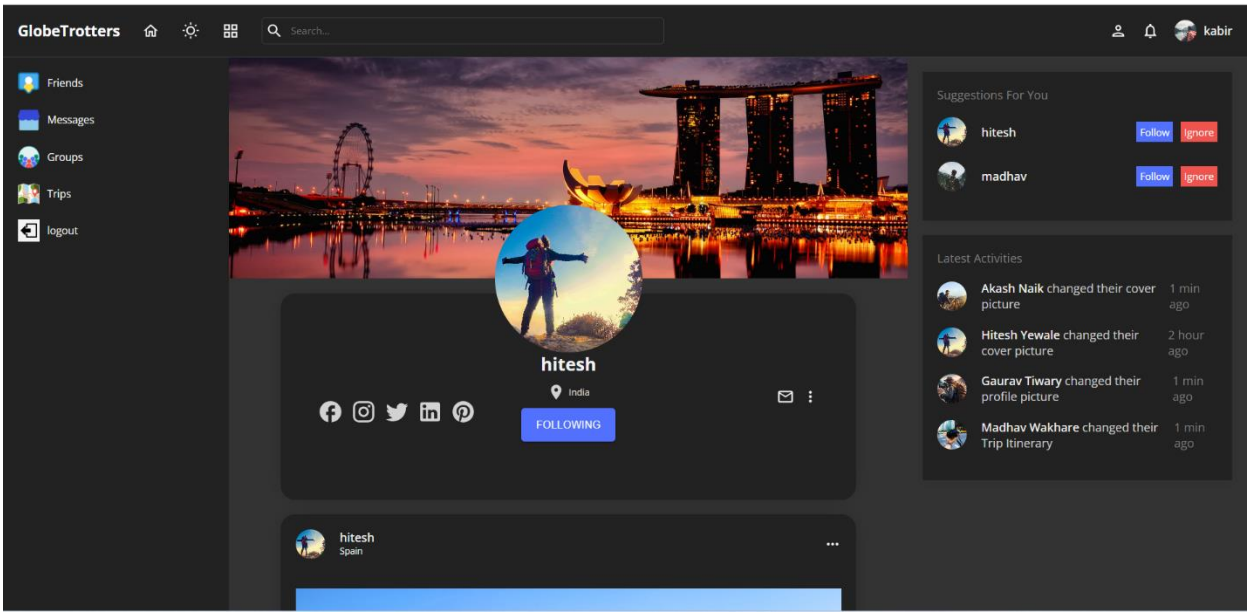


Figure 4.11 Followed

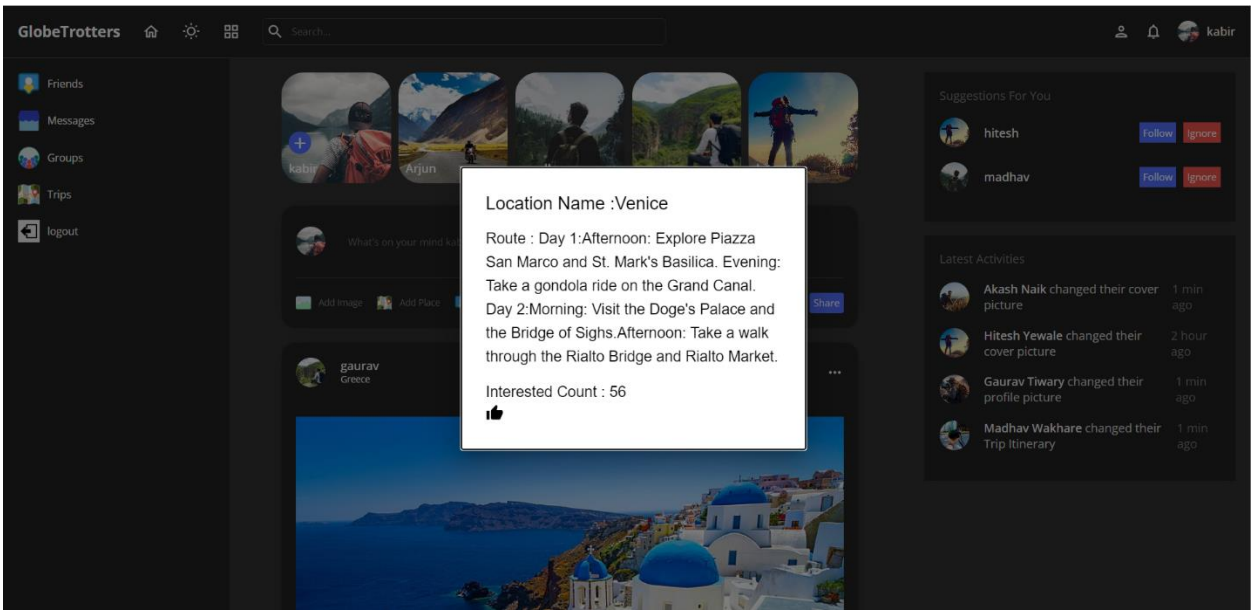


Figure 4.12 Trip

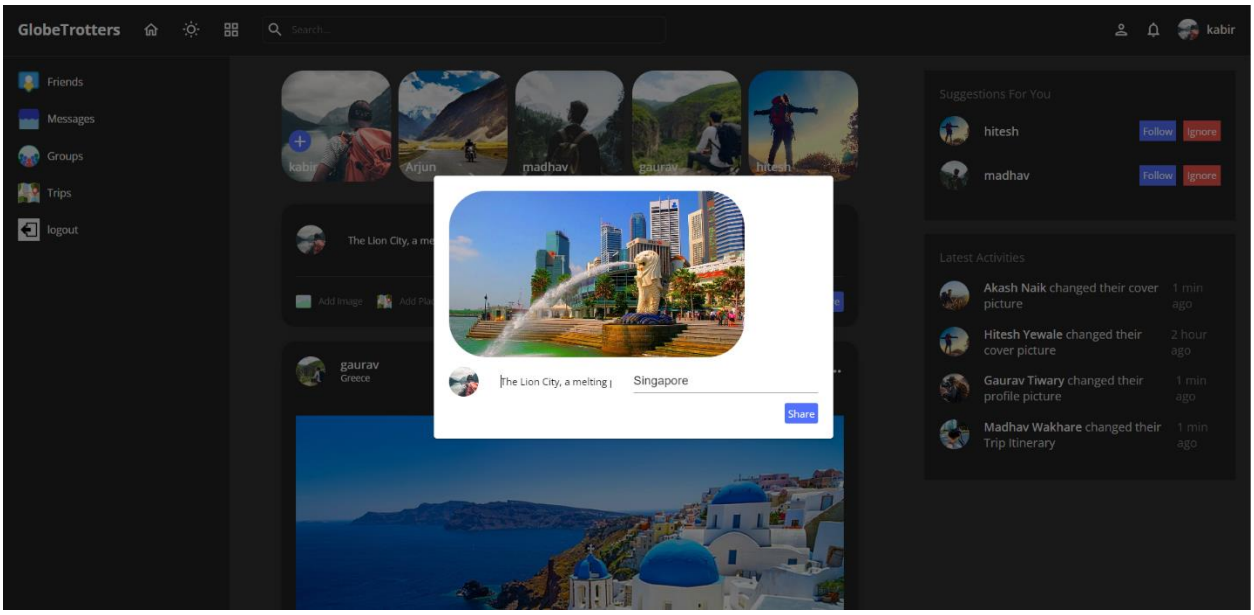


Figure 4.13 Post Share

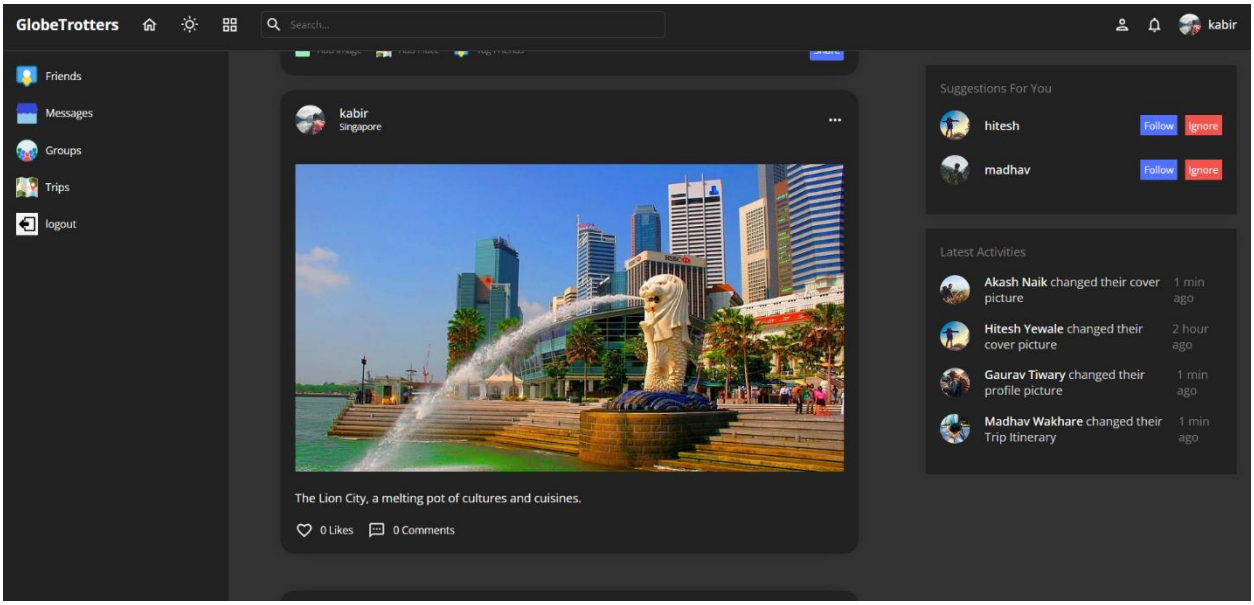


Figure 4.14 Post Shared

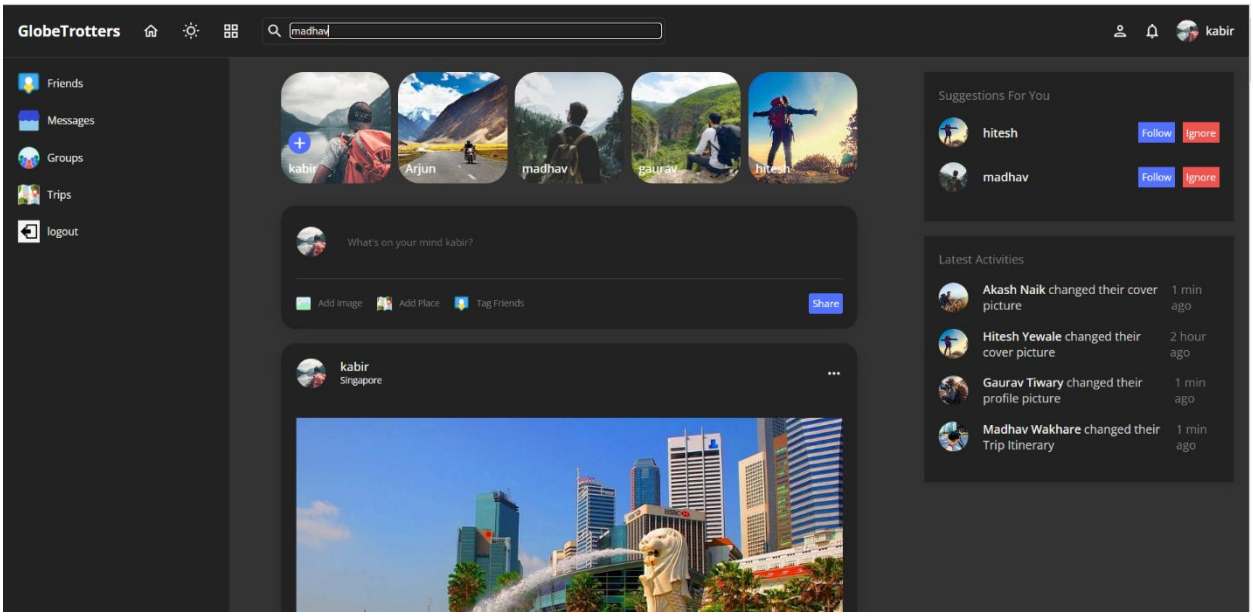


Figure 4.15 Search

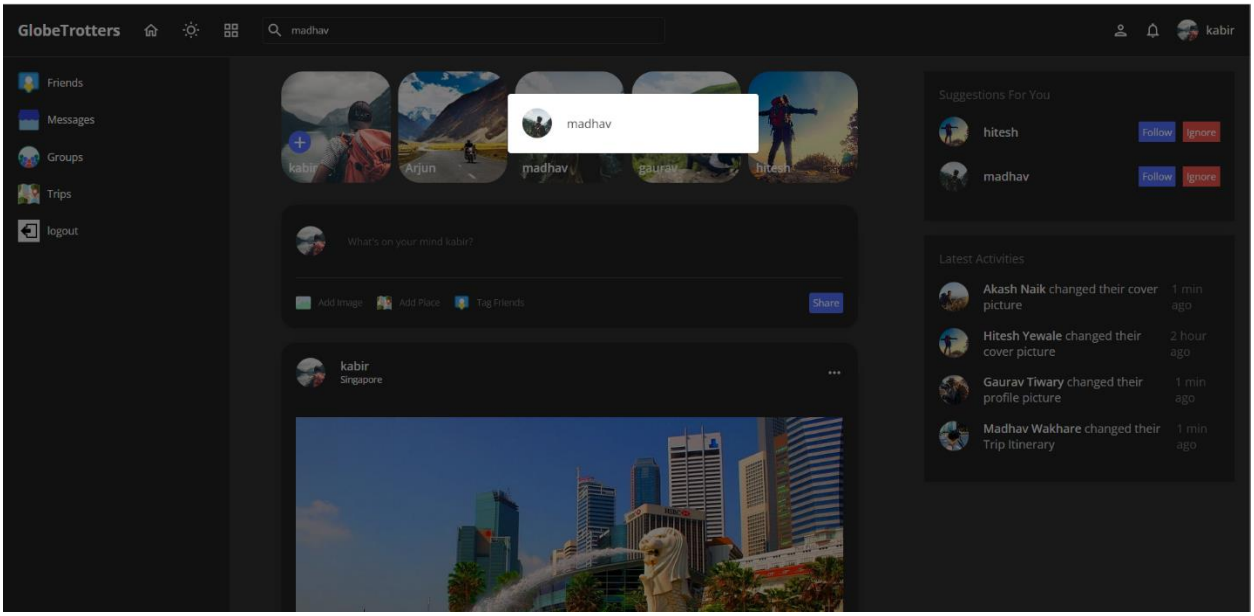


Figure 4.16 Search Result

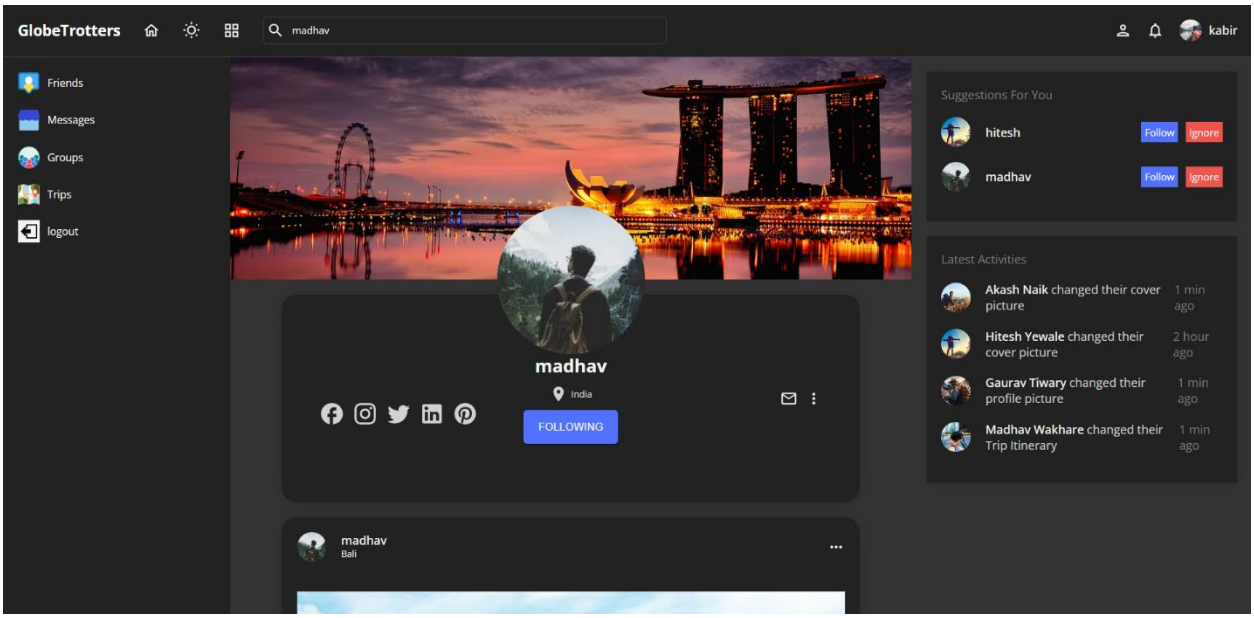


Figure 4.17 Searched Profile

5. CODING STANDARDS IMPLEMENTED

Naming and Capitalization

- Below summarizes the naming recommendations for identifiers in Pascal casing is used mainly (i.e. capitalize first letter of each word) with camel casing (capitalize each word except for the first one) being used in certain circumstances.

Identifier	Case	Examples	Additional Notes
Class	Pascal	Person, BankVault, SMSMessage, Dept	Class names should be based on "objects" or "real things" and should generally be nouns. No '_' signs allowed. Do not use type prefixes like 'C' for class.
Method	Camel	getDetails, updateStore	Methods should use verbs or verb phrases.
Parameter	Camel	personName, bankCode	Use descriptive parameter names. Parameter names should be descriptive enough that the name of the parameter and its type can be used to determine its meaning in most scenarios.
Interface	Pascal with "I" prefix	Disposable	Do not use the '_' sign
Property	Pascal	ForeColor, BackColor	Use a noun or noun phrase to name properties.
Associated private member variable	_camelCase	_foreColor, _backColor	Use underscore camel casing for the private member variables
Exception Class	Pascal with "Exception" suffix	WebException,	

Comments

- Comment each type, each non-public type member and each declaration
- Use end-line comments only on variable lines. End-line comments are comments that follow code on a single line.
- Separate comments from comments delimiters (apostrophe) or // with one space
- Begin the comment text with period
- End the comment with a period
- Explain the code. Do not repeat it

6. TESTING

6.1 System Testing

System testing is a critical element of quality assurance and represents the ultimate review of analysis, design and coding. Test case design focuses on a set of techniques for the creation of test because that meet overall testing objective. When a system is developed it is hoped that it performs properly. The main purpose of testing an information system is to find the errors and correct them. The scope of system testing should include both manual and computerized operations. System testing is comprehensive evaluation of the programs, manual procedures, computer operations and controls.

System testing is the process of checking whether the developed system is working according to the objective and requirement. All testing is to be conducted in accordance to the test conditions specified earlier. This will ensure that the test coverage meets the requirements and that testing is done in a systematic manner.

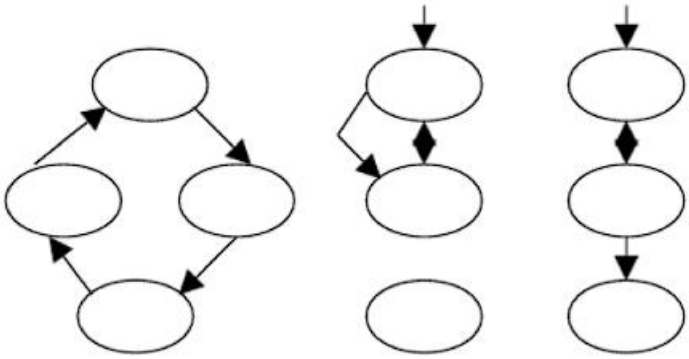
The process of analyzing the software item to detect the differences between existing or required condition and evaluate the features of the software items. The thorough testing of the system before release of the software needs to be done vide the various test cases and modes so that the software becomes devoid of bugs and uses minimum space requirements as well as minimum time to perform. The test cases were selected beforehand with expected results defined and actual results recorded for comparison. The selection of test cases is done vide "White Box Testing" technique to check the internal programming logic and efficiency and vide "Black Box Testing" technique to check software requirement fulfilment with intension of finding maximum number of errors with minimum effort and time. Although test cases are a design by considering the cyclomatic complexity, conditional test, still the software code is not in its optional form, as all other possible alternative parts in the software are not considered. At the integration level, the software will be passing to the third-party tests which would further enhance the software optimality and efficiency.

6.2 Test Data Implementation

The quality and standardization of the software application package depends truly on the

various predefined testing norms and on the performances of the software over those norms. There are various standards existing in the software industry the engineered end product strives to achieve viz. ISO 9002 SEI CMM Level5 etc. These standards are achieved only when the concerned software fulfils the tests as per the respective testing norms predefined in them vide the various test cases and parameters using the CASE topologies. Generally, software is tested both on a stand-alone mode as well after integrating all the modules in the system vide deferent available testing methods/norms.

The following Flow Graph methodology was used while testing the software:



6.1 Flow Chart

Here each circle represents one or more non branching procedural language or source code statements in Flow Graph. While performing Condition Testing Domain Testing methodology was selected. While performing Loop Testing simple loops, concatenated loops, nested and unstructured loops were tested thoroughly.

- **Test Characters:**
    1. A good test has a high probability of finding an error.
    2. A good test is not redundant.
    3. A good test should be "best of breed".
    4. A good test should be neither too simple nor too complex.

6.3 Black Box Testing

The method of Black Box Testing is used by the software engineer to derive the required results of the test cases:

1. Black Box Testing alludes to test that are conducted at the software interface.
2. A Black Box Test examines some fundamental aspect of a system with little regard for the internal logic structure of the software.
3. A limited number of important logical paths can be selected and exercised.
4. Important data structure can be probed for validity.

Black box testing was performed to find errors in the following categories:

- Incorrect or missing functions
- Graphics error.
- Errors in data in binary format.
- Error in data in integer format.
- File error.
- Pointer error.
- Memory access error.
- Variable error.
- Performance error

6.4 White Box Testing

White Box Testing is sometimes called Glass Box Testing. Using White Box Testing methods, the software engineer can derive the following test cases:

1. Guarantee that all independent paths within a module have been exercised at least once.
2. Exercise all logical decisions on their true and false sides.
3. Execute all loops at their boundaries and within their operational bounds.
4. Exercise internal data structures to ensure the validity.50

In White Box Testing efforts were made to handle the following:

- Number of input parameters equal to number of arguments.

- Parameters and arguments attributes match.
- Number of arguments transmitted is called modules equal to attributes of parameters.
- Unit system of argument transmitted is called modules equal unit system of parameter.
- Number of attributes and order of arguments to build in functions correct.
- Any references to parameters not associated to build in functions correct.
- Input only arguments altered.
- Global variable definition consistent across module.
- Files attributes correct.
- Format specifications matches I/O specification.
- Files opened before use.
- File closed while working is going on.
- I/O errors handled.
- Any textual errors in output information.

6.5 Unit Testing

The unit testing is performed to test the validity of the individual units. This is done in the coding phase with the interactive testing. Thus, it itself constitutes a majority of functionality test for each logical unit.

6.6 Integrity Testing

When all the development of all the units or modules is completed and integrated the integrity test phase is started. In this phase the interface between the modules are tested. This phase basically verifies whether inter module exchange of information and events are as per required system behavior.

6.7 Validation Testing

Tests were performed to find conformity with the requirements. Plans and procedures were designed to ensure that all functional requirements are satisfied. The software was alpha-tested. There are two goals in preparing test plans. Firstly, a properly detailed test plan demonstrates that the program specifications are understood completely. Secondly, the test plan is used during program testing to prove the correctness of the program.

6.8 Test Cases

- **Login**

Sr. No	Input Value	Test Case	Conditional being checked	Result
1	Email	Empty	Please Enter valid Email Id	Successful
2	Email	Already Exists or not	Email Id should be unique	Successful
3	Password	Empty	Please Enter valid Password	Successful
4	Password	If wrong Password	Please Enter valid Password	Successful

- **Registration**

Sr. No	Input Value	Test Case	Conditional being checked	Result
1	Name	Empty	It must not be empty	Successful
2	User Name	Empty	It must be valid User name/ It must be unique	Successful
3	Email	Empty	It must be valid Email Id/ It must be unique	Successful
4	Password	Empty	Please Enter valid Password	Successful
5	Date of Birth	Empty	It must not be empty/ It must be past date	Successful
6	Gender	Select	Please select gender	Successful
7	Mobile	Empty	It must not be empty	Successful

- **Edit Profile**

Sr. No	Input Value	Test Case	Conditional being checked	Result
1	Name	Empty	Name can not be empty	Successful

2	User Name	Empty	It must be valid User name/ It must be unique	Successful
3	Password	Empty	Please Enter valid Password	Successful
4	Date of Birth	Empty	It must not be empty/ It must be past date	Successful
5	Gender	Select	Please select gender	Successful
6	Mobile	Empty	It must not be empty	Successful

- **Post**

Sr. No	Input Value	Test Case	Conditional being checked	Result
1	Title	Empty	Image Title can't be empty	Successful
2	Add Image	Browse	Please browse image	Successful
3	Description/ Caption	Empty	Please Enter valid caption	Successful
4	Location	Empty	It must not be empty	Successful
5	Delete Image	Select	Please select image to delete	Successful

7. CONCLUSION

While developing the system a conscious effort has been made to create and develop a software package, making use of available tools, techniques and resources that would generate a proper system for ONLINE SOCIAL SITE/NETWORKING [Globe Trotters].

While making the system, an eye has been kept on making it as user-friendly and secure. As such one may hope that the system will be acceptable to any user who wants to share his travelling experience and make new connections. As in case of any system development process where there are a number of short comings, there have been some shortcomings in the development of this system also.

There are some of the areas of improvement which couldn't be implemented due to time constraints. One such feature was online chat where members can chat with his friends through this website.

8. REFERENCES

1. Indira Jain Social networking trends New Delhi: The Times of India
2. Overview of ASP.NET Core | Microsoft Learn
3. Getting Started – React (reactjs.org)
4. W3Schools Online Web Tutorials
5. GeeksforGeeks | A computer science portal for geeks
6. http://www.wikipedia.org