EXPERIMENT 3a

NAME: Shivam Pawar **UID: 2019230068**

NAME: Vishal Salvi **UID: 2019230069**

NAME: Shreyas Patel **UID: 2018130043**

CLASS: TE COMPS **BATCH: C**

DATE:

Aim: Understanding MVC architecture and Prototype design.

Theory:

What is MVC Framework?

The **Model-View-Controller (MVC)** framework is an architectural pattern that separates an application into three main logical components Model, View, and Controller. Hence the abbreviation MVC. Each architecture component is built to handle specific development aspect of an application. MVC separates the business logic and presentation layer from each other. It was traditionally used for desktop graphical user interfaces (GUIs). Nowadays, MVC architecture has become popular for designing web applications as well as mobile apps.

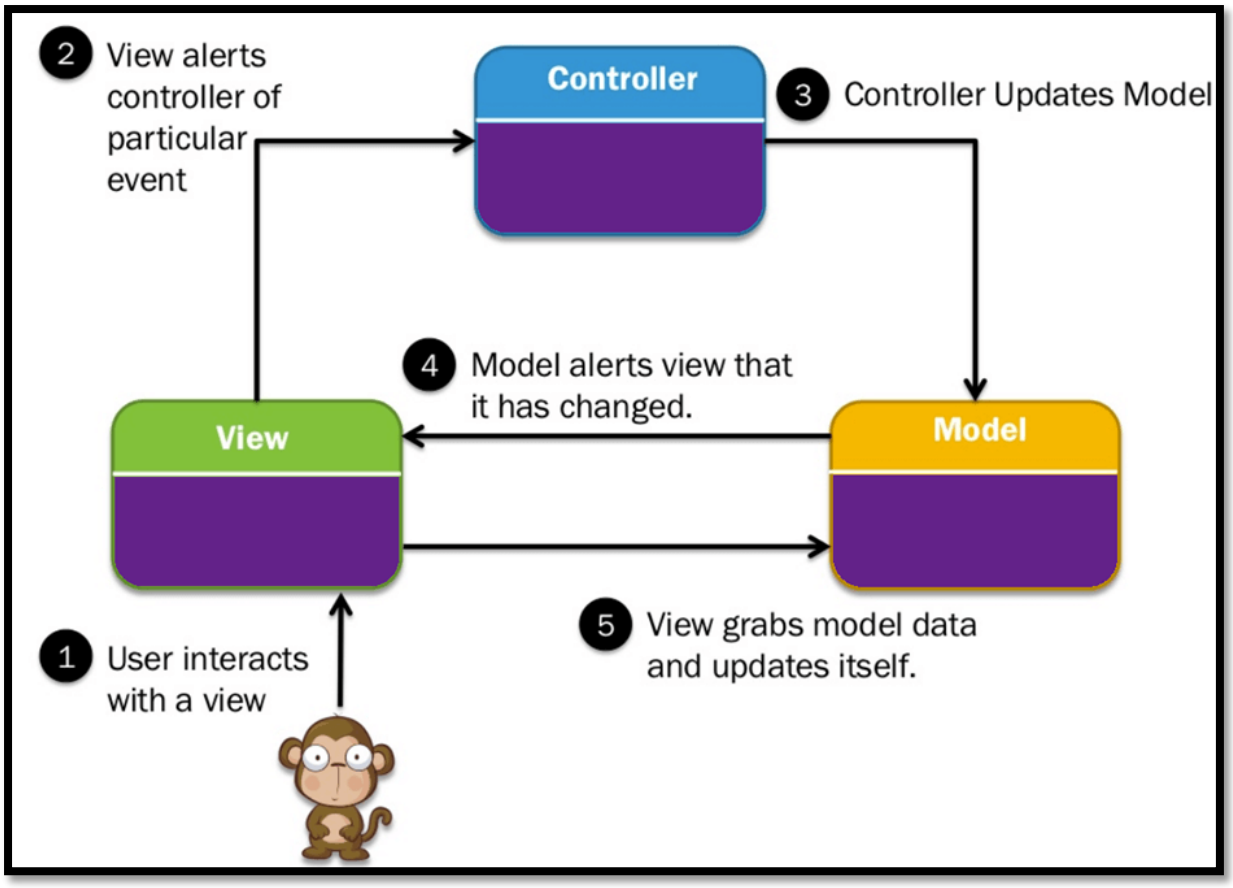
History of MVC

- MVC architecture first discussed in 1979 by Trygve Reenskaug
- MVC model was first introduced in 1987 in the Smalltalk programming language.
- MVC was first time accepted as a general concept, in a 1988 article
- In the recent time, MVC pattern is widely used in modern web applications

Features of MVC

- Easy and frictionless testability. Highly testable, extensible and pluggable framework
- Offers full control over your HTML as well as your URLs
- Leverage existing features provided by ASP.NET, JSP, Django, etc.
- Clear separation of logic: Model, View, Controller. Separation of application tasks viz. business logic, Ul logic, and input logic
- URL Routing for SEO Friendly URLs. Powerful URL- mapping for comprehensible and searchable URLs
- Supports for Test Driven Development (TDD)

MVC Architecture



Three important MVC the components are:

- Model: It includes all the data and its related logic
- View: Present data to the user or handles user interaction
- Controller: An interface between Model and View components

Let's see each other this component in detail:

View

A View is that part of the application that represents the presentation of data.

Views are created by the data collected from the model data. A view requests the model to give information so that it resents the output presentation to the user.

The view also represents the data from chats, diagrams, and table. For example, any customer view will include all the UI components like text boxes, drop downs, etc.

Controller

The Controller is that part of the application that handles the user interaction. The controller interprets the mouse and keyboard inputs from the user, informing model and the view to change as appropriate.

A Controller send's commands to the model to update its state(E.g., Saving a specific document). The controller also sends commands to its associated view to change the view's presentation (For example scrolling a particular document).

Model

The model component stores data and its related logic. It represents data that is being transferred between controller components or any other related business logic. For example, a Controller object will retrieve the customer info from the database. It manipulates data and send back to the database or use it to render the same data.

It responds to the request from the views and also responds to instructions from the controller to update itself. It is also the lowest level of the pattern which is responsible for maintaining data.

Advantages of MVC: Key Benefits

Here, are major benefits of using MVC architecture.

- Easy code maintenance easy to extend and grow
- MVC Model component can be tested separately from the user
- Easier support for new type of clients
- Development of the various components can be performed parallelly.
- It helps you to avoid complexity by dividing an application into the three units. Model, view, and controller
- It only uses a Front Controller pattern which process web application requests through a single controller.
- Offers the best support for test-driven development
- It works well for Web apps which are supported by large teams of web designers and developers.
- Provides clean separation of concerns(SoC).
- Search Engine Optimization (SEO) Friendly.
- All classed and objects are independent of each other so that you can test them separately.
- MVC allows logical grouping of related actions on a controller together.

Disadvantages of using MVC

- Difficult to read, change, to unit test, and reuse this model
- The framework navigation can some time complex as it introduces new layers of abstraction which requires users to adapt to the decomposition criteria of MVC.
- No formal validation support
- Increased complexity and Inefficiency of data
- The difficulty of using MVC with the modern user interface
- There is a need for multiple programmers to conduct parallel programming.
- Knowledge of multiple technologies is required.
- Maintenance of lots of codes in Controller

What is a Web Framework?

A web framework is a code library which helps you to build a flexible, scalable, and maintainable; dynamic website, web app, and web services. Different web frameworks are Zend for PHP, Ruby on Rails for Ruby, etc.

What is Django?

Django is a web development framework for Python which offers a standard method for fast and effective website development. It helps you to assists in building and maintaining quality web applications. It enables you to make the development process smooth and time-saving.

It is a high-level web framework which allows performing rapid development. The primary goal of this web framework is to create complex database-driven websites.

Why Django? Key Advantages

Here, are prime reasons for using Django:

- Django is easy to set up and run. It offers a variety of options to get started
- It provides a ready-to-use user interface for administrative activities
- It enables multilingual websites by using its built-in internationalization system
- Helps you to meet the massive traffic demands quickly
- Django is used to build all types of content management systems, social networks as well as scientific computing platforms.
- Django helps you to provide end-to-end application testing
- Helps you to document your API with an HTML output
- REST Framework has rich support for several authentication protocols
- Permissions and throttling policies
- It is widely used for rate limiting API requests from a single user.

Features of Django

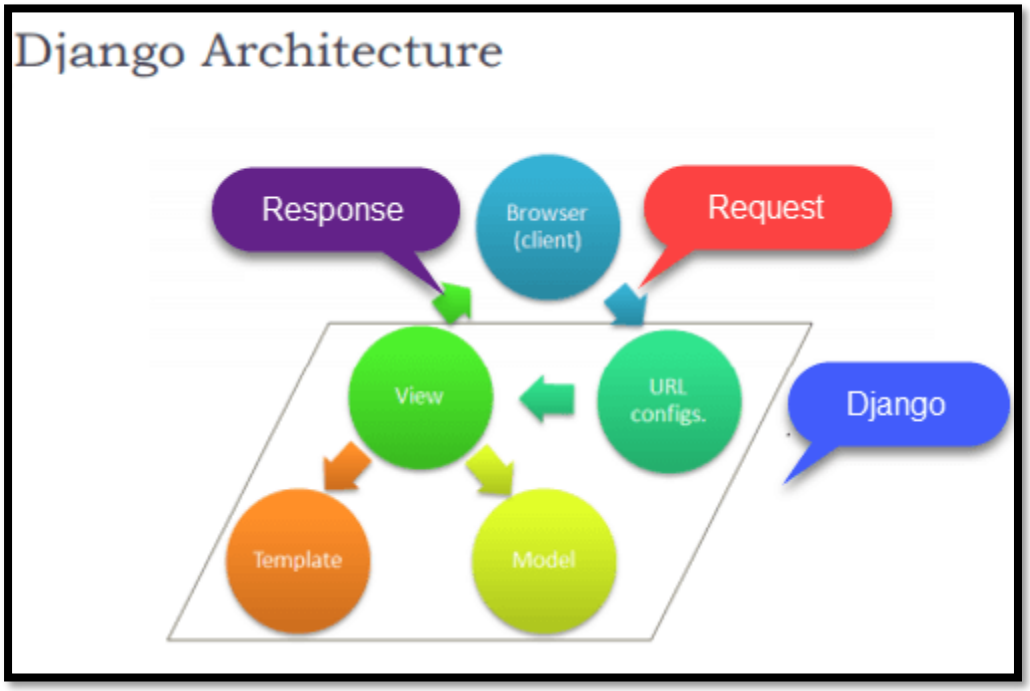
- Helps you to define patterns for the URLs in your application
- Simple but powerful URL system
- Built-in authentication system
- Object-oriented programming language database which offers best in class data storage and retrieval
- Automatic admin interface feature allows the functionality of adding, editing and deleting items. You can customize the admin panel as per your need.
- Cache framework comes with multiple cache mechanisms.

Characteristics of Django

- Loosely Coupled − Django helps you to make each element of its stack independent of the others.
- Less code - Ensures effective development

- Not repeated- Everything should be developed in precisely one place instead of repeating it again
- Fast development- Django's offers fast and reliable application development.
- Consistent design - Django maintains a clean design and makes it easy to follow the best web development practices.

Django Architecture: MVC and MVT



MVC Pattern:

When talking about applications which provide UI (web or desktop), we usually talk about MVC architecture. MVC pattern is based on Model, View, and Controller.

The **Model** defines the data structure and takes to care for querying the database.

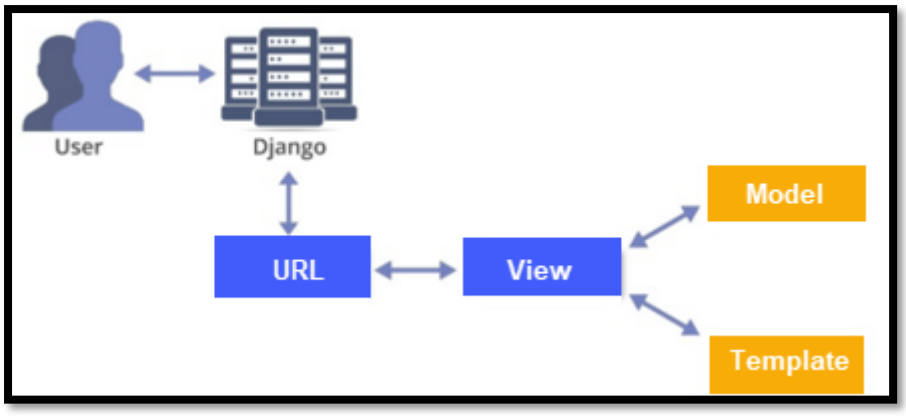
The **View** defines what data should be presented and returns an HTTP response.

The **Controller** is that part of the application that handles the user interaction.

Django MVC-MVT pattern

The Model-View-Template (MVT) is a different concept compared to MVC. The main difference between these two architectural patterns is that Django itself manages the Controller part (software code that controls the interactions between the Model and View). The template is an HTML file which mixed with Django Template Language file which also called DTL.

The below-given diagram shows how all the components of the MVT pattern interact with each other to serve specific to a user request.



MVT Architecture

As seen in above diagram, a user requests for a resource to Django. Django acts as a controller and checks to the available resource in URL.

If URL maps, a view is called which interacts with model and template. Django then responds to the user and sends a template as a response.

Components of Django

Form:

Django has a powerful form library which handles rendering forms as HTML. The library helps in validating submitted data and converting it to Python types.

Authentication:

It handles user accounts, groups, cookie-based user sessions, etc.

Admin:

It reads metadata in your models to provide a robust interface which can be used to manage content on your site.

Internationalization:

Django provides support for translating text into various languages, locale-specific formatting of dates, times, numbers, and timezones.

Security:

Django provides safeguard against the following attacks:

- Cross-Site Request Forgery (CSRF)
- Cross-site scripting
- SQL injection

- Clickjacking
- Remote code execution

Disadvantages of using Django

- A process cannot handle multiple requests simultaneously.
- Django is a very much monolithic tool.
- Components get deployed together which can create confusion.
- Knowledge of full system needed to work on Django
- Makes web application components tightly-coupled
- You need to do lengthy programming for minor tasks.

What is Flask?

Flask is an micro framework offering basic features of web app. This framework has no dependencies on external libraries. The framework offers extensions for form validation, object-relational mappers, open authentication systems, uploading mechanism, and several other tools.

Features of Flask

Here, are important features of Flask

- Integrated support for unit testing.
- RESTful request dispatching.
- Uses a Ninja2 template engine.
- It is based on Werkzeug toolkit.
- Support for secure cookies (client-side sessions).
- Extensive documentation.
- Google app engine compatibility.
- APIs are nicely shaped and coherent
- Easily deployable in production

Advantages of Flask

Here, are pros/benefits of using Flask

- Higher compatibility with latest technologies
- Technical experimentation
- Easier to use for simple cases
- Codebase size is relatively smaller
- High scalability for simple applications,
- Easy to build a quick prototype
- Routing URL is easy
- Easy to develop and maintain applications
- Database integration is easy
- Small core and easily extensible
- Minimal yet powerful platform
- Lots of resources available online especially on GitHub

Disadvantage of Flask

Here, are cons/drawback of Flask

- Slower MVP development in most cases,
- Higher maintenance costs for more complex systems
- Complicated maintenance for larger implementations.
- Async may be a little problem
- Lack of database and ORM
- Setting up a large project requires some previous knowledge of the framework
- Offers limited support and smaller community compared to Django

Which is Better?

- You should prefers flask if you want the granular level of control while a Django developer relies on an extensive community to create unique website.
- Django combined with the REST Framework helps you to build powerful APIs, whereas Flask requires more work, so there are high chances to make a mistake.
- The best method is to build a few basic CRUD apps with both frameworks and decide which framework fits your project style better.

What is the Spring Framework?

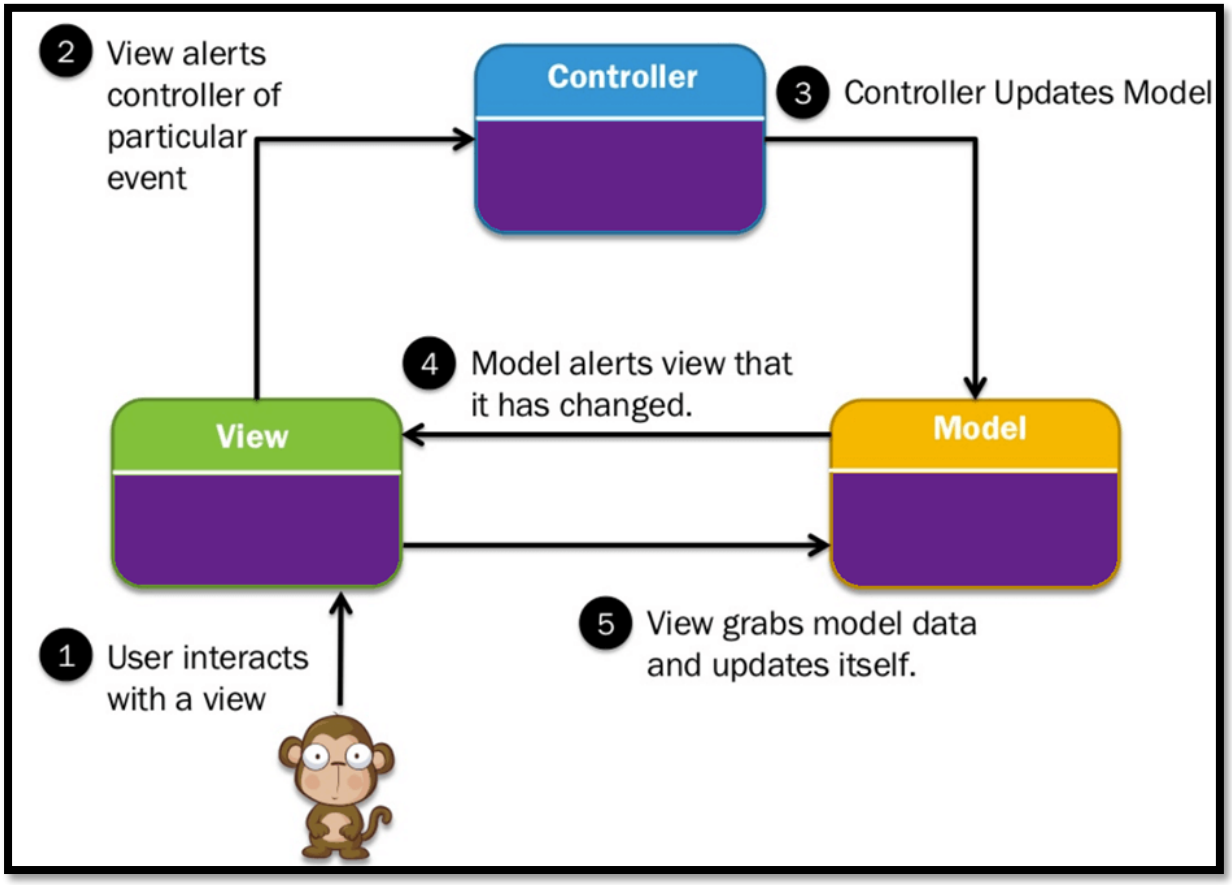
Spring Framework is an open-source framework for building web applications with Java as a programming language. It is powerful and lightweight yet easy to use, and it provides support for developing Java applications easily. Spring is a lightweight framework which can be thought of as a framework of frameworks because it also offers support for various frameworks such as hibernate, struts, tapestry, and JSF.

Features of Spring Framework

Here are some most prominent features of Spring Framework:

- Predefined templates
- Easy to test
- Loose coupling
- Lightweight
- Fast development
- Powerful abstraction
- Offers an array of resources
- Declarative support
- Offers comprehensive tools

Spring - MVC Framework

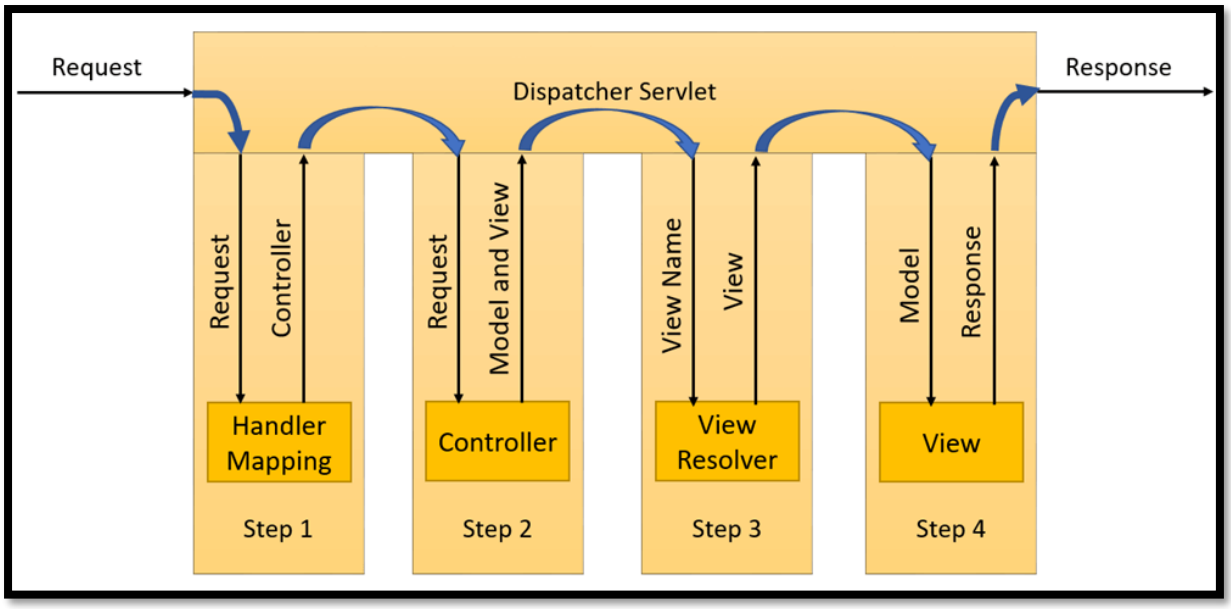


The Spring web MVC framework offers model-view-controller architecture offers components which helps you to be flexible and loosely coupled web applications.

The MVC pattern allows you to separate different aspects of the application while offering loose coupling between these elements. Spring MVC also helps you to build flexible and loosely coupled web applications.

The MVC design also allows you to separate business logic, presentation logic, and navigation logic. It also offers an elegant solution to use MVC in Spring Framework with the help of DispatcherServlet.

How MVC works in Spring?



- DispatcherServlet receives a request.
- After that, the DispatcherServlet communicates with HandlerMapping. It also revokes controller associated with that specific request.
- The Controller processes this request by calling the service methods, and a ModelAndView object returned by the DispatcherServlet.
- The view name is sent to a ViewResolver to search the actual View to invoke.
- After that, DispatcherServlet is passed to View to render the result.
- By using the model data, the View renders and send back result back to the user.

Advantages of Spring Framework

Here, are important pros/ benefits of using spring.

- Spring allows developers to develop enterprise-class applications with the help of POJOs.
- Offers templates for Hibernate, JDBC, Hibernate, JPA, etc. to reduce avoid writing lengthy code.
- Provides abstraction to Java Enterprise Edition (JEE).
- You can organize a spring in a modular fashion. So that if a number of packages and classes are substantial, you only need to about you need and ignore the rest.
- It offers declarative support for transactions, formatting, validation, caching, etc.
- The application developed using spring is simple as the environment-dependent code is moved into this framework.

Disadvantages of Spring Framework

Here, are drawbacks/cons of Spring Framework.

- Java spring offers too many restrictions.
- Code is public is it is visible to all.
- It also not offers custom-built features.

Struts Architecture and Life Cycle:

Architecture:

- Struts follows MVC (Model – View – Controller) architecture wherein (as discussed in previous chapters),
  - Model: JavaBeans (Business Logic)
  - View: JSP- Java Server Pages (Presentation Layer)
  - Controller: Servlet (Action classes – mainly used for session handling, security, primary logic of action etc.)

Flow of Servlet Execution:

- The basic flow of request to response is:
  - The client sends a request from the browser
  - The controller (ActionServlet) invokes the execute() method and function responsible for intended action is called. (in turn calls interceptors)
  - If required, the data is retrieved from the database by using getter and setter methods.
  - A response is created by the service () method which then forwards it to intended JSP.
  - The client receives the response via JSP in the browser.
- The complete scenario can be represented in sequence diagram as follows:

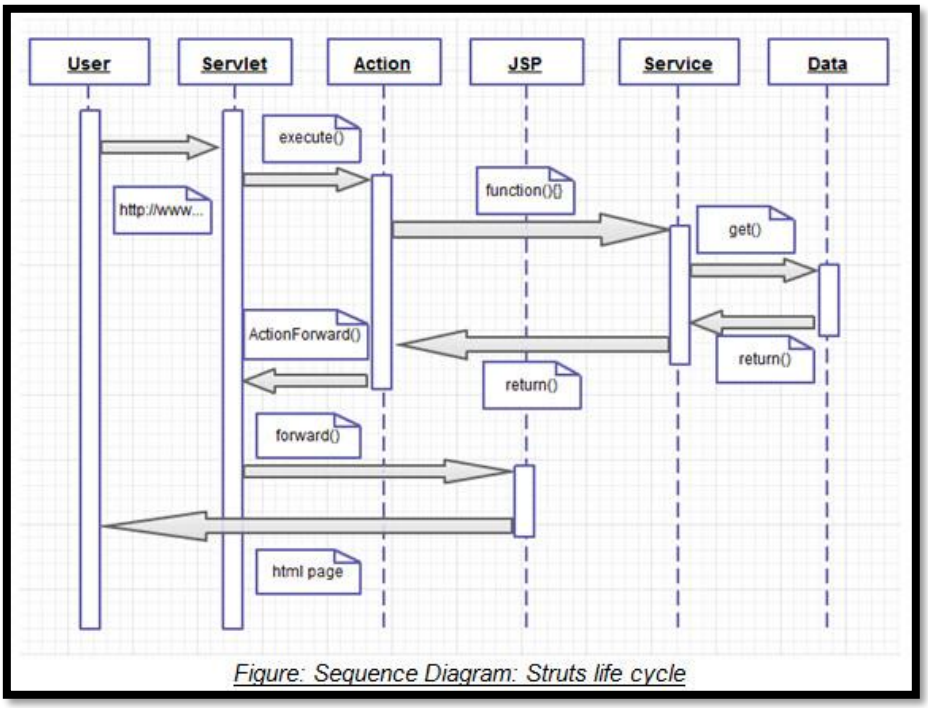
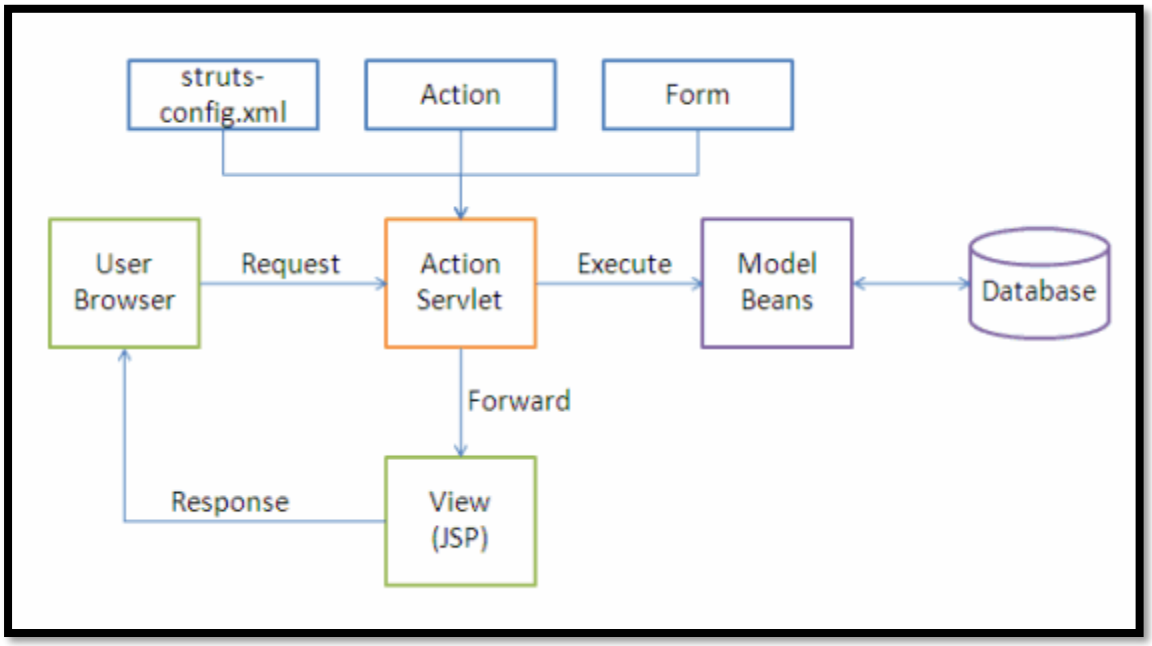


Figure: Sequence Diagram: Struts life cycle

The **model** contains the business logic and interact with the persistance storage to store, retrive and manipulate data.

The **view** is responsible for dispalying the results back to the user. In Struts the view layer is implemented using JSP.

The **controller** handles all the request from the user and selects the appropriate view to return. In Sruts the controller's job is done by the ActionServlet.



The following events happen when the Client browser issues an HTTP request.

- The *ActionServlet* receives the request.
- The *struts-config.xml* file contains the details regarding the *Actions*, *ActionForms*, *ActionMappings* and *ActionForwards*.
- During the startup the *ActionServelet* reads the *struts-config.xml* file and creates a database of configuration objects. Later while processing the request the *ActionServlet* makes decision by refering to this object.

When the *ActionServlet* receives the request it does the following tasks.

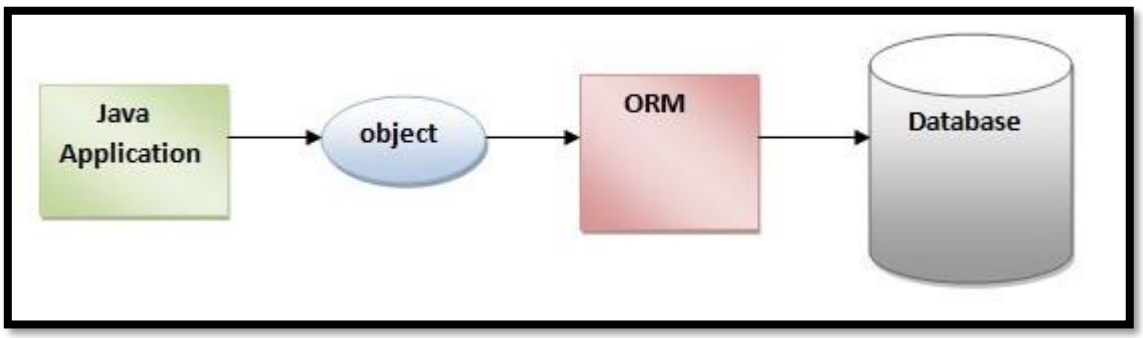
- Bundles all the request values into a JavaBean class which extends Struts *ActionForm* class.
- Decides which action class to invoke to process the request.
- Validate the data entered by the user.
- The action class process the request with the help of the model component. The model interacts with the database and process the request.
- After completing the request processing the *Action* class returns an *ActionForward* to the controller.
- Based on the *ActionForward* the controller will invoke the appropriate view.
- The HTTP response is rendered back to the user by the view component.

Hibernate Framework

Hibernate is a Java framework that simplifies the development of Java application to interact with the database. It is an open source, lightweight, ORM (Object Relational Mapping) tool. Hibernate implements the specifications of JPA (Java Persistence API) for data persistence.

ORM Tool

An ORM tool simplifies the data creation, data manipulation and data access. It is a programming technique that maps the object to the data stored in the database.



The ORM tool internally uses the JDBC API to interact with the database.

Advantages of Hibernate Framework

Following are the advantages of hibernate framework:

1) Open Source and Lightweight

Hibernate framework is open source under the LGPL license and lightweight.

2) Fast Performance

The performance of hibernate framework is fast because cache is internally used in hibernate framework. There are two types of cache in hibernate framework first level cache and second level cache. First level cache is enabled by default.

3) Database Independent Query

HQL (Hibernate Query Language) is the object-oriented version of SQL. It generates the database independent queries. So you don't need to write database specific queries. Before Hibernate, if database is changed for the project, we need to change the SQL query as well that leads to the maintenance problem.

4) Automatic Table Creation

Hibernate framework provides the facility to create the tables of the database automatically. So there is no need to create tables in the database manually.

5) Simplifies Complex Join

Fetching data from multiple tables is easy in hibernate framework.

6) Provides Query Statistics and Database Status

Hibernate supports Query cache and provide statistics about query and database status.

Django is one of the most popular web frameworks in Python. It is secure, robust and allows developers to rapidly develop projects and meet deadlines. It is free and open-source, it works both on Windows and *nix systems.

In this tutorial, we are going to learn how to install Django on Windows.

Install Python

Before you install Django, you must make sure that you have Python installed. You can check out this guide I wrote: how to install python on windows to learn how to install Python on Windows.

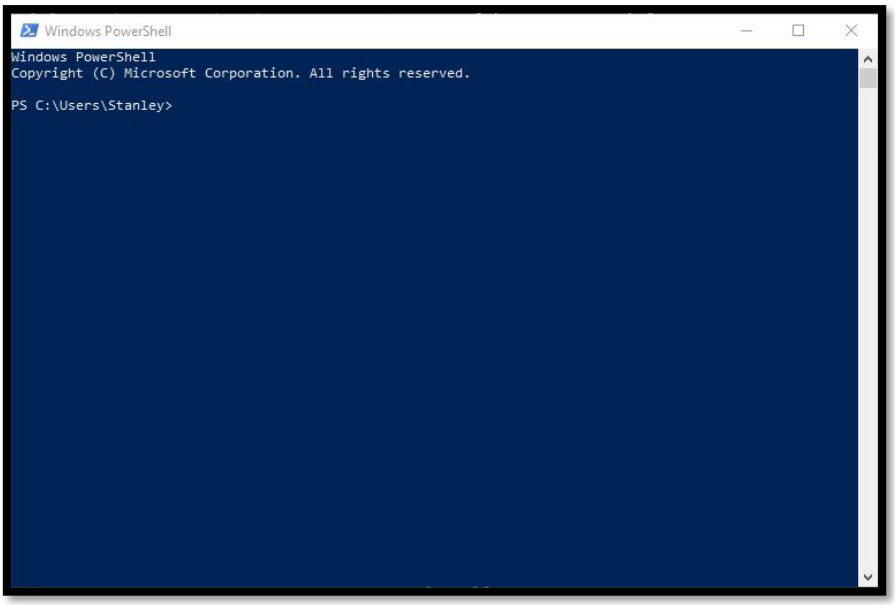
Install Django on Windows

The commands that we will run in this tutorial will work on both Windows Command Prompt(CMD) and Powershell. I will use Powershell but you can CMD if that's what you prefer.

Step 1 - Open Powershell

You can search for Powershell in the Windows search box or you can open the **Run** dialog box by holding **Windows logo key** and **R**(Win+R) and type powershell and then click **ok**.

You should now have a Powershell window opened.



Now that we have opened Powershell, Let's verify that Python has been installed.

Step 2 - Verify Python Installation
Type **python -V** on the prompt to verify that Python has been successfully installed

```
> python -V
```

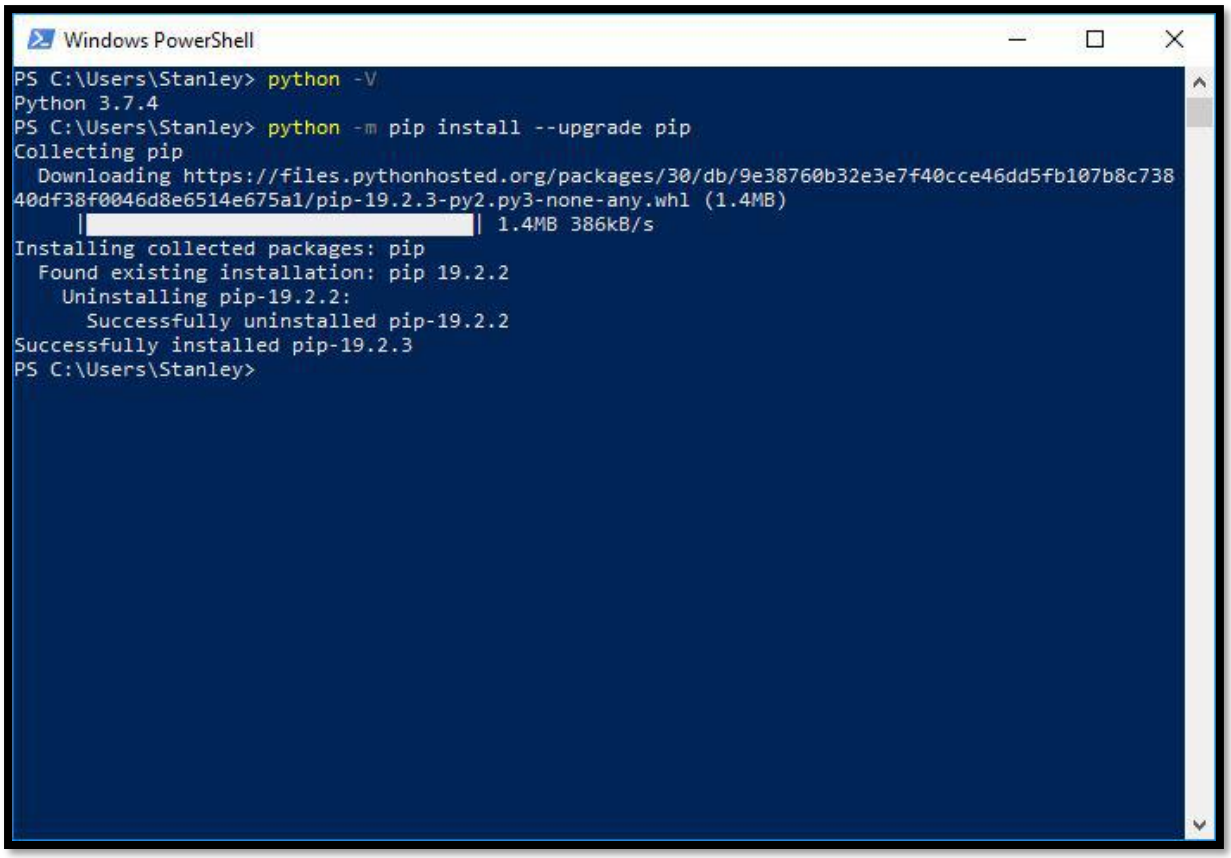
you should see the Python version installed, being printed like below:

```
PS C:\Users\Stanley> python -V
Python 3.7.4
```

Step 3 - Upgrade Pip
Python now comes with **pip** by default. But most of the time, it comes with an old version. it is always a good practice to upgrade pip to the latest version

```
> python -m pip install --upgrade pip
```

You should see something like the screenshot below showing that the upgrade was a success:



Step 4 - Create a Project Directory
let's create a project directory. We will name it **django_project** since this tutorial is just a demo but in the real world, the project directory's name would be forum, blog, etc.

To create the directory:

```
> mkdir django_project
```
Change into the django_project directory:

```
> cd django_project
```
Your prompt should now show you that you're in the **"django_project"** directory like below:

```
PS C:\Users\Username\django_project>
```

Step 5 - Create Virtual Environment

A virtual environment(virtualenv) is an isolated Python environment where you can install packages and dependencies without affecting other Python projects. You can learn more about virtualenv here.

To create a virtual environment, type `python -m venv venv` and wait for a few seconds:

```
> python -m venv venv
```
It will create a directory called `venv` in the project directory.

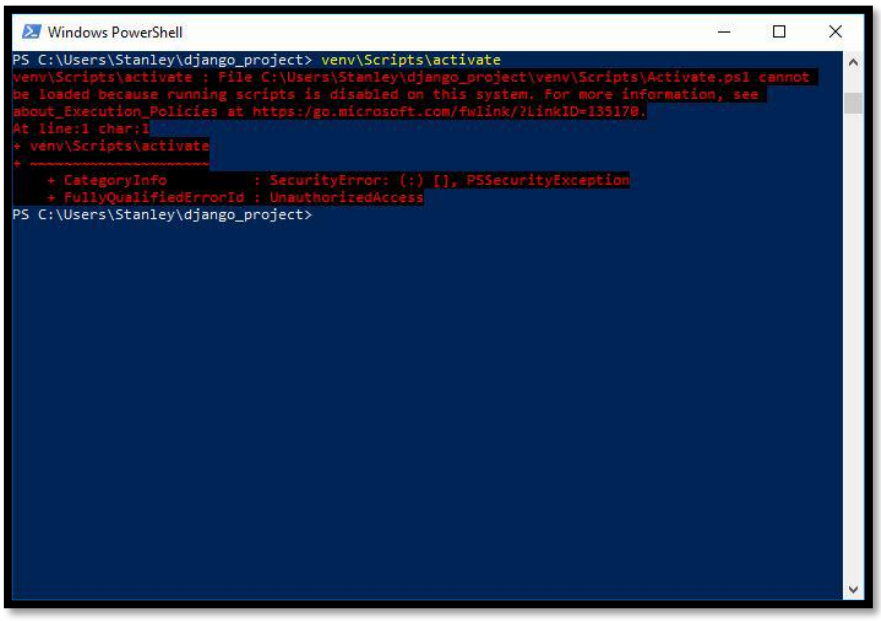
Next, we will activate the virtual environment.

Step 6 - Activate Virtual Environment

Run the following command to activate the virtualenv.

```
> venv\Scripts\activate
```
When virtualenv is activated, you will see a **(venv)** at the beginning of the prompt.

```
(venv) PS C:\Users\Stanley\django_project>
```
If you run into an error like the one below on Powershell when activating virtualenv, for the sake of brevity, I described the reason and the solution here: https://www.stanleyulili.com/powershell/solution-to-running-scripts-is-disabled-on-this-system-error-on-powershell/



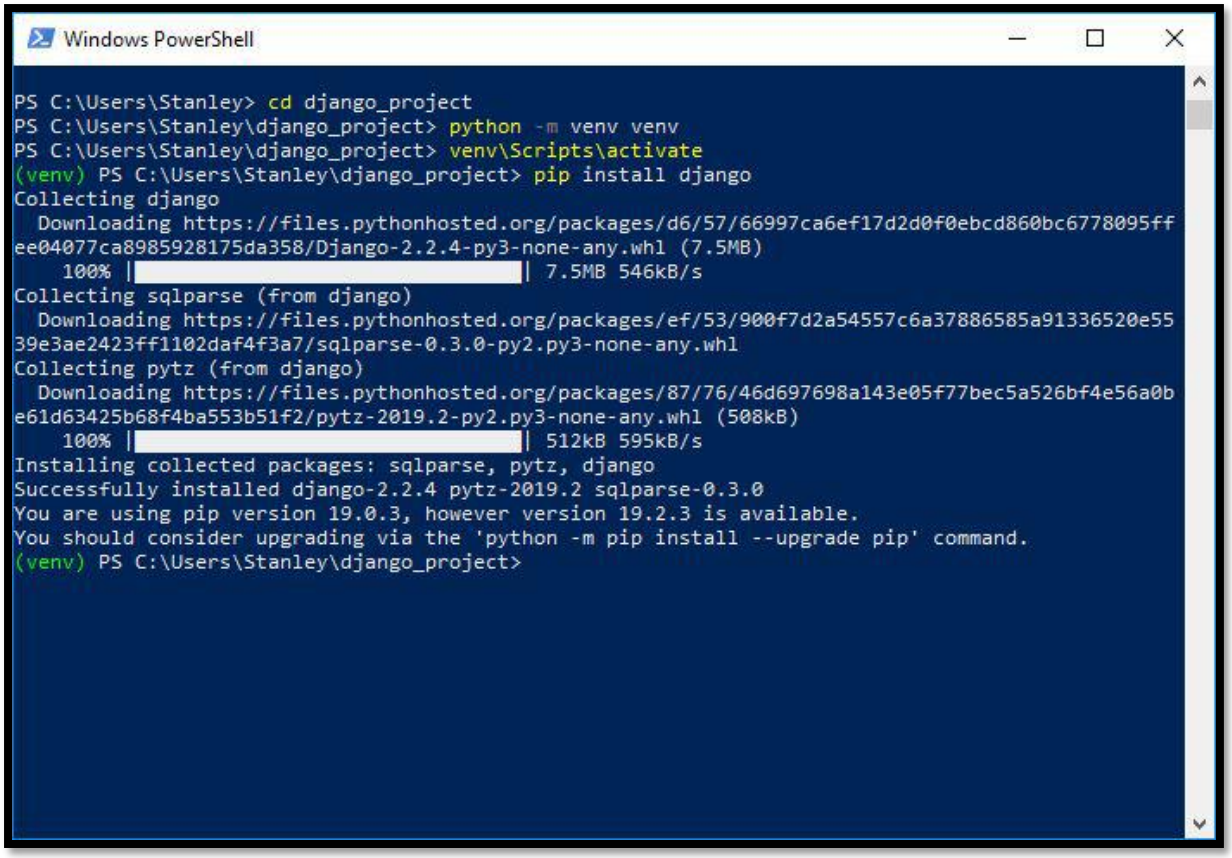
Now that virtual environment is activated, let's install Django.

Step 7 - Install Django

We are going to use pip to install Django. So run the following command to start the installation:

```
(venv)> pip install django
```

The command will install the latest version of Django. You should see Django being downloaded just like the screenshot below:



If you want to install a different Django version, you can specify the version:

```
(venv)> pip install django==2.1
```

Step 8 - Start a New Project

We will use django-admin startproject to create our Django project structure called "testsite".

Run the following command:

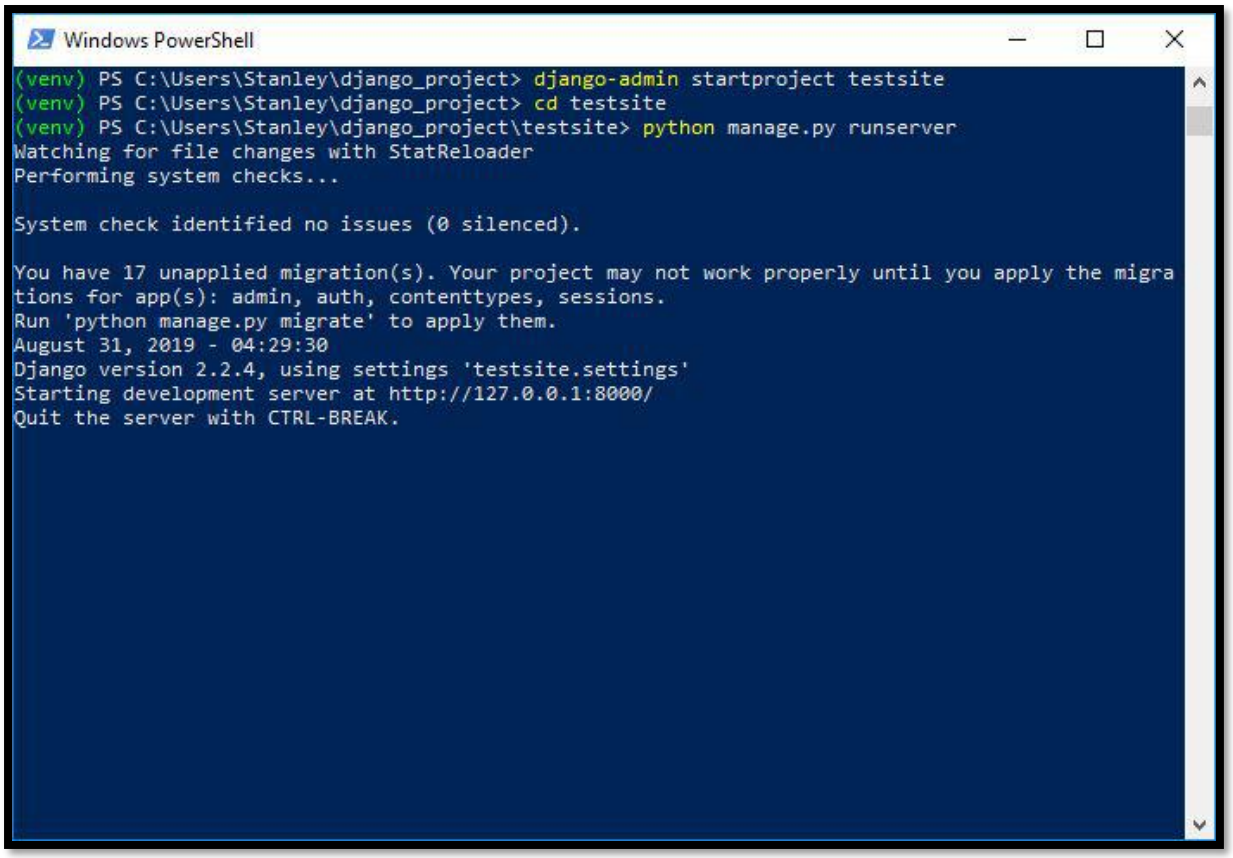
```
(venv)> django-admin startproject testsite
```

Navigate into the **testsite** directory by running the following command:

```
(venv)> cd testsite
```

Step 9 - Run the Server
Let's now run the development server using the **manage.py runserver**:

```
(venv)> python manage.py runserver
```



Visit http://127.0.0.1:8000/ in your web browser.

Step 1 - Open Command prompt

You can search for cmd in the Windows search box or you can open the Run dialog box by holding Windows logo key and R(Win+R) and type cmd and then click ok.

You should now have a cmd window opened.

Step 2 –

Verify Python Installation Type python -V on the prompt to verify that Python has been successfully installed

```
C:\Users\Vishal>python -V
Python 3.8.5
```

Step 3 - Upgrade Pip Python now comes with pip by default. But most of the time, it comes with an old version. it is always a good practice to upgrade pip to the latest version

```
C:\Users\Vishal>python -m pip install --upgrade pip
Collecting pip
  Downloading pip-20.2.3-py2.py3-none-any.whl (1.5 MB)
     |████████████████████████████████| 1.5 MB 1.1 MB/s
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 20.2.2
    Uninstalling pip-20.2.2:
      Successfully uninstalled pip-20.2.2
Successfully installed pip-20.2.3
```

Step 4 - Create a Project Directory let's create a project directory.

We will name it django_project since this tutorial is just a demo but in the real world, the project directory's name would be forum, blog, etc.

create a directory and change into that directory. Your prompt should now show you that you're in the "django_project" directory like below:

```
C:\Users\Vishal>cd Desktop

C:\Users\Vishal\Desktop>mkdir django_project

C:\Users\Vishal\Desktop>cd django_project

C:\Users\Vishal\Desktop\django_project>
```

Step 5 – Create Virtual Environment

A virtual environment(virtualenv) is an isolated Python environment where you can install packages and dependencies without affecting other Python projects.

To create a virtual environment, type python -m venv venv and wait for a few seconds: It will create a directory called venv in that project folder

```
C:\Users\Vishal\Desktop>cd django_project

C:\Users\Vishal\Desktop\django_project>python -m venv venv
```

Next, we will activate the virtual environment.

Step 6 - Activate Virtual Environment Run the following command to activate the virtualenv. When virtualenv is activated, you will see a (venv) at the beginning of the prompt.

```
C:\Users\Vishal\Desktop\django_project>venv\scripts\activate

(venv) C:\Users\Vishal\Desktop\django_project>
```

Now that virtual environment is activated, let's install Django.

Step 7 - Install Django We are going to use pip to install Django. So run the following command to start the installation:

```
(venv) C:\Users\Vishal\Desktop\django_project>pip install django
Collecting django
  Downloading Django-3.1.1-py3-none-any.whl (7.8 MB)
     |████████████████████████████████| 7.8 MB 365 kB/s
Collecting sqlparse>=0.2.2
  Using cached sqlparse-0.3.1-py2.py3-none-any.whl (40 kB)
Collecting pytz
  Using cached pytz-2020.1-py2.py3-none-any.whl (510 kB)
Collecting asgiref~=3.2.10
  Using cached asgiref-3.2.10-py3-none-any.whl (19 kB)
Installing collected packages: sqlparse, pytz, asgiref, django
Successfully installed asgiref-3.2.10 django-3.1.1 pytz-2020.1 sqlparse-0.3.1
```

If you want to install a different Django version, you can specify the version:

e.g. (venv)> pip install django==2.1

Step 8 - Start a New Project We will use django-admin startproject to create our Django project structure called "tour".

Run the following command: and Navigate into the tour directory by running the following command:

```
(venv) C:\Users\Vishal\Desktop\django_project>django-admin startproject tour

(venv) C:\Users\Vishal\Desktop\django_project>cd tour

(venv) C:\Users\Vishal\Desktop\django_project\tour>
```

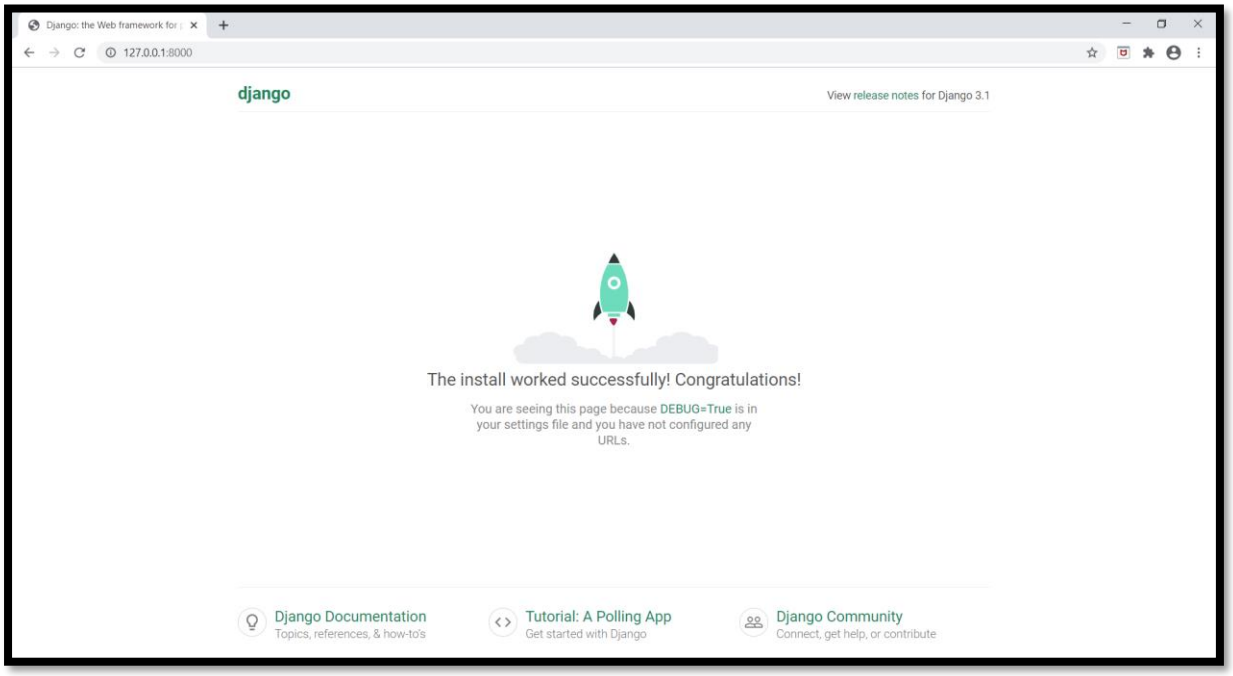
```
(venv) C:\Users\Vishal\Desktop\django_project\tour>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
September 16, 2020 - 20:39:12
Django version 3.1.1, using settings 'tour.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

```
(venv) C:\Users\Vishal\Desktop\django_project\tour>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin,
 auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
September 16, 2020 - 20:39:12
Django version 3.1.1, using settings 'tour.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Step 9 - Run the Server Let's now run the development server using the manage.py runserver:

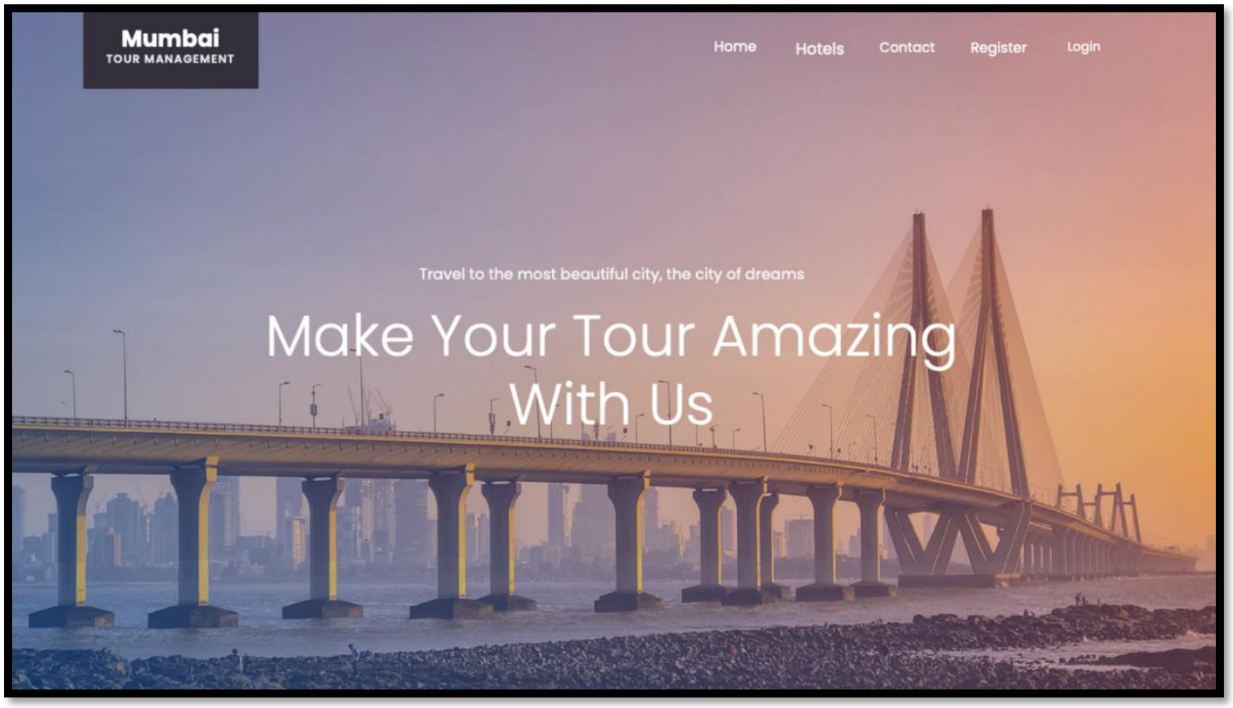
```
(venv) C:\Users\Vishal\Desktop\django_project\tour>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin,
 auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
September 16, 2020 - 20:39:12
Django version 3.1.1, using settings 'tour.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[16/Sep/2020 20:41:05] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:41:05] "GET /static/admin/css/fonts.css HTTP/1.1" 200 423
[16/Sep/2020 20:41:05] "GET /static/admin/fonts/Roboto-Bold-webfont.woff HTTP/1.1" 200 86184
[16/Sep/2020 20:41:05] "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 200 85876
[16/Sep/2020 20:41:05] "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 200 85692
Not Found: /favicon.ico
[16/Sep/2020 20:41:05] "GET /favicon.ico HTTP/1.1" 404 1970
[16/Sep/2020 20:43:28] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:43:28] "GET /static/admin/css/fonts.css HTTP/1.1" 200 423
[16/Sep/2020 20:43:29] "GET /static/admin/fonts/Roboto-Bold-webfont.woff HTTP/1.1" 200 86184
[16/Sep/2020 20:43:29] "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 200 85876
[16/Sep/2020 20:43:29] "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 200 85692
Not Found: /favicon.ico
[16/Sep/2020 20:43:29] "GET /favicon.ico HTTP/1.1" 404 1970
[16/Sep/2020 20:44:09] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:09] "GET /static/admin/css/fonts.css HTTP/1.1" 304 0
[16/Sep/2020 20:44:09] "GET /static/admin/fonts/Roboto-Bold-webfont.woff HTTP/1.1" 304 0
[16/Sep/2020 20:44:09] "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 304 0
[16/Sep/2020 20:44:09] "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 304 0
[16/Sep/2020 20:44:10] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:11] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:11] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:11] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:12] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:12] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:12] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:13] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:13] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:13] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:13] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:27] "GET / HTTP/1.1" 200 16351
[16/Sep/2020 20:44:27] "GET /static/admin/css/fonts.css HTTP/1.1" 304 0
[16/Sep/2020 20:44:29] "GET / HTTP/1.1" 200 16351
```

Visit http://127.0.0.1:8000/ in your web browser.

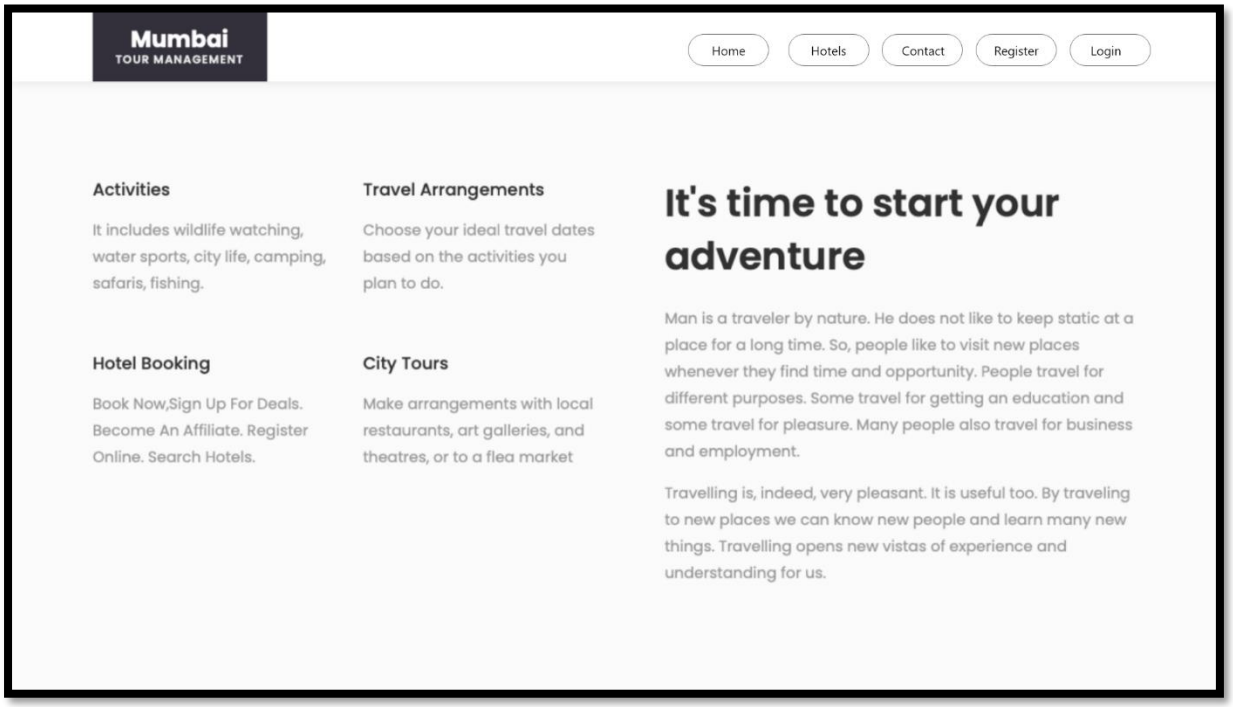


Prototype:

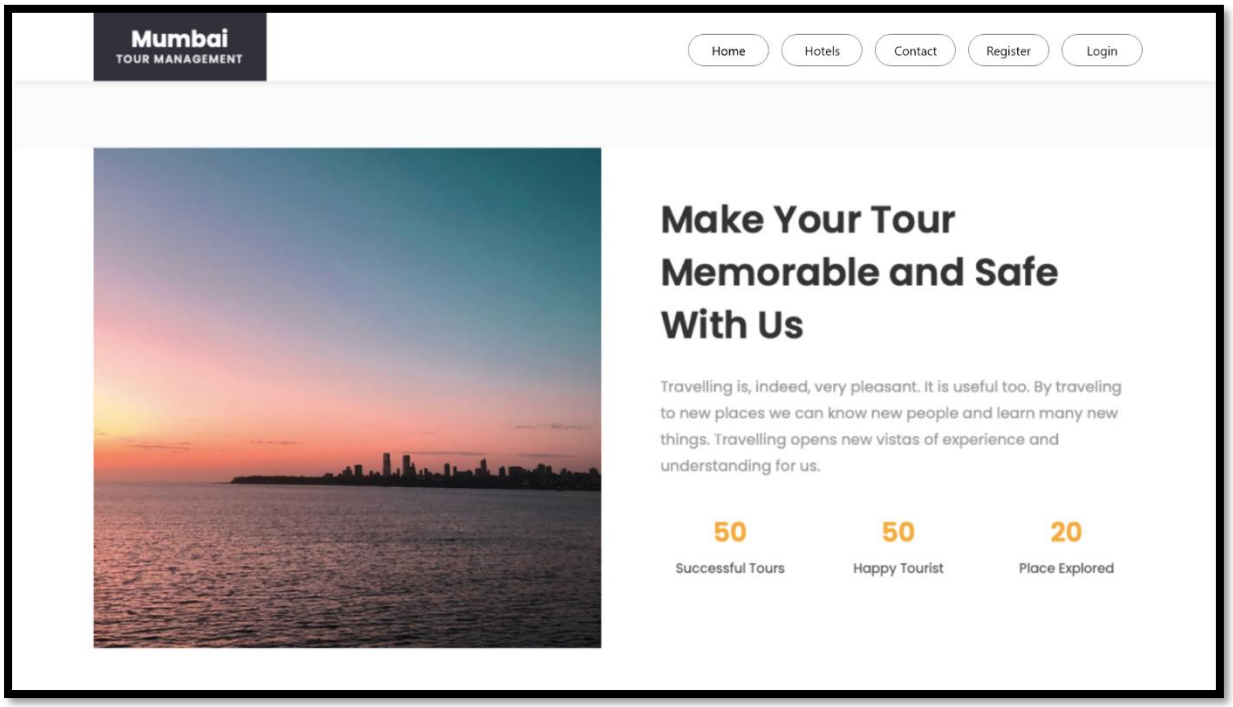
This is the home page of Mumbai Tour and guide management.



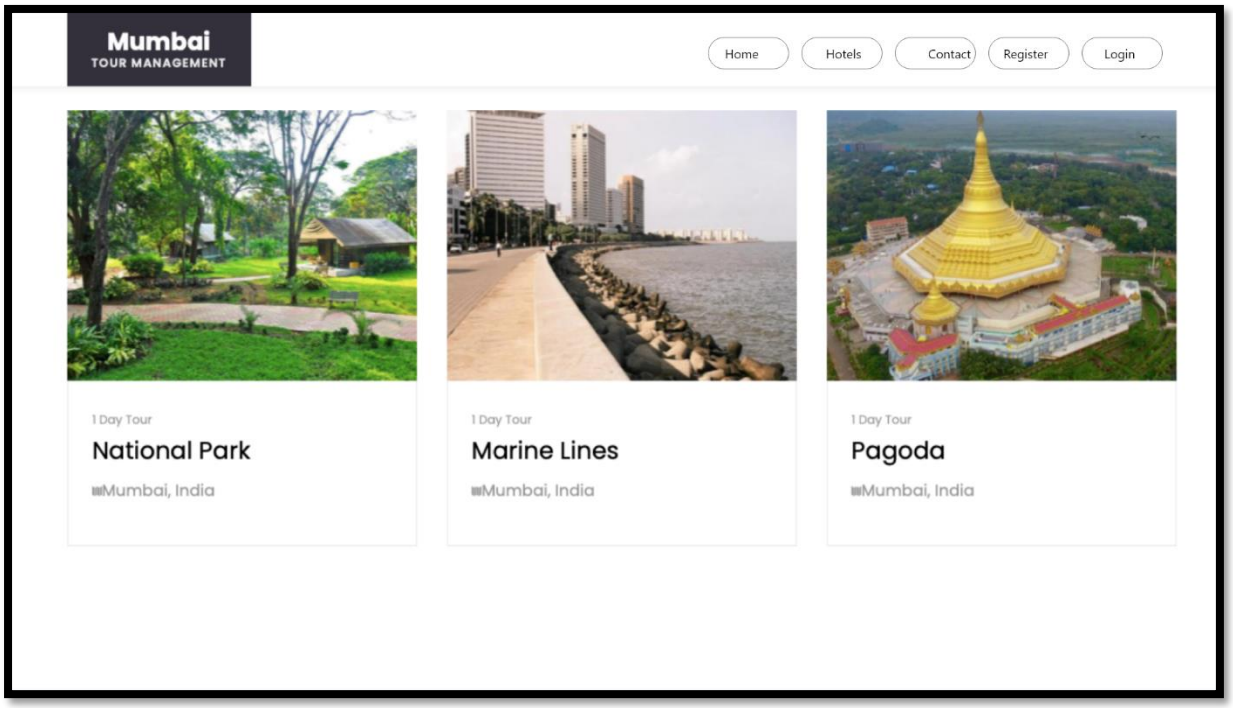
Here user can see whatever activities, tours as well as hotel booking features in system

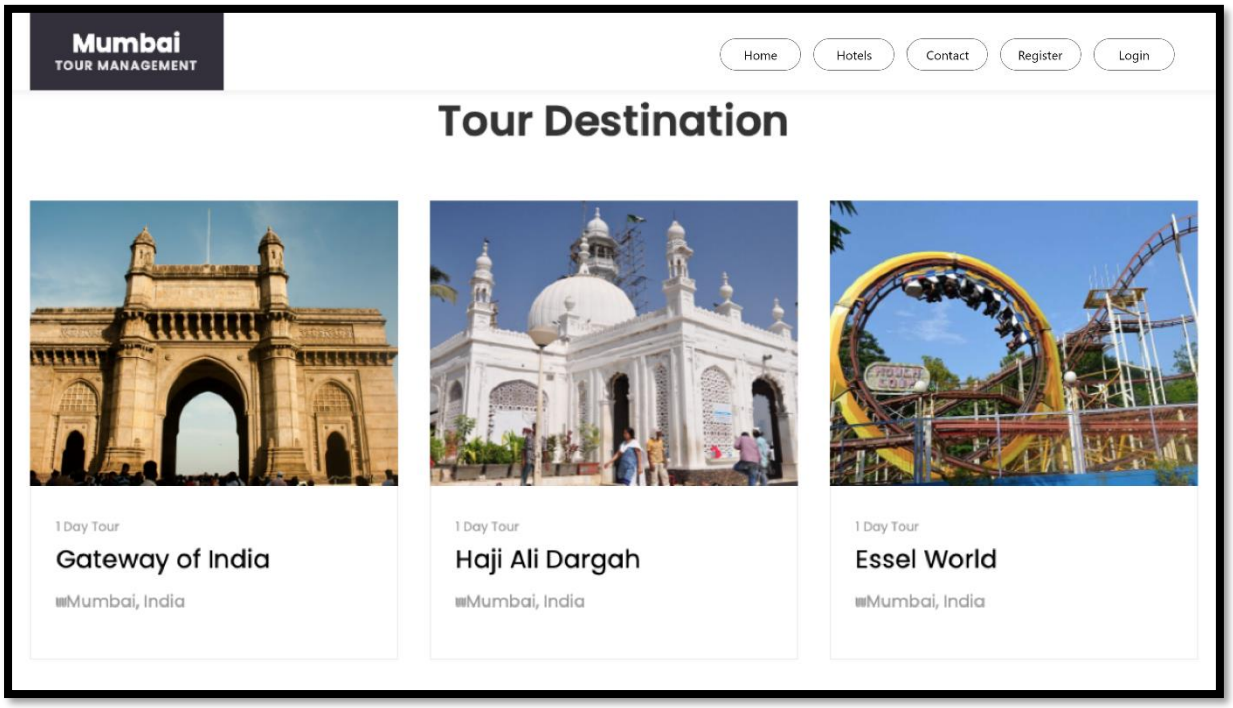


User can see count of happy tourist and place explored by tourist

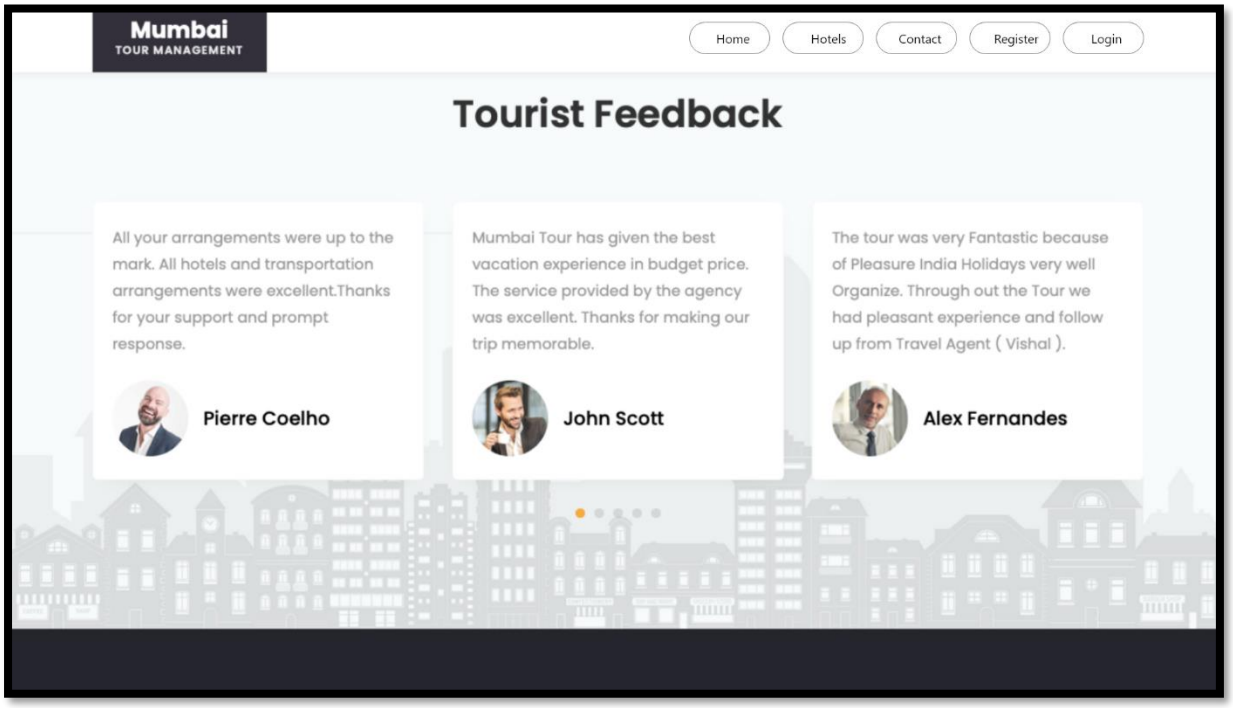


After that user can view tour destination

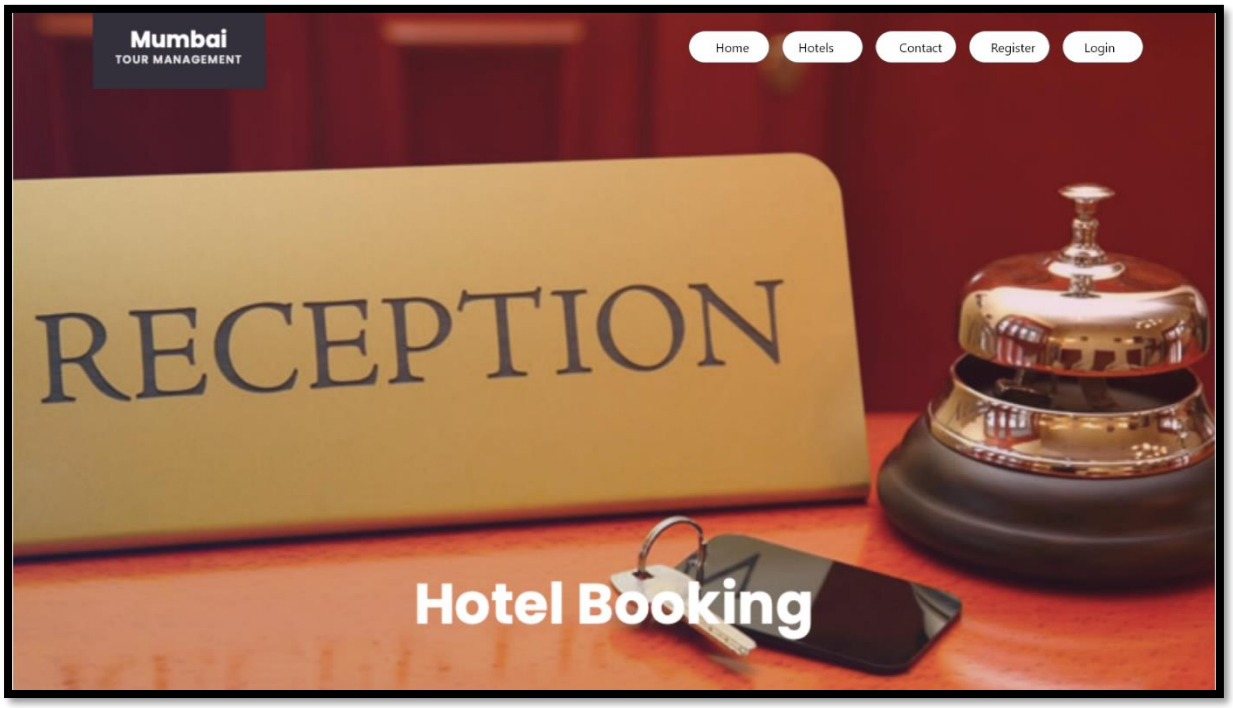




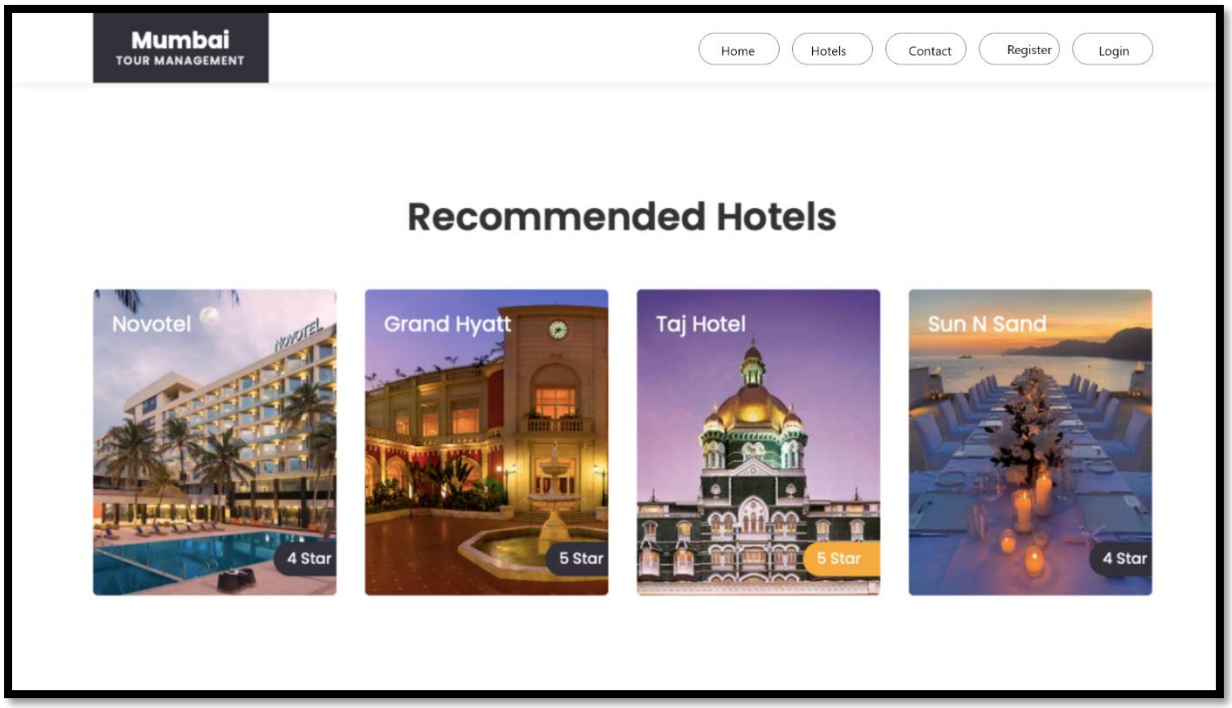
Then user get tourist feedback



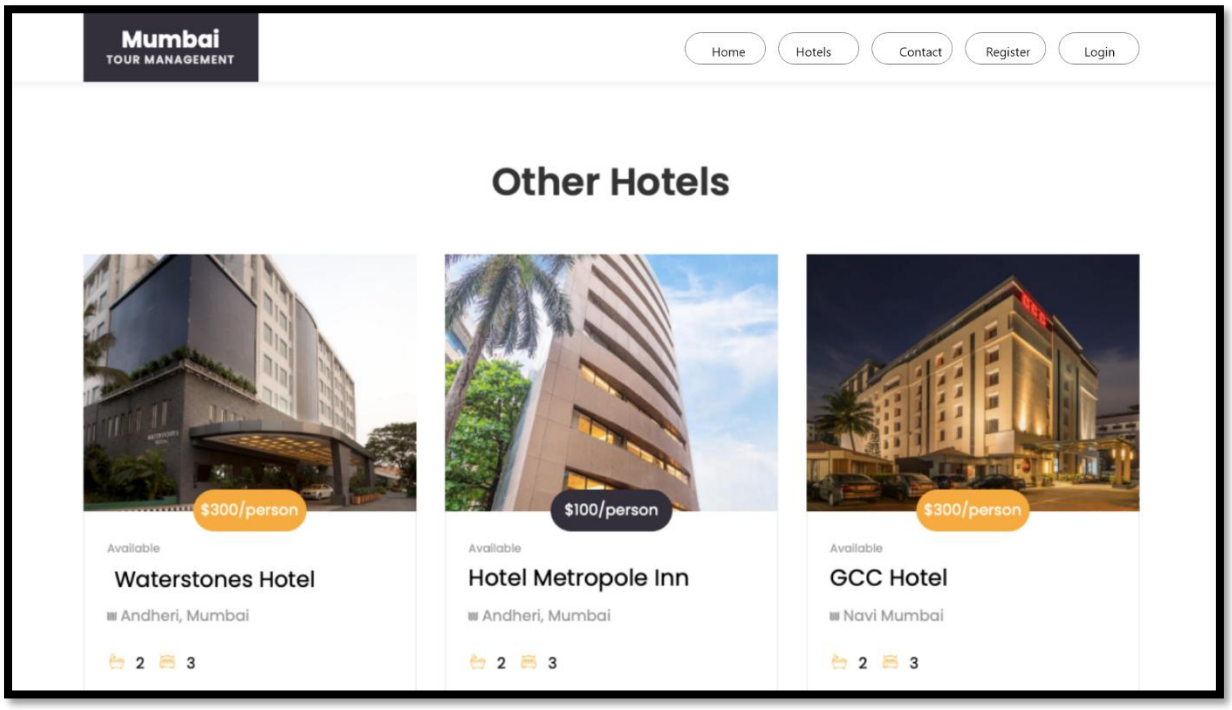
This is Hotel Reception view



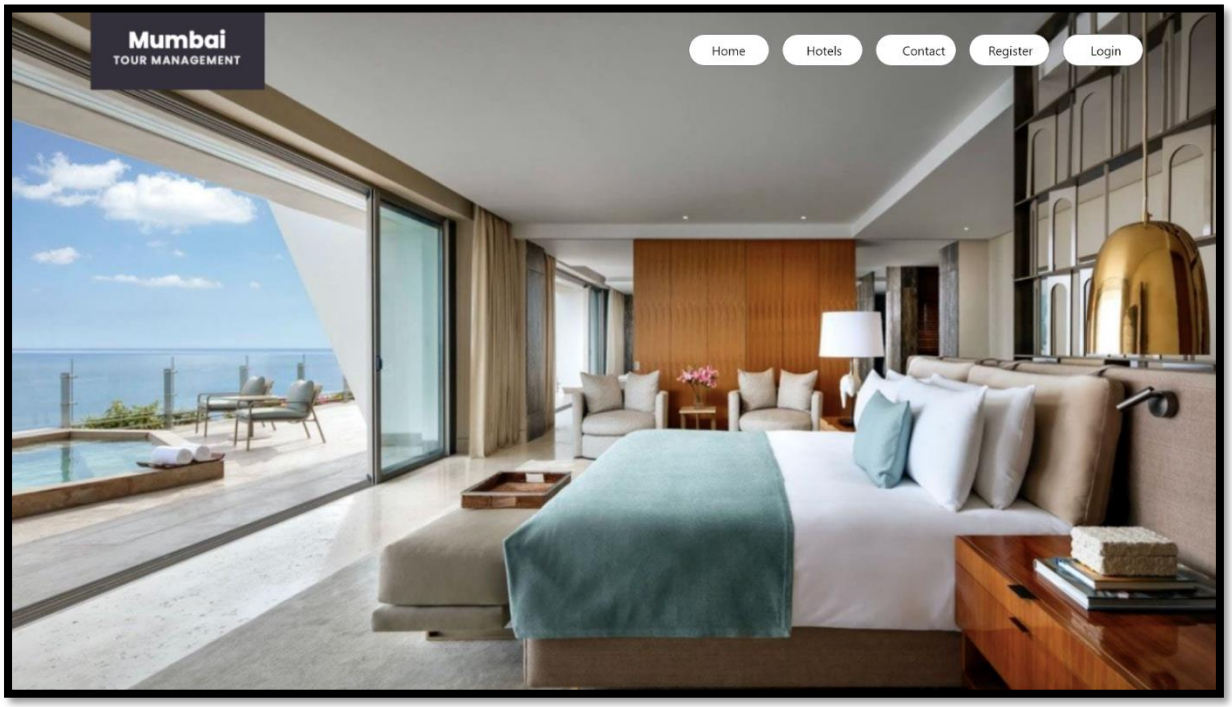
Here, User can see Recommended Hotels which are near by tour location



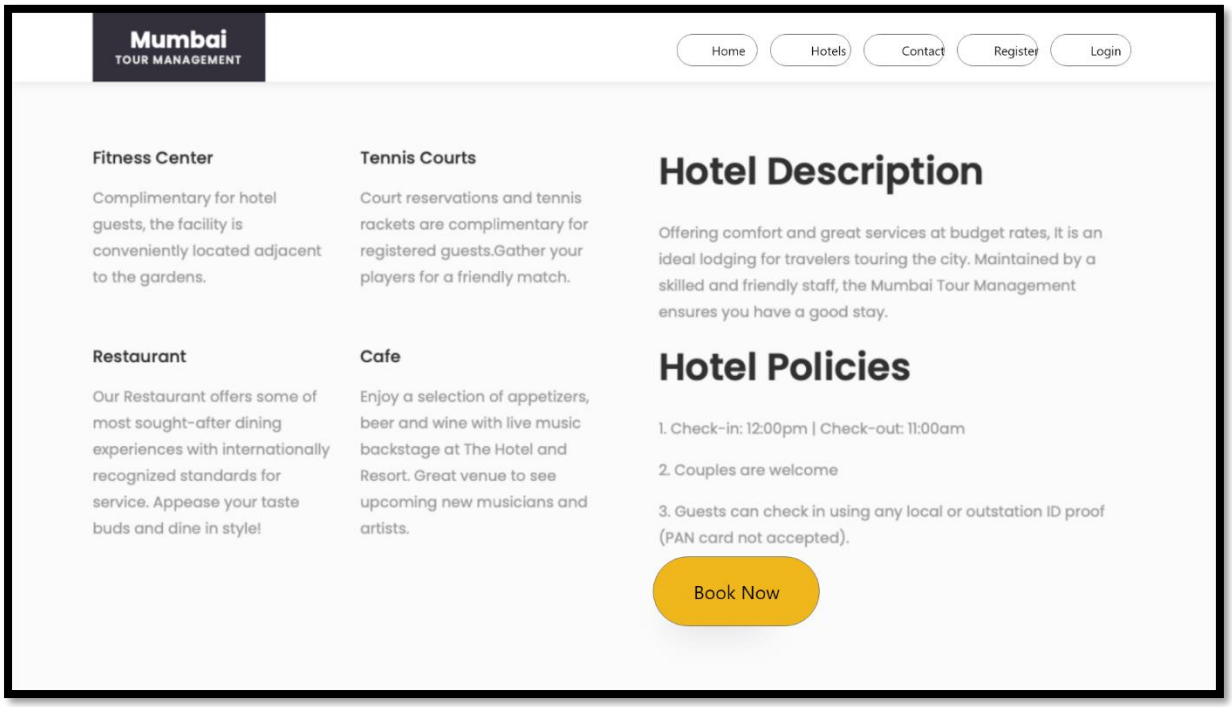
Other Hotels



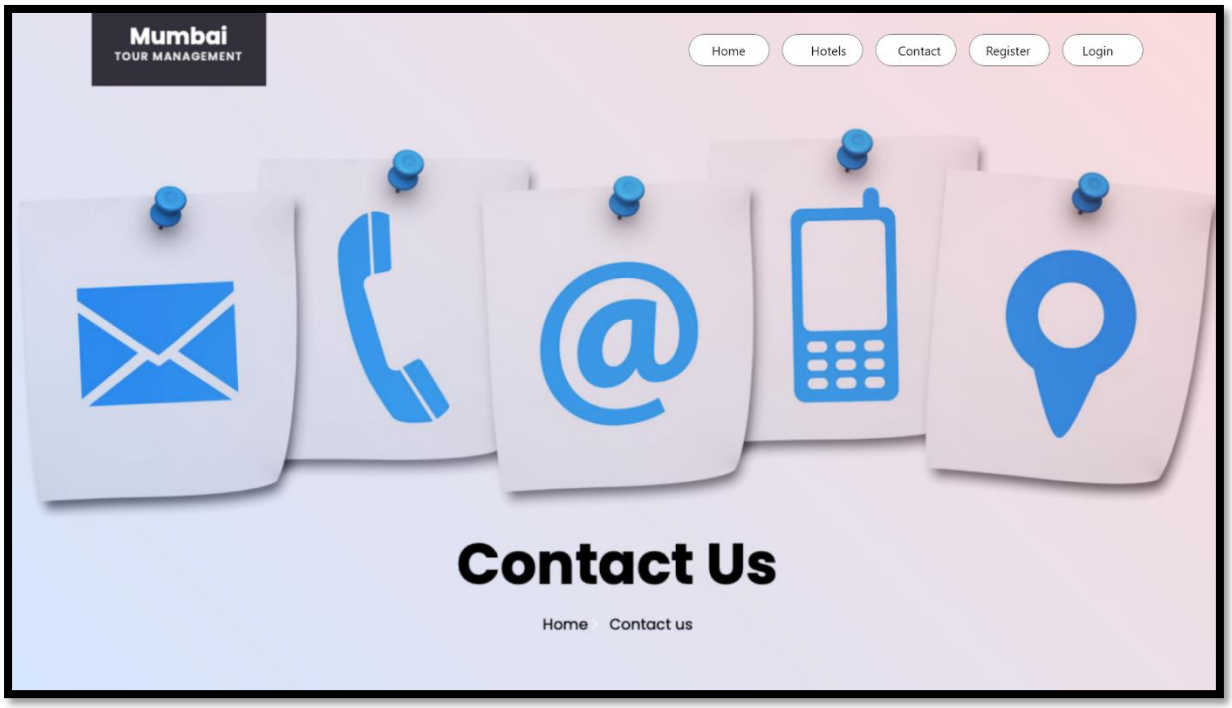
Then, User can view of room in hotel.



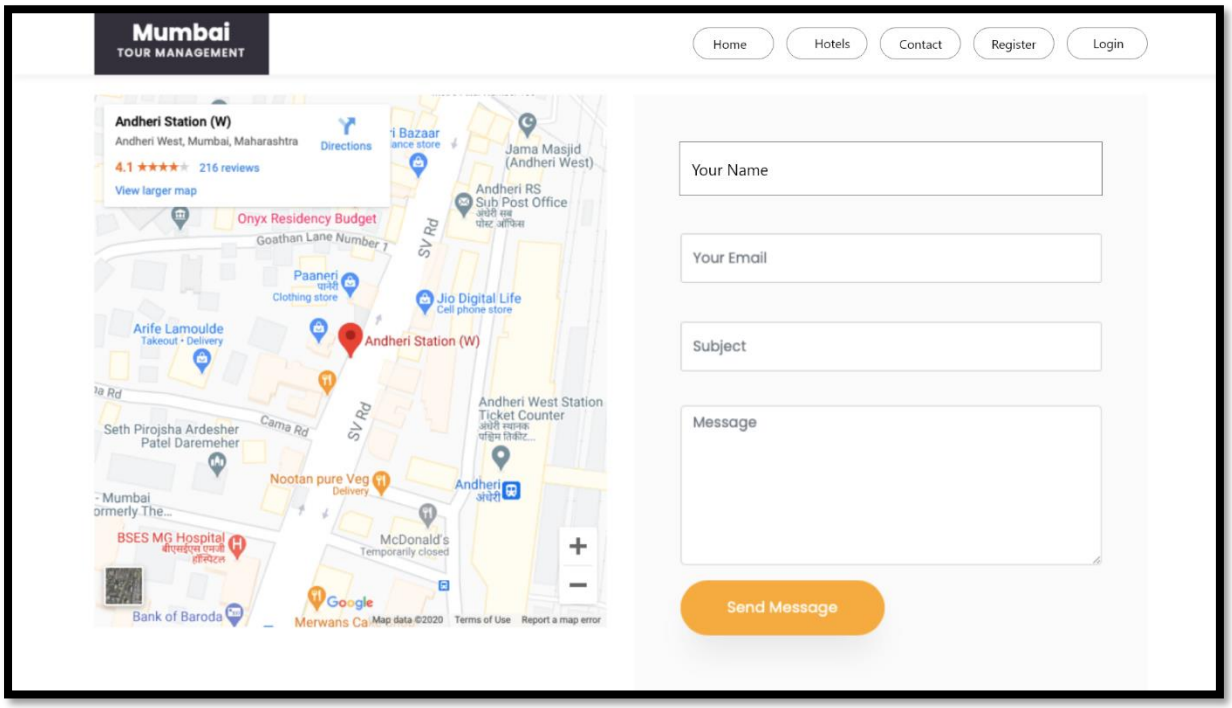
After that can get features of hotel like Fitness centre, Tennis courts, Cafe etc.

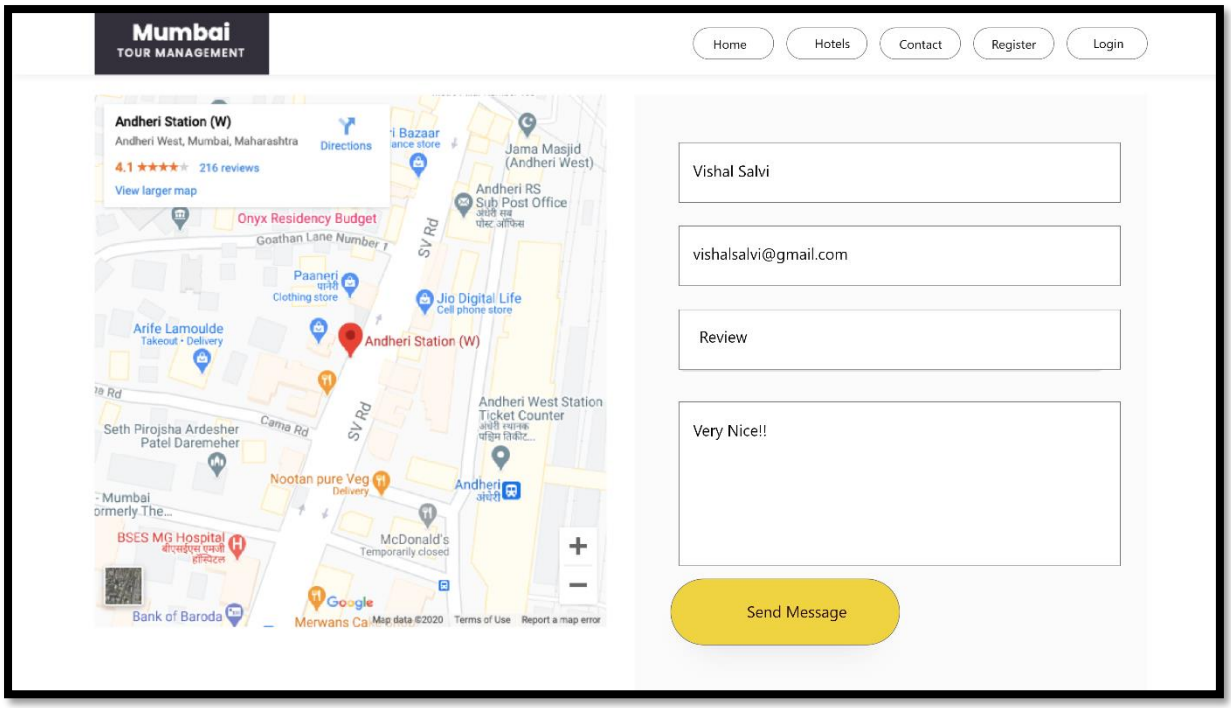


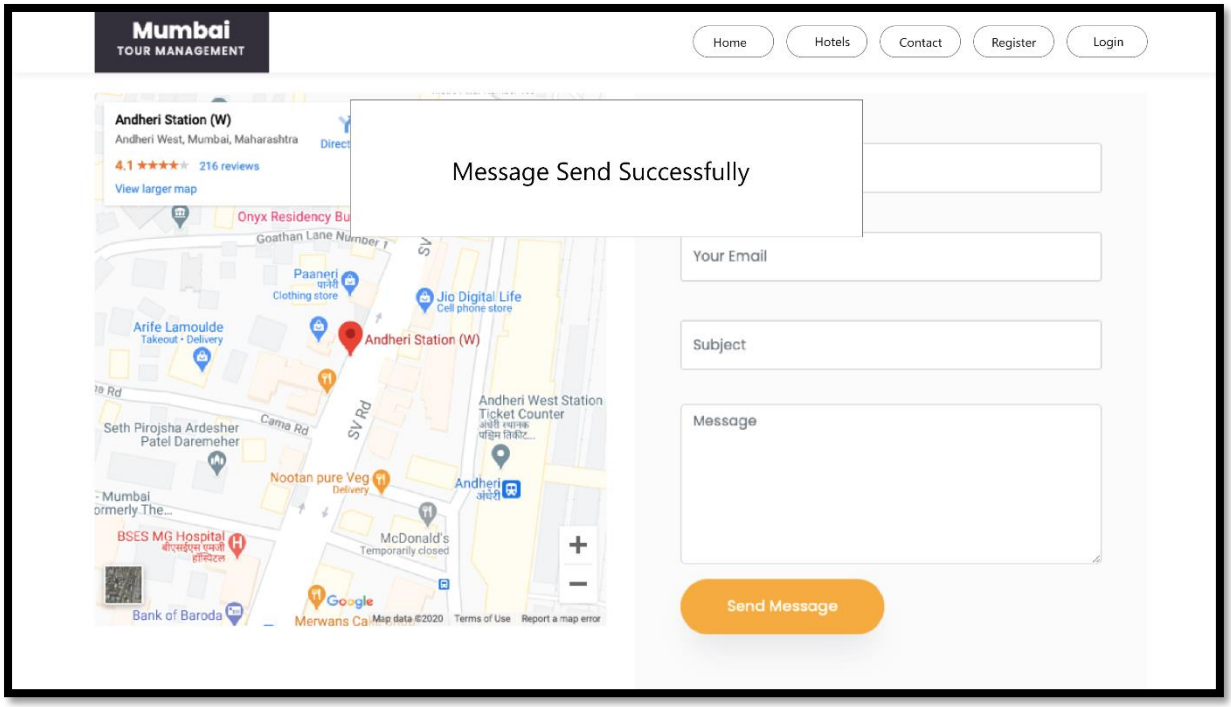
Next is contact page



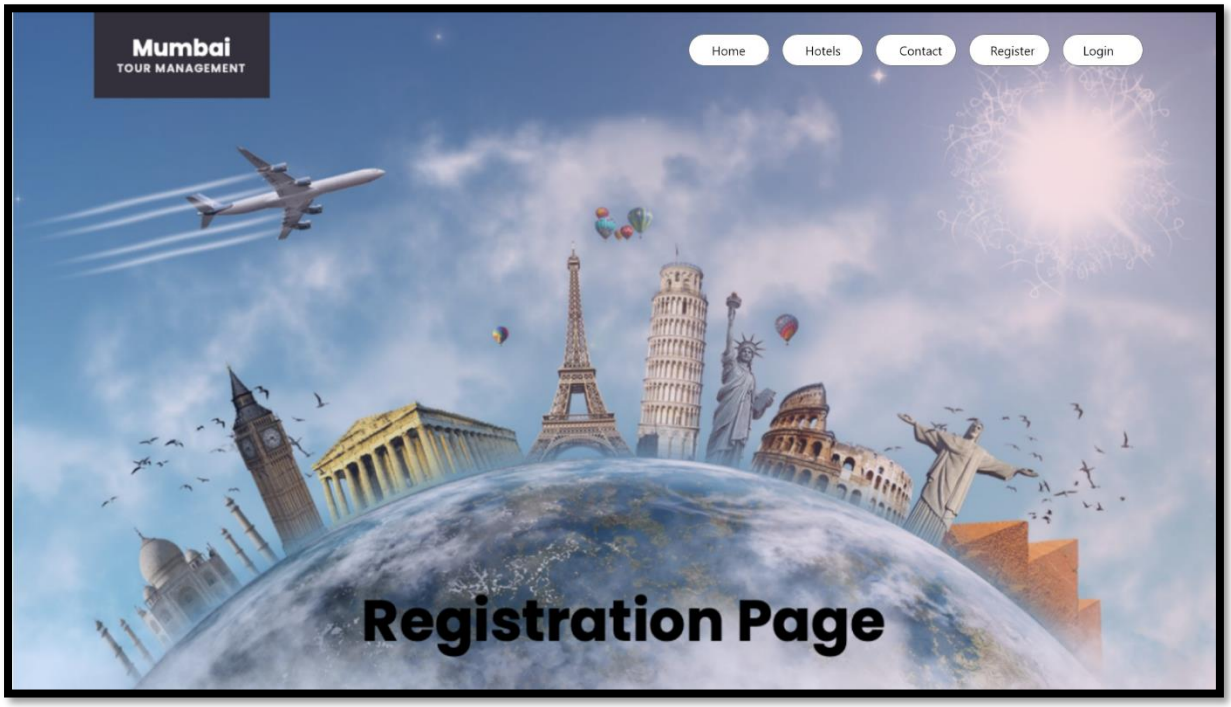
User can send message to hotel as well as can add review.



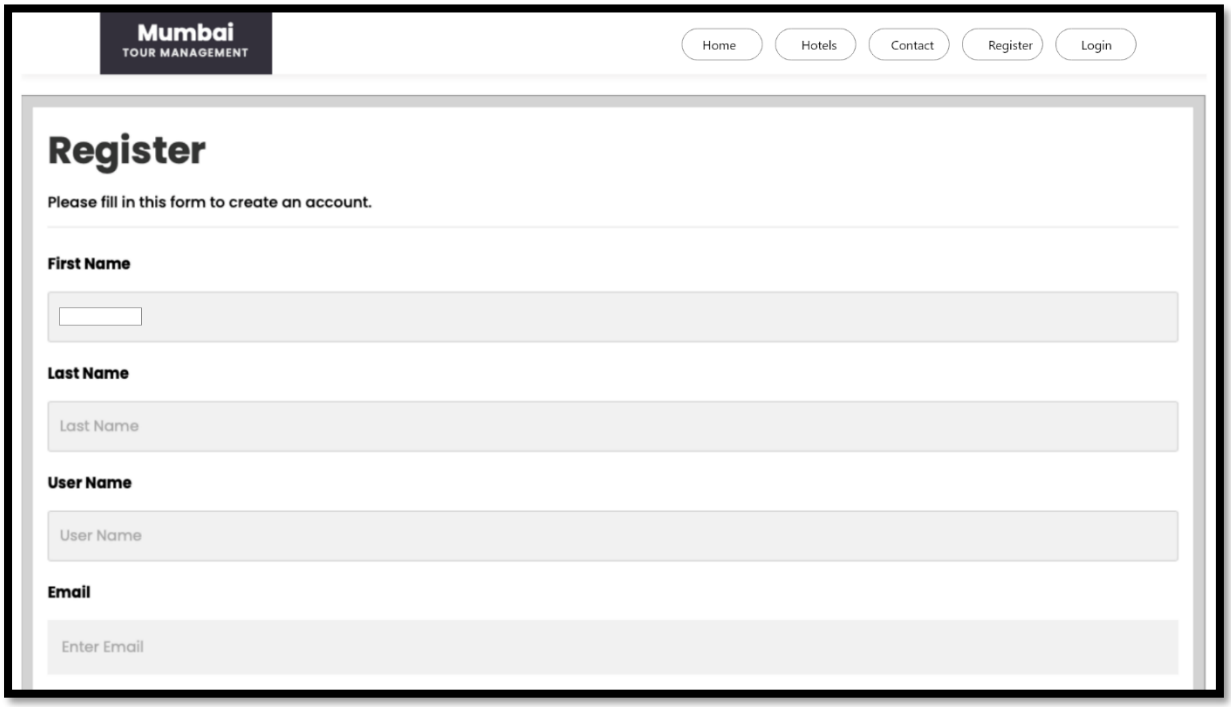

Mumbai
TOUR MANAGEMENT
Home
Hotels
Contact
Register
Login
Andheri Station (W)
Andheri West, Mumbai, Maharashtra
Directions
4.1 ★★★★☆ 216 reviews
View larger map
Onyx Residency Budget
Goathan Lane Number 1
SV Rd
Paaneri
Clothing store
Jio Digital Life
Cell phone store
Arife Lamoulde
Takeout • Delivery
Andheri Station (W)
Andheri West Station
Ticket Counter
Seth Pirojsha Ardesher
Patel Daremeher
Cama Rd
Nootan pure Veg
Delivery
Andheri
Mumbai
BSES MG Hospital
McDonald's
Temporarily closed
Bank of Baroda
Google
Map data ©2020
Terms of Use
Report a map error
Jama Masjid
(Andheri West)
Andheri RS
Sub Post Office
Vishal Salvi
vishalsalvi@gmail.com
Review
Very Nice!!
Send Message

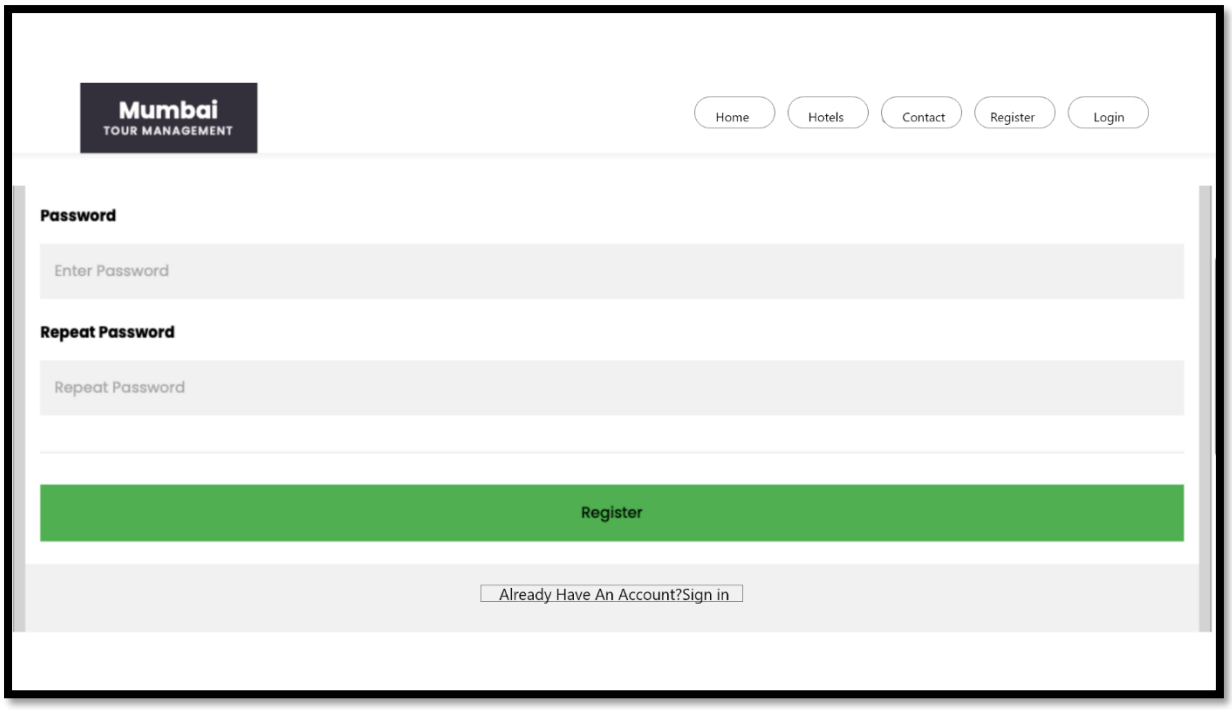

Mumbai
TOUR MANAGEMENT
Home
Hotels
Contact
Register
Login
Andheri Station (W)
Andheri West, Mumbai, Maharashtra
4.1 ★★★★☆ 216 reviews
View larger map
Message Send Successfully
Your Email
Subject
Message
Send Message
Onyx Residency Bu
Goathan Lane Number 1
Paaneri
Clothing store
Jio Digital Life
Cell phone store
Arife Lamoulde
Takeout • Delivery
Andheri Station (W)
Andheri West Station
Ticket Counter
Seth Pirojsha Ardesher
Patel Daremeher
Cama Rd
SV Rd
Nootan pure Veg
Delivery
Andheri
Mumbai
BSES MG Hospital
McDonald's
Temporarily closed
Bank of Baroda
Google
Map data ©2020
Terms of Use
Report a map error

For booking hotel user need to log in system .If user not register in the system then he/she need to register in the application .

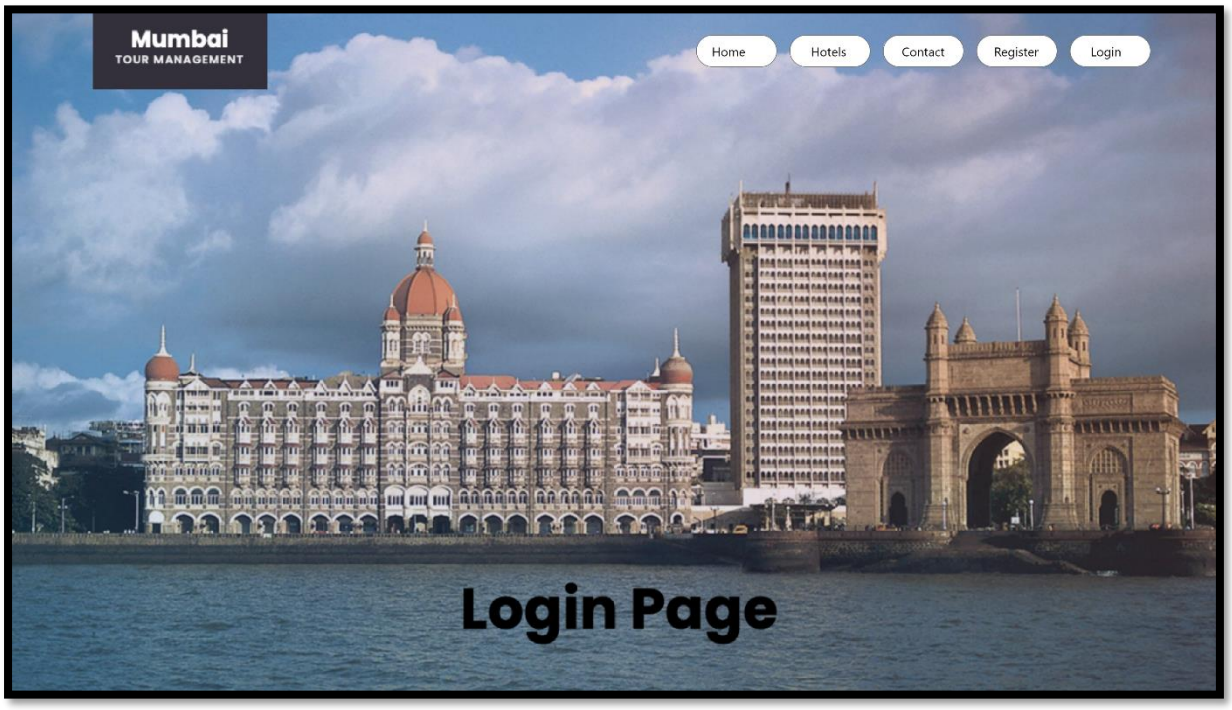


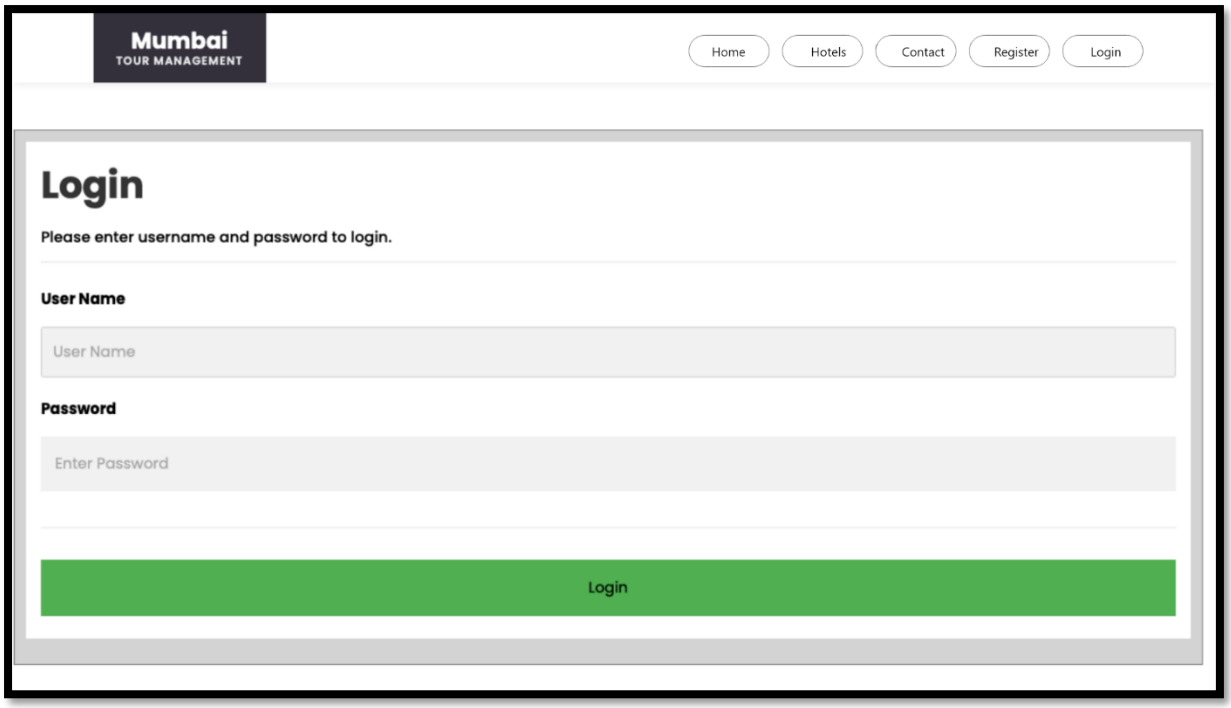
For registration user need to fill the details like First name, Last name User Name, Email as well as Password etc.



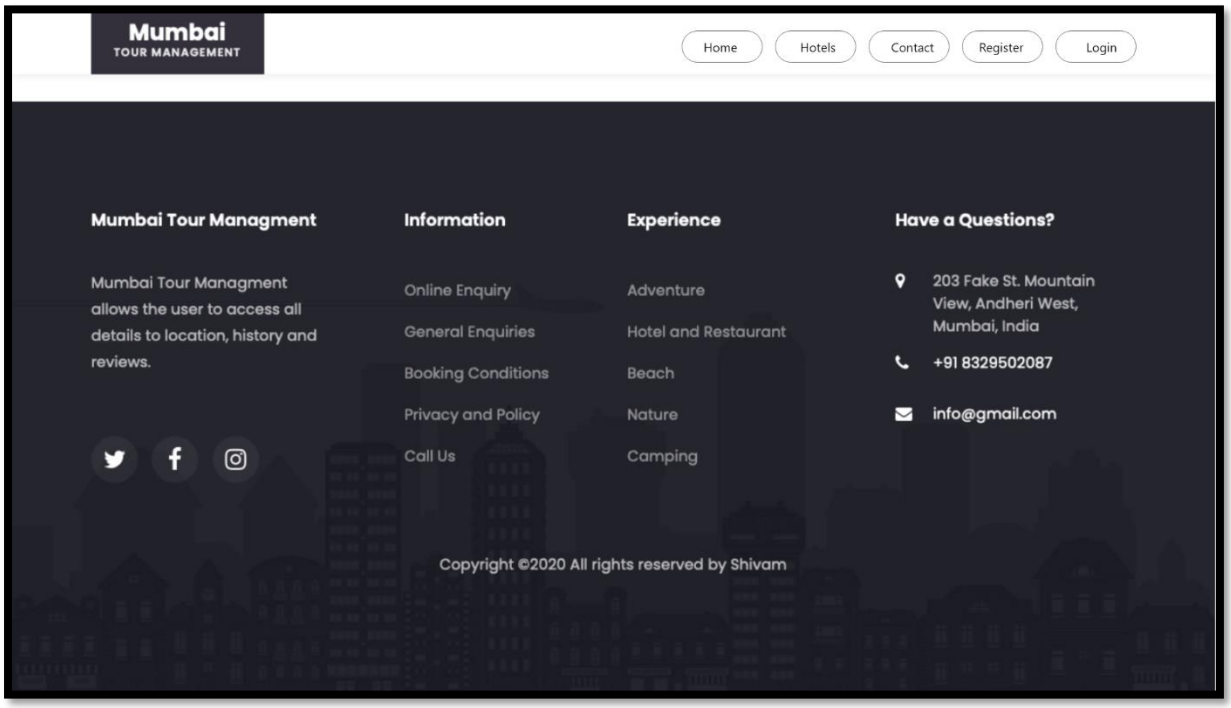


Once user register in system after that he/she can log in system for booking hotel.

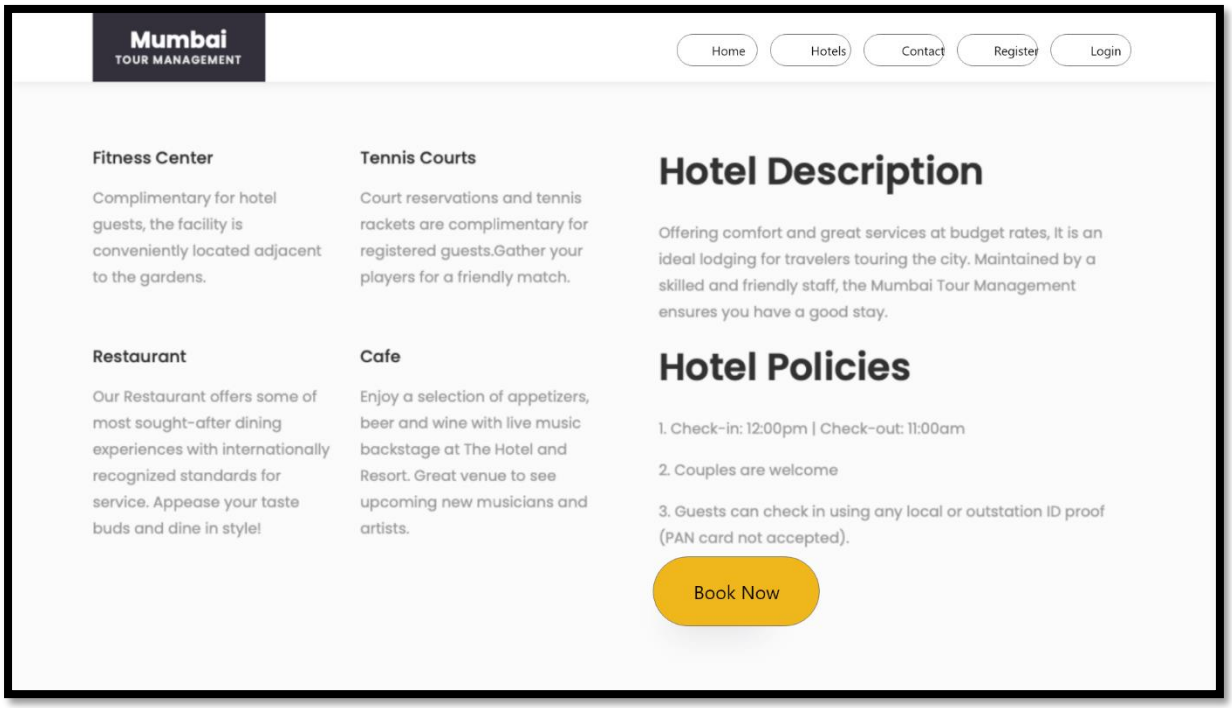




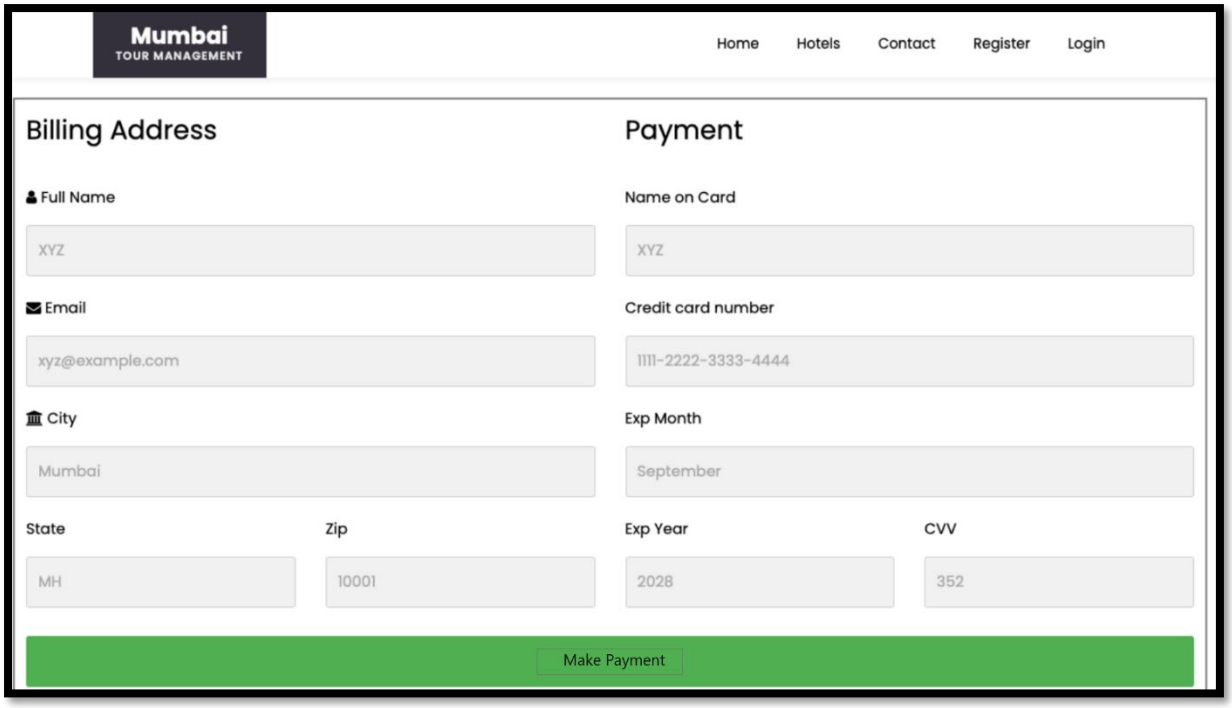
Contact Info



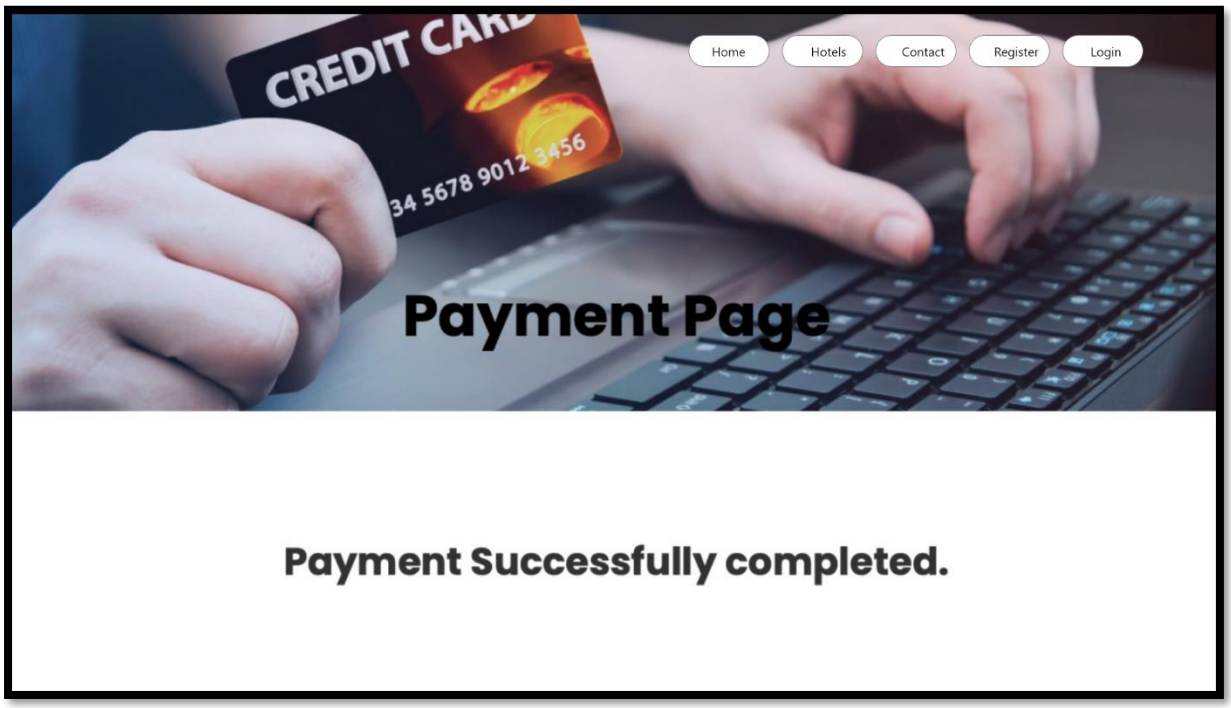
Click on Book Now Button for booking hotel.



After that user need to add details about payment like city, state, zip, Name on card , Credit card number, Exp month, Exp year and CVV etc.



When user click on make payment button if all required details are correct then payment will be successful.



Conclusion:

We can understand the MVC architectural pattern, as the model being the data, the controller being kind of the traffic controller, and the view being what the user sees and also interacts with.