**Project Name: Vacation Spot Development & Property Renting.**

**Project Member:**

**Prasad Shinde 210543181122**

**Vikas Kumbhar 210543181049**

**Suraj Bavdekar 210543181121**

**Rushikesh Kadam 210543181033**

**Abstract:**  
 “Our Project is based on PDCA which is stands for **Plan Do Check Act**. The concept of our project is simple, we are here to give the platform to **Plan** the vacation **Do** take a single minute to decide vacation spot **Check** the nearby property for stay **Act** without worries & enjoy the vacation.”

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality for performing commercial transactions and commercial development over the web. It is reasonable to say that the process of searching everything on the web is becoming common place.

The objective of this project is to develop rural side vacation spots and give an opportunity to the people to level up their livelihood by renting their property for visitors to stay. All this are happening on single platform by just one web site/ application where visitors or the people who want to make their vacation worriless and make stay at their vacation spot by renting nearby personal property, it can be done from the comfort of home through the Internet.

Our virtual platform allows the client to browse the catalogue and select vacation spots of their interest and choose nearby property for stay. The client can directly contact with the property owner and also check for the reviews provided by other users. Usually, the client will be asked to fill form with the Aadhar no. for confirmation of property safety and booking. If someone wants to add their property for rent 'Add property' option will be there.

**Implementation Technologies:**

The system is implemented using **MERN** stack approach.

1. **MongoDB**

MongoDB’s document data model naturally supports JSON and its expressive query language is simple for developers to learn and use. Functionality such as automatic failover, horizontal scaling, and the ability to assign data to a location are built-in.

**1.1 Features of MongoDB:**

**1. MongoDB uses BSON format**

One of the key features of MongoDB is that it uses BSON format. BSON is a JSON-like storage format. BSON stands for Binary JSON which is a binary-encoded serialization of JSON-like documents that MongoDB uses when storing documents in collections

**2. MongoDB Sharding**

MongoDB is a schema-less database (written in C++) because of which is much more flexible than traditional database tables. The benefit is the lack of setup and the reduced friction with OOP. So, in order to save an object, you just have to serialize it to JSON and send it to MongoDB.

**3. Capped Collections**

MongoDB supports fixed-size collections called capped collections. This type of collection maintains insertion order.

**4. MongoDB Indexing**

Indexes are created to improve the performance of searches. The good thing is that any field in a MongoDB document can be indexed with primary and secondary indices.

**1.2 Advantages of MongoDB:**

1. **MongoDB is schema less**. It is a document database in which one collection holds different documents.
2. There may be **difference between number of fields, content and size of the document** from one to other.
3. **Structure of a single object is clear** in MongoDB.
4. There are **no complex joins** in MongoDB.
5. MongoDB provides the **facility of deep query** because it supports a powerful dynamic query on documents.
6. It is very **easy to scale**.
7. **Express**

The next level down is the Express.js server-side framework, running inside a Node.js server. Express.js bills itself as a “fast, minimalist web framework for Node.js,” and that is indeed exactly what it is. Express.js has powerful models for URL routing (matching an incoming URL with a server function), and handling HTTP requests and responses.

1. **React**

The top tier of the MERN stack is React.js, the declarative JavaScript framework for creating dynamic client-side applications in HTML. React lets you build up complex interfaces through simple Components, connect them to data on your backend server, and render them as HTML.

**Features of REACT:**

* **Virtual DOM**

Virtual DOM is one of the chief characteristics that facilitate fast and flexible application development using React.

* **JSX**

JSX stands for [**JavaScript XML**](https://www.geeksforgeeks.org/reactjs-introduction-jsx/). It is a markup syntax that is very similar to HTML and is used to describe the appearance of an application’s UI.

JSX is one of the best features that React has to offer. It makes the syntax, used by developers to create React components, almost identical to the HTML they will inject in the web page.

* **One-way data binding**

One of the major reasons why you should choose Reactjs for your next project is its one-way data flow. Reactjs uses a unidirectional data flow. Meaning, the developers cannot edit any component directly. They have to utilize the callback function in order to make changes in the components. This process is called one-way data binding.

1. **Node.js**

Node.js (Node) is an open source development platform for executing [JavaScript](https://www.theserverside.com/definition/JavaScript) code server-side. Node is useful for developing applications that require a persistent connection from the [browser](https://whatis.techtarget.com/definition/browser) to the server and is often used for [real-time applications](https://searchunifiedcommunications.techtarget.com/definition/real-time-application-RTA) such as chat, news feeds and web push notifications.

Node.js is intended to run on a dedicated [HTTP](https://whatis.techtarget.com/definition/HTTP-Hypertext-Transfer-Protocol) server and to employ a single thread with one process at a time. Node.js applications are [event-based](https://searchapparchitecture.techtarget.com/definition/event-driven-architecture-EDA) and run [asynchronously](https://techtarget.com/searchnetworking/definition/asynchronous).

**SDLC Model Used**

**Iterative Model**

In this Model, you can start with some of the software specifications and develop the first version of the software. After the first version if there is a need to change the software, then a new version of the software is created with a new iteration. Every release of the Iterative Model finishes in an exact and fixed period that is called iteration.

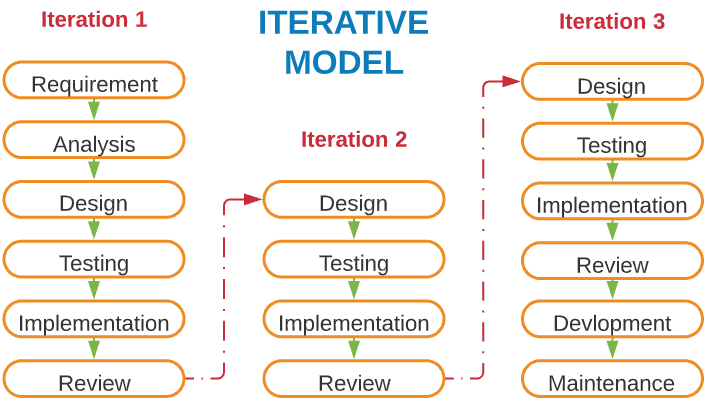
****

Figure 1: Iterative Model

**Advantage(Pros) of Iterative Model:**

1. Testing and debugging during smaller iteration is easy.
2. A Parallel development can plan.
3. It is easily acceptable to ever-changing needs of the project.
4. Risks are identified and resolved during iteration.

**Disadvantage(Cons) of Iterative Model:**

1. Design can be changed again and again because of imperfect requirements.
2. Requirement changes can cause over budget.
3. Project completion date not confirmed because of changing requirements.

**Hardware and Software Requirements (Minimum):**

**Hardware:**

1. Intel i3 processor 3rd generation or later.

2. 2 GB ram.

3. Windows 7 Home edition or later.

4. Data Connection 200 kbps.

**Software:**

1. Visual Studio
2. MongoDB (Atlas)
3. Google Chrome

**ER Diagram:**

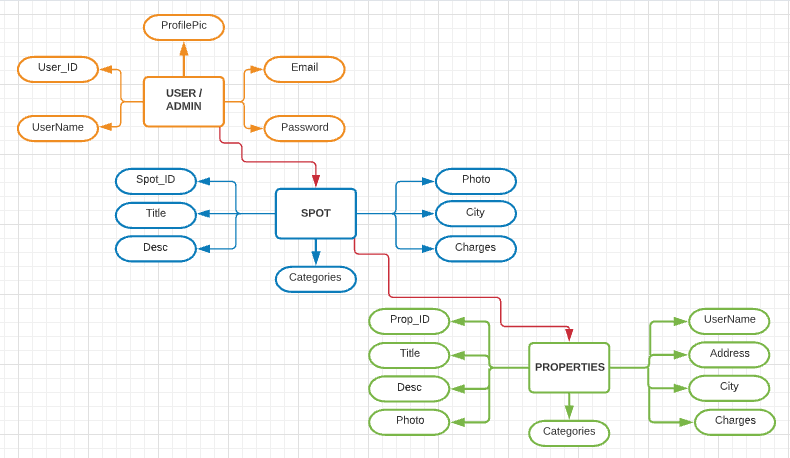


Figure 2: ER Diagram

**Table Structures:**

1. **Table name: Properties Table**

**Column name Type**

Prop\_Id int (03) NO PRI auto\_increment

Title varchar(50) YES UNI

Desc varchar(50) YES

Photo varchar(500) YES

Username varchar(50) YES

Address varchar(50) YES

City varchar(50) YES

Charges varchar(50) YES

Categories varchar(50) YES

1. **Table name: Spot Table**

**Column name Type**

Spot\_Id int (03) NO PRI auto\_increment

Title varchar(50) YES UNI

Desc varchar(50) YES

Photo varchar(500) YES

City varchar(50) YES

Categories varchar(50) YES

1. **Table name: User**

**Column name Type**

User\_Id int (03) NO PRI auto\_increment

UserName varchar(50) YES

Email varchar(50) YES UNI

Password varchar(50) YES

ProfilePic varchar(500) YES

**UML Diagrams:**

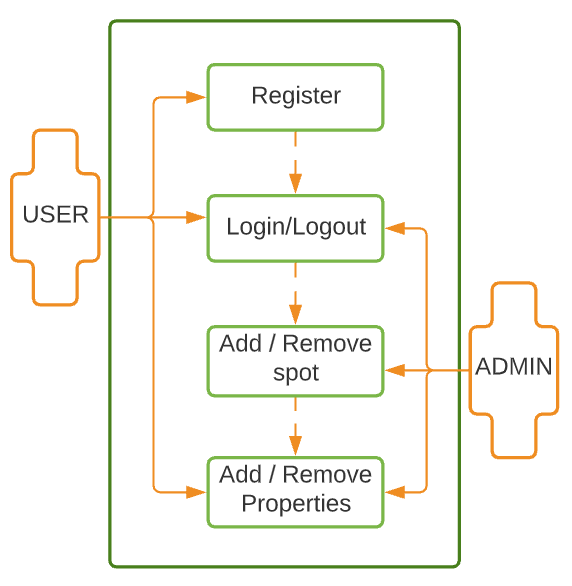


Figure 3: Use Case

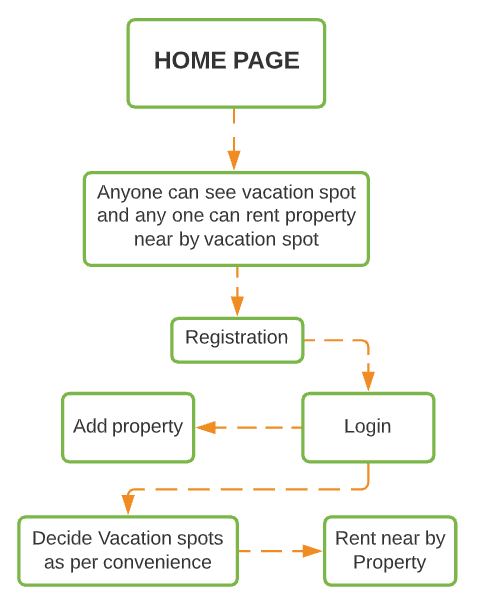


Figure 4: Sequence Diagram

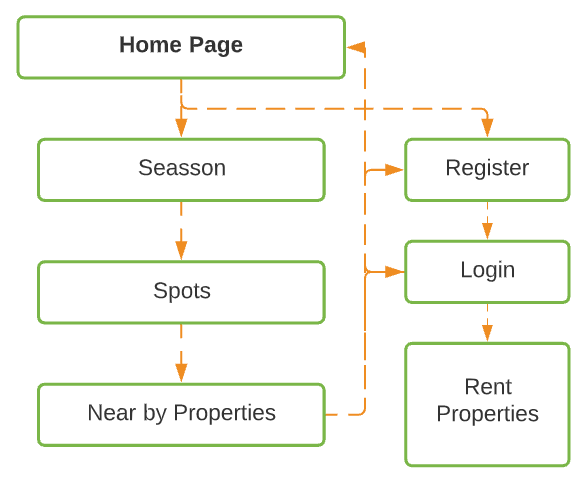
­­ 

Figure 5: Component Diagram

**End to End Flow of Application:**

**User:**

* 1. User will login to the portal or will have to register if he is not a registered user.
  2. After registration User will login and Dashboard page will be displayed to him which will display the Vacations spots categories by session’s (Summer, Winter, Monsoon).
  3. From that page User can see all the spots and nearby properties
  4. But for rent property they must have to login
  5. User can also add their properties for renting nearby the vacation spots.

**Admin:**

1. Admin will login as Admin and will be able to add new or delete spots or properties.
2. Admin login with all the authorities.

**Future Scope:**

User can directly do the payment to as he/she booked the property for renting as per property owner decided by online payment method.

Top most visited spots and properties are sorted as per their review and ratings

**Thank You!**