PROJECT REPORT

TRAVEL AGENCY MANAGEMENT SYSTEM

ABSTRACT

Here We have developed a project on Travel agency management system. Online Tour and Travel Booking is a system that gives you the facility of booking any type of packages (Summer special packages, Manali tour packages, Shimla special packages, adventures etc.). This system is made so that customers can easily book tickets for all packages of tourist places such as hills, trekking, adventures, spirituals and users can also register for hotels for different types of room.

The project ‘Travel Agency Management system’ has been developed to replace the currently existing system, which helps in keeping records of the customer, details of destination as well as payment received. It saves the precious asset that is time, and also accuracy, reliability and uniformity can be maintained. This project is useful for the manager of the company as it helps them to search for the data faster than the existing system, to get customer records easily and reports of the customer's payment, etc. are generated as per requirement.

INTRODUCTION

Online Tour Agency Management is a very user-friendly project. In this project the user can easily understand and book all packages and also register for hotel. In this project there are various types of forms, like all types of packages, book for package and Room Registration Page in which user can insert his personal needs & facilities then upload his form by which user can register in the web site. Through this site, users can book different types of room such as deluxe rooms, ac room etc. at flexible prices.

This site provides different packages, destination, and hotel booking services in one state of India. In this website, according to the tour packages it provides services like online booking. facilities for all packages and hotels registration to the customer. In this project the user can easily understand and fill in the Room Reservation form. Users can search for various types of rooms.

Travel Agency Management is a web-based application made by Asp.net. So, with the help of this user can apply for packages of tour and room reservation.

Admin can also add travel packages and all details of the hotel.

This site gives new packages at flexible prices, so that users can book easily. According to season admin can change packages. We revise our itineraries regularly. Vacation-O-City offers a large range of travel opportunities. The Tours and Travels Project is designed in asp.net programming language. In this site, Admin can add packages, add destinations (adventures, spirituals), add hotels details, and delete hotel details. Admin can also add different types of rooms.

OBJECTIVES AND SCOPE

This system provides Online Registration, Tour Package Information, Flight ticket booking, Train ticket booking, Bus ticket booking, and all other destinations.

Our objective is to form strong relationships with customers so that they can enjoy the holiday of their dreams.

It saves lots of time, now a days there is a fast process world anyone can work quickly so, this project helps people who are searching for reservation in organization, so they can easily fill up the form and search rooms and different tourist place.

The scope for a Travel Agency Management System project typically includes features like booking management, customer information, itinerary planning, payment processing, and reporting. Additionally, consider user roles, security measures, and integration with external services like payment gateways and travel APIs. Define the project's objectives, constraints, and deliverables clearly to ensure a comprehensive system.

PROSPOSED PROJECT DESCRIPTION

The purpose of Travel Agency Management is mainly to provide online booking services for the user. The purpose of this site is to book online tickets for hotels and all tourist places.

The problem is that although we have many websites, they offer different kinds of services. The customers are enjoying themselves a lot but there is a lack of a relationship between travel agency and customers and hence we are establishing that relationship by carrying and servicing all customers in the same manner that we wish to serve.

In our Travel Agency Management project, we used Visual Studio 2022 and Visual Studio Code.

Front end design:

Front end refers to the visual layer of websites and applications. The UI, layout, images and other visual elements.

We have designed pages like login and register page, index page, about us and contact us.

Backend design:

Firstly, we created a Web API in Visual Studio 2022 with four endpoints like customer, hotel, package and booking.

As a next step created get, put, post, get by id and delete methods for all the endpoints using angular in Visual Studio 2022.

SYSTEM FEATURES

1. \*User Authentication: \*

- Secure login and registration for customers and administrators.

2. \*Customer Management: \*

- Profile creation and management for customers.

- View and edit personal information.

- Booking history and preferences tracking.

3. \*Hotel Booking: \*

- Search and filter hotels based on location, price, and amenities.

- Real-time availability and booking confirmation.

- Integration with payment gateways for secure transactions.

4. \*Package Management: \*

- Creation and customization of travel packages.

- Include options for flights, accommodations, activities, and transportation.

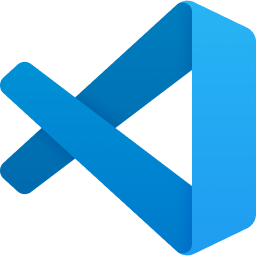
- Dynamic pricing based on selected components.

SOFTWARE REQUIREMENTS

VISUAL STUDIO CODE:

**Visual Studio Code**, also commonly referred to as **VS Codes** by [Microsoft](https://en.wikipedia.org/wiki/Microsoft) for [Windows](https://en.wikipedia.org/wiki/Windows), [Linux](https://en.wikipedia.org/wiki/Linux) and [macOS](https://en.wikipedia.org/wiki/MacOS).Features include support for [debugging](https://en.wikipedia.org/wiki/Debugging), [syntax highlighting](https://en.wikipedia.org/wiki/Syntax_highlighting), [intelligent code completion](https://en.wikipedia.org/wiki/Intelligent_code_completion), [snippets](https://en.wikipedia.org/wiki/Snippet_(programming)), [code refactoring](https://en.wikipedia.org/wiki/Code_refactoring), and embedded [Git](https://en.wikipedia.org/wiki/Git). Users can change the [theme](https://en.wikipedia.org/wiki/Theme_(computing)), [keyboard shortcuts](https://en.wikipedia.org/wiki/Keyboard_shortcut), preferences, and install [extensions](https://en.wikipedia.org/wiki/Plug-in_(computing)) that add functionality.

Learning to code is intimidating, so set yourself up for success with a tool built for you. Visual Studio Code is a free coding editor that helps you start coding quickly. Use it to code in any programming language, without switching editors. Visual Studio Code has support for many languages, including Python, Java, C++, JavaScript, and more. Ready to get started? Check out these [introductory videos](https://www.youtube.com/watch?v=Uvf2FVS1F8k&list=PLj6YeMhvp2S4oEV_bT1Uk3oXfpvGW6Xqy) or check out our [coding packs for Java and .NET](https://code.visualstudio.com/learn/educators/installers).



VISUAL STUDIO CODE 2022:

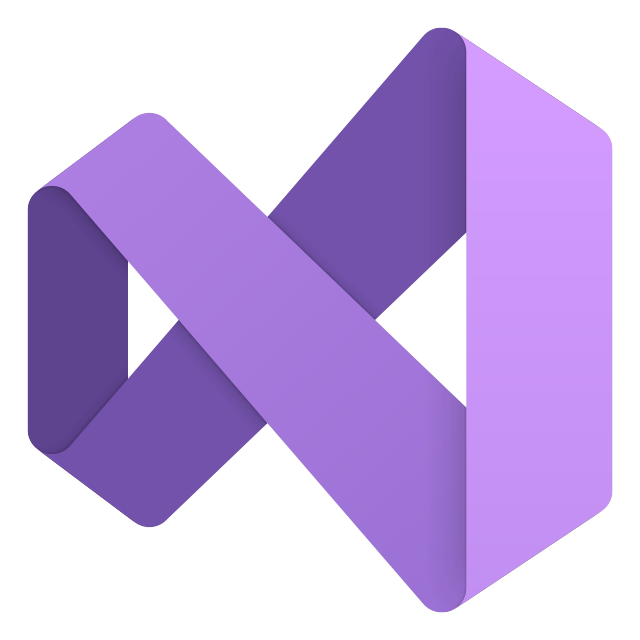
Visual Studio 2022 on Windows is now a 64-bit application. This means you can open, edit, run, and debug even the biggest and most complex solutions without running out of memory. To learn more, see both the [Visual Studio 2022 vision](https://devblogs.microsoft.com/visualstudio/visual-studio-2022/) and [Visual Studio 2022 17.0 Preview](https://devblogs.microsoft.com/visualstudio/visual-studio-2022-preview-1-now-available/) blog posts.

Visual Studio 2022 makes it quick and easy to build modern, cloud-based applications with Azure. Also, our new version also has full support for .NET and its unified framework for web, client, and mobile apps for both Windows and Mac developers. And, Visual Studio 2022 includes robust support for the C++ workload with new productivity features, C++20 tooling, and [IntelliSense](https://learn.microsoft.com/en-us/visualstudio/ide/using-intellisense?view=vs-2022).

[Visual Studio 2022](https://devblogs.microsoft.com/visualstudio/visual-studio-2022-preview-4-is-now-available/) includes a big update for the Blazor and [Razor](https://devblogs.microsoft.com/visualstudio/introducing-the-new-razor-editor-in-visual-studio-2022/) editors, and new capabilities for **Hot Reload** in ASP.NET Core—including **Hot Reload** when you save a file or when you apply changes to CSS files live!

* **Whole line completion**: In Visual Studio 2022, the [IntelliCode](https://learn.microsoft.com/en-us/visualstudio/intellicode/) feature can now automatically complete code up to a whole line at a time. For details, see the [**Type less, code more with IntelliCode completions**](https://devblogs.microsoft.com/visualstudio/type-less-code-more-with-intellicode-completions/) blog post.
* **Quick Actions recommendations**: [IntelliCode](https://learn.microsoft.com/en-us/visualstudio/intellicode/) can now spot when you’re performing a common task and recommend the right [Quick Action](https://learn.microsoft.com/en-us/visualstudio/ide/quick-actions?view=vs-2022), completing it right as you’re typing. To learn more, see the [**Discover quick actions for common tasks as you type, with IntelliCode**](https://devblogs.microsoft.com/visualstudio/discover-quick-action-intellicode/) blog post.

One of our key focus areas is to make Visual Studio more personalized and flexible so that you can make the IDE your own. For example, [Visual Studio 2022](https://devblogs.microsoft.com/visualstudio/visual-studio-2022-preview-3-now-available/) offers you the ability to sync with your Windows theme. So, if you've enabled the "night light" feature there, Visual Studio uses it, too. For more info, see the [**Personalize your Visual Studio 2022**](https://devblogs.microsoft.com/visualstudio/personalize-your-visual-studio-2022/) blog post.



**SYSTEM REQUIREMENTS**

The specific software and hardware requirements for a Training Management System can vary depending on the system's complexity, features, and the scale of deployment. It's essential to consult the specific documentation provided by the vendor for detailed software and hardware requirements. Below are general considerations for both software and hardware requirements:

**Software Requirements:**

**1. Operating System:**

The TMS should specify compatibility with a particular operating system. Common choices include Windows, Linux, or cloud-based solutions that operate independently of a specific OS.

**2. Database Management System (DBMS):**

Choose a compatible database system, such as MySQL, PostgreSQL, Microsoft SQL Server, or Oracle Database, depending on the TMS requirements.

**3. Programming Language:**

Identify the required programming language for customization and integration. Common choices include HTML,CSS,JS,Web Api, C#,Angular.

**4.Platforms:**

When selecting platforms for a Training Management System, it's important to consider the specific needs and requirements of the organization, as well as factors like scalability, user experience, and integration capabilities. We used platforms like Visual Studio, Visual Studio Code.

**5.Libraries:**

Before starting development, ensure that the necessary extensions and tools are installed in VS Code for your chosen language and framework. Here we use Entity Framework Core, ASP.NET Core MVC,Collections

**Hardware Requirements:**

**1. Memory (RAM):**

Sufficient RAM is critical for handling concurrent user sessions and ensuring smooth operation. The amount required depends on the application size and user based performance.

**2. Storage:**

Adequate storage space is necessary for storing the application, database, and associated files. Consider both the storage capacity and speed (SSD vs. HDD) based on performance requirements.

**3. Network Infrastructure:**

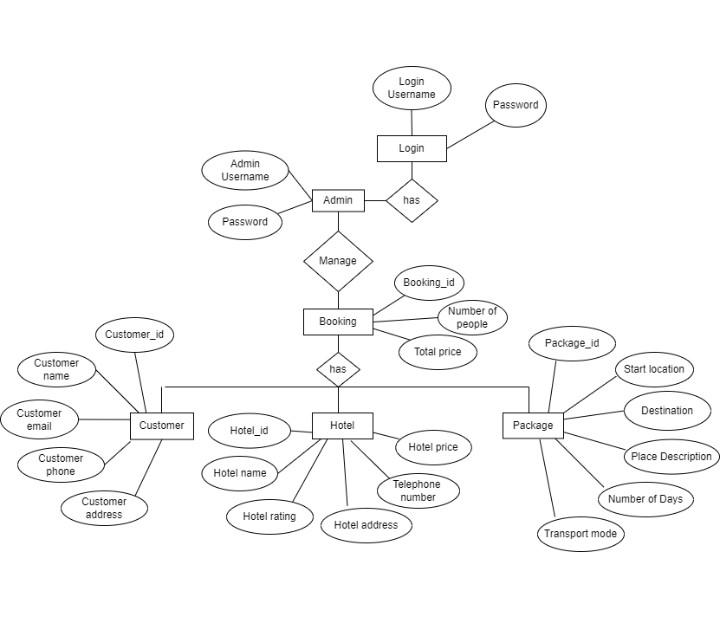
Ensure a reliable network infrastructure with sufficient bandwidth to handle data transfer between the TMS and user devices.

**4. Backup System:**

Implement a robust backup system to regularly back up the data and configurations.

E-R DIAGRAM

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.



IMPLEMENTATION

**Code Snippets:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Mvc;

using Microsoft.EntityFrameworkCore;

using TravelAgencyProjectWebApi.Data;

using TravelAgencyProjectWebApi.Models;

namespace TravelAgencyProjectWebApi.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class CustomerController : ControllerBase

{

private readonly TravelAgencyProjectWebApiContext \_context;

public CustomerController(TravelAgencyProjectWebApiContext context)

{

\_context = context;

}

// GET: api/Customer

[HttpGet]

public async Task<ActionResult<IEnumerable<Customer>>> GetCustomer()

{

if (\_context.Customer == null)

{

return NotFound();

}

return await \_context.Customer.ToListAsync();

}

// GET: api/Customer/5

[HttpGet("{id}")]

public async Task<ActionResult<Customer>> GetCustomer(Guid id)

{

if (\_context.Customer == null)

{

return NotFound();

}

var customer = await \_context.Customer.FindAsync(id);

if (customer == null)

{

return NotFound();

}

return customer;

}

// PUT: api/Customer/5

// To protect from overposting attacks, see https://go.microsoft.com/fwlink/?linkid=2123754

[HttpPut("{id}")]

public async Task<IActionResult> PutCustomer(Guid id, Customer customer)

{

if (id != customer.CustomerId)

{

return BadRequest();

}

\_context.Entry(customer).State = EntityState.Modified;

try

{

await \_context.SaveChangesAsync();

}

catch (DbUpdateConcurrencyException)

{

if (!CustomerExists(id))

{

return NotFound();

}

else

{

throw;

}

}

return NoContent();

}

// POST: api/Customer

// To protect from overposting attacks, see https://go.microsoft.com/fwlink/?linkid=2123754

[HttpPost]

public async Task<ActionResult<Customer>> PostCustomer(Customer customer)

{

if (\_context.Customer == null)

{

return Problem("Entity set 'TravelAgencyProjectWebApiContext.Customer' is null.");

}

\_context.Customer.Add(customer);

await \_context.SaveChangesAsync();

return CreatedAtAction("GetCustomer", new { id = customer.CustomerId }, customer);

}

// DELETE: api/Customer/5

[HttpDelete("{id}")]

public async Task<IActionResult> DeleteCustomer(Guid id)

{

if (\_context.Customer == null)

{

return NotFound();

}

var customer = await \_context.Customer.FindAsync(id);

if (customer == null)

{

return NotFound();

}

\_context.Customer.Remove(customer);

await \_context.SaveChangesAsync();

return NoContent();

}

private bool CustomerExists(Guid id)

{

return (\_context.Customer?.Any(e => e.CustomerId == id)).GetValueOrDefault();

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.EntityFrameworkCore;

using TravelAgencyProjectWebApi.Models;

namespace TravelAgencyProjectWebApi.Data

{

public class TravelAgencyProjectWebApiContext : DbContext

{

public TravelAgencyProjectWebApiContext (DbContextOptions<TravelAgencyProjectWebApiContext> options)

: base(options)

{

}

public DbSet<TravelAgencyProjectWebApi.Models.Customer> Customer { get; set; } = default!;

public DbSet<TravelAgencyProjectWebApi.Models.Hotel> Hotel { get; set; } = default!;

public DbSet<TravelAgencyProjectWebApi.Models.Package> Package { get; set; } = default!;

public DbSet<TravelAgencyProjectWebApi.Models.Booking> Booking { get; set; } = default!;

}

}

**Models In Web. API:**

Booking:

namespace TravelAgencyProjectWebApi.Models

{

public class Booking

{

public Guid BookingId { get; set; }

public int NumberOfPeople { get; set; }

public Double TotalPrice { get; set; }

public Hotel? Hotel { get; set; }

public Package? Package { get; set; }

public Customer? Customer { get; set; }

}

}

Customer:

namespace TravelAgencyProjectWebApi.Models

{

public class Customer

{

public Guid CustomerId { get; set; }

public string? CustomerName { get; set; }

public string? CustomerEmail { get; set; }

public string? CustomerPhone { get; set; }

public string? CustomerAddress { get; set; }

}

}

Package:

namespace TravelAgencyProjectWebApi.Models

{

public enum Transport

{

Train,

Bus,

Flight

}

public class Package

{

public Guid PackageId { get; set; }

public string? StartLocation { get; set; }

public string? Destination { get; set; }

public string? PlaceDescription { get; set; }

public int NumberOfDays { get; set; }

public Transport TransportMode { get; set; }

public Double Price { get; set; }

}

}

Hotel:

namespace TravelAgencyProjectWebApi.Models

{

public class Hotel

{

public Guid Hotelid { get; set; }

public string? Hotelname { get; set; }

public int HotelRating { get; set; }

public string? HotelAddress { get; set; }

public string? TelephoneNumber { get; set; }

public Double Hotelprice { get; set; }

}

}

Program.cs:

using Microsoft.EntityFrameworkCore;  
 using Microsoft.Extensions.DependencyInjection;  
 using TravelAgencyProjctWebAPI.Data;  
 var builder = WebApplication.CreateBuilder(args);  
 builder.Services.AddDbContext<TravelAgencyProjctWebAPIContext>(options =>  
 options.UseSqlServer(builder.Configuration.GetConnectionString("TravelAgencyProjctWebAPIContext") ?? throw new InvalidOperationException("Connection string 'TravelAgencyProjctWebAPIContext' not found.")));

// Add services to the container.

builder.Services.AddControllers();  
 // Learn more about configuring Swagger/OpenAPI at <https://aka.ms/aspnetcore/swashbuckle> builder.Services.AddEndpointsApiExplorer();  
 builder.Services.AddSwaggerGen();

var app = builder.Build();

// Configure the HTTP request pipeline.  
 if (app.Environment.IsDevelopment())  
 {  
 app.UseSwagger();  
 app.UseSwaggerUI();

}

app.UseHttpsRedirection();  
 app.UseCors(options =>  
 options.WithOrigins("http://localhost:4200")

.AllowAnyMethod()  
 .AllowAnyHeader());

app.UseAuthorization();

app.MapControllers();

app.Run();

FRONTEND CODE:

HomePage.component.html:

<!--Banner image-->

<div class="banner-container">

<img class="banner-img" src="https://globetrender.com/wp-content/uploads/2020/05/Caribbean-beach.jpg"

alt="Banner Image" />

<div class="banner-text">

<p id="banner-text-1">Explore and travel</p>

<p id="banner-text-2">TRAVEL THE WORLD WITH US</p>

<p id="banner-text-3">Book your vacation at GlobeTrek Tours</p>

</div>

</div>

<!-- intro -->

<div class="into-text">

<p id="intro1">Choose your</p>

<p id="intro2">Perfect holiday</p>

<p id="intro3"> Find your calling destination and travel the world, book a package and find out yourself with Glob

Trek tours!

</p>

</div>

<div class="home-package-gallery">

<div class="row">

<!-- Main Destination Image -->

<div class="col-3 destination">

<img src="https://www.fairobserver.com/wp-content/uploads/2022/11/shutterstock\_176969939.jpg"

alt="Destination" class="img-fluid destination">

<p>HIMALAYAS</p>

</div>

<!-- Places in the Tour -->

<div class="col-3 tour-place">

<img src="https://www.enfieldtoursindia.com/images/nubra-valley-leh-ladakh.jpg" alt="Leh Ladakh"

class="img-fluid places">

<p>Leh Ladakh</p>

</div>

<div class="col-3 tour-place">

<img src="https://img.onmanorama.com/content/dam/mm/en/travel/beyond-kerala/images/2020/6/11/pangong-lake.jpg.transform/576x300/image.jpg"

alt="Pangong Lake" class="img-fluid places">

<p>Pangong Lake</p>

</div>

<div class="col-3 tour-place">

<img src="https://www.bharatbooking.com/admin/webroot/img/uploads/holiday-package-gallery/1688803586\_201162-Exotic-Manali-Package-Slider-Image.webp"

alt="Manali" class="img-fluid places">

<p>Manali</p>

</div>

</div>

<div class="row">

<!-- Places in the Tour -->

<div class="col-3 tour-place">

<img src="https://a.cdn-hotels.com/gdcs/production56/d879/db09ee68-7ff8-4bfe-9611-bfba885f6b6f.jpg"

alt="ubud" class="img-fluid places">

<p>Ubud</p>

</div>

<div class="col-3 tour-place">

<img src="https://img.traveltriangle.com/blog/wp-content/uploads/2019/02/tanah-lot-Og.jpg" alt="Tanah Lot"

class="img-fluid places">

<p>Tanah Lot</p>

</div>

<div class="col-3 tour-place">

<img src="https://img.traveltriangle.com/blog/wp-content/uploads/2019/07/cover-Things-to-Do-In-Uluwatu.jpg"

alt="Uluwatu" class="img-fluid places">

<p>Uluwatu</p>

</div>

<!-- Main Destination Image -->

<div class="col-3 destination">

<img src="https://www.theknot.com/tk-media/images/0108e30b-f23f-4d5b-bacb-c13df8215c94~rs\_768.h"

alt="Destination" class="img-fluid destination">

<p>BALI</p>

</div>

</div>

<div class="row">

<!-- Main Destination Image -->

<div class="col-3 destination">

<img src="https://cdn.kimkim.com/files/a/content\_articles/featured\_photos/fbdebcfadec1546418836b917b252e75e678b6ba/big-a0f9c5581be5f2db211a093bdfedd2b4.jpg"

alt="Destination" class="img-fluid destination">

<p>KERALA</p>

</div>

<!-- Places in the Tour -->

<div class="col-3 tour-place">

<img src="https://miro.medium.com/v2/resize:fit:1000/1\*cWyYxjVyB80sUhUwV-hK5A.jpeg" alt="Munnar"

class="img-fluid places">

<p>Munnar</p>

</div>

<div class="col-3 tour-place">

<img src="https://www.authenticindiatours.com/app/uploads/2022/05/Kerala-Tour-Package-with-4-Star-Xandari-Resorts.jpg"

alt="Kochi" class="img-fluid places">

<p>Kochi</p>

</div>

<div class="col-3 tour-place">

<img src="https://meanderwander.com/wp-content/uploads/2022/11/munnar-to-alleppey-kerala-itinerary-.jpg"

alt="Allepey" class="img-fluid places">

<p>Allepey</p>

</div>

</div>

<div class="row">

<!-- Places in the Tour -->

<div class="col-3 tour-place">

<img src="https://www.esikkimtourism.in/wp-content/uploads/2019/04/3rd-image.jpg" alt="Gangtok"

class="img-fluid places">

<p>Gangtok</p>

</div>

<div class="col-3 tour-place">

<img src="https://hillscenic.co.in/images/p-destination/east/tsomgo.jpg" alt="Tsomgo Lake"

class="img-fluid places">

<p>Tsomgo Lake</p>

</div>

<div class="col-3 tour-place">

<img src="https://nomadicweekends.com/blog/wp-content/uploads/2019/03/Nathula-Pass.jpg" alt="Nathula Pass"

class="img-fluid places">

<p>Nathula Pass</p>

</div>

<!-- Main Destination Image -->

<div class="col-3 destination">

<img src="https://www.holidify.com/images/cmsuploads/compressed/silk-route-to-nathula-pass\_20220117160035.jpeg"

alt="Destination" class="img-fluid destination">

<p>SIKKIM</p>

<!-- </div> -->

</div>

</div>

<div class="greetings">

<img src="https://setsail.qodeinteractive.com/wp-content/uploads/2018/11/h1-img-1.png"

class="vc\_single\_image-img attachment-full" alt="greetings">

</div>

<div class="home-banner2">

<img

src="https://uhdwallpapers.org/uploads/converted/18/04/15/photographer-takes-pictures-in-nature-1920x1080\_84895-mm-90.jpg">

<p id="banner2-text1">Go and discover</p>

<p id="banner2-text2">BREATH TAKING CITIES</p>

<video id="banner2-video" controls autoplay>

<source src="/images/WhatsApp Video 2023-11-21 at 22.22.44.mp4" type="video/mp4">

<!-- </div> -->

<div class="row icons">

<div class="col-3">

<img src="https://setsail.qodeinteractive.com/wp-content/uploads/2018/09/h1-custom-icon-img-1.png"

alt="Restaurants" class="icon-img">

<h5 class="title">Restaurants</h5>

<p class="description">Aenean commodo ligula eget dolor. Lorem ipsum</p>

</div>

<div class="col-3">

<img src="https://setsail.qodeinteractive.com/wp-content/uploads/2018/09/h1-custom-icon-img-2.png"

alt="Sightseeing" class="icon-img">

<h5 class="title">Sightseeing</h5>

<p class="description">Aenean commodo ligula eget dolor. Lorem ipsum</p>

</div>

<div class="col-3">

<img src="https://setsail.qodeinteractive.com/wp-content/uploads/2018/09/h1-custom-icon-img-3.png"

alt="Shops & Boutiques" class="icon-img">

<h5 class="title">Shops & Boutiques</h5>

<p class="description">Aenean commodo ligula eget dolor. Lorem ipsum</p>

</div>

<div class="col-3">

<img src="https://setsail.qodeinteractive.com/wp-content/uploads/2018/09/h1-custom-icon-img-4.png"

alt="Where to Stay" class="icon-img">

<h5 class="title">Where to Stay</h5>

<p class="description">Aenean commodo ligula eget dolor. Lorem ipsum</p>

</div>

</div>

<div class="row testimony">

<div class="testimonial-item">

<p class="testimony-title">Our Customers say...</p>

<img src="https://restaurantindia.s3.ap-south-1.amazonaws.com/s3fs-public/content9670.jpg"

alt="Testimonial Image" class="testimonial-image">

<span class="testimonial-text">"We went on a Family vacation to bali. it was the best experience ever. planning

was detailed and organized." </span>

<span class="testimonial-author"> Nishant</span>

</div>

<div class="testimonial-item">

<span class="testimonial-text">"Loved the organisation of the entire tour plan, added time for relaxing too.

Thank you <b>GlobeTrek</b>.it was a great experience " </span>

<span class="testimonial-author"> Rudra Arthi</span><img

src="https://i.kinja-img.com/image/upload/c\_fit,q\_60,w\_645/c50066225960fc6fb30b1491cd50fc54.jpg"

alt="Testimonial Image" class="testimonial-image">

</div>

</div>

<footer>

<div class="footer-banner">

<p>Subscribe now and quench your wonderlust <button class="subscribe-btn">subscribe</button></p>

</div>

<div class="footer">

<div class="qodef-footer-top-inner qodef-full-width">

<div class="row">

<div class="col-3">

<h5 class="contact">Contact Info</h5>

<p>Email: [info@example.com</p](mailto:info@example.com</p)>

<p>Phone: +1 123 456 789</p>

<p>Address: 123 Main Street, Cityville</p>

</div>

<div class="col-3">

<div class="newsletter">News letter</div>

<div id="newsletter-form">

<form action="/subscribe" method="post" class="init" aria-label="Newsletter Subscription" novalidate="novalidate" data-status="init">

<div>

<span data-name="your-name">

<input size="25" class="wpcf7-form-control wpcf7-validates-as-required" aria-required="true" aria-invalid="false" placeholder="Your Name" value="" type="text" name="your-name">

</span>

</div>

<div>

<span data-name="your-email">

<input size="25" class="wpcf7-form-control wpcf7-validates-as-required wpcf7-validates-as-email" aria-required="true" aria-invalid="false" placeholder="Your Email" value="" type="email" name="your-email">

</span>

</div>

<input class="wpcf7-form-control subscribe-btn " type="submit" value="Subscribe">

</form>

</div>

</div>

<div class="col-3">

<div class="qodef-column-content qodef-grid-col-3">

<div class="widget qodef-blog-list-widget">

<h5 class="posts" style="margin-bottom: 26px">Our Recent Posts</h5>

<ul class="qodef-blog-list">

<li class="qodef-bl-item clearfix">

<h6 class="entry-title qodef-post-title">

<a href="https://example.com/visit-thailand-bali-and-china/" title="Visit Thailand, Bali And China">Visit Thailand, Bali And China</a>

</h6>

<div class="qodef-post-info-date entry-date published updated">

<a href="https://example.com/2016/09/">September 7, 2016</a>

<meta content="UserComments: 0">

</div>

</li>

<li class="qodef-bl-item clearfix">

<h6 class="entry-title qodef-post-title">

<a href="https://example.com/visit-thailand-bali-and-china/" title="Visit Thailand, Bali And China">Visit Thailand, Bali And China</a>

</h6>

<div class="qodef-post-info-date entry-date published updated">

<a href="https://example.com/2016/09/">September 7, 2016</a>

<meta content="UserComments: 0">

</div>

</li>

</ul>

</div>

</div>

</div>

<div class="col-3">

<div class="media">our Social media platforms</div>

<div class="social-media-icons">

<a href="https://www.instagram.com/" target="\_blank">

<i class="fab fa-instagram"></i> Instagram

</a>

<br>

<a href="https://www.youtube.com/" target="\_blank">

<i class="fab fa-youtube"></i> YouTube

</a>

<br>

<a href="https://twitter.com/" target="\_blank">

<i class="fab fa-twitter"></i> Twitter

</a>

<br>

<a href="https://www.facebook.com/" target="\_blank">

<i class="fab fa-facebook"></i> Facebook

</a>

<!-- Add more social media links and icons as needed -->

</div>

</div>

</div>

</div>

</div>

</footer>

Homepage.component.ts:

import { Component } from '@angular/core';

@Component({

selector: 'app-homepage',

templateUrl: './homepage.component.html',

styleUrls: ['./homepage.component.css']

})

export class HomepageComponent {

}

Regiser.component.ts:

import { Component, OnInit } from '@angular/core';

@Component({

selector: 'app-register',

templateUrl: './register.component.html',

styleUrls: ['./register.component.css']

})

export class RegisterComponent implements OnInit{

constructor(){}

ngOnInit(): void {

const inputs = document.querySelectorAll(".input-field");

const toggle\_btn = document.querySelectorAll(".toggle");

const main = document.querySelector("main");

const bullets = document.querySelectorAll(".bullets span");

const images = document.querySelectorAll(".image");

inputs.forEach((inp: any) => {

inp.addEventListener("focus", () => {

inp.classList.add("active");

});

inp.addEventListener("blur", () => {

if (inp.value != "") return;

inp.classList.remove("active");

});

});

toggle\_btn.forEach((btn: any) => {

btn.addEventListener("click", () => {

main.classList.toggle("sign-up-mode");

});

});

function moveSlider(this: any) {

let index = this.dataset.value;

console.log(index);

let currentImage = document.querySelector(`.img-${index}`);

console.log(currentImage);

images.forEach((img: any) => img.classList.remove("show"));

currentImage.classList.add("show");

const textSlider = document.querySelector(".text-group");

//textSlider.style.transform = `translateY(${-(index - 1) \* 2.2}rem)`;

bullets.forEach((bull: any) => bull.classList.remove("active"));

this.classList.add("active");

}

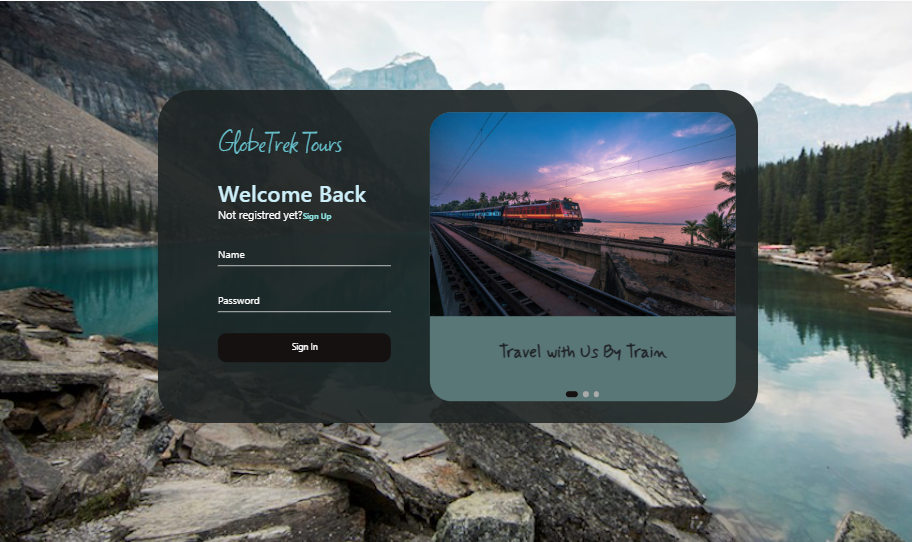
bullets.forEach((bullet: any) => {

bullet.addEventListener("click", moveSlider);

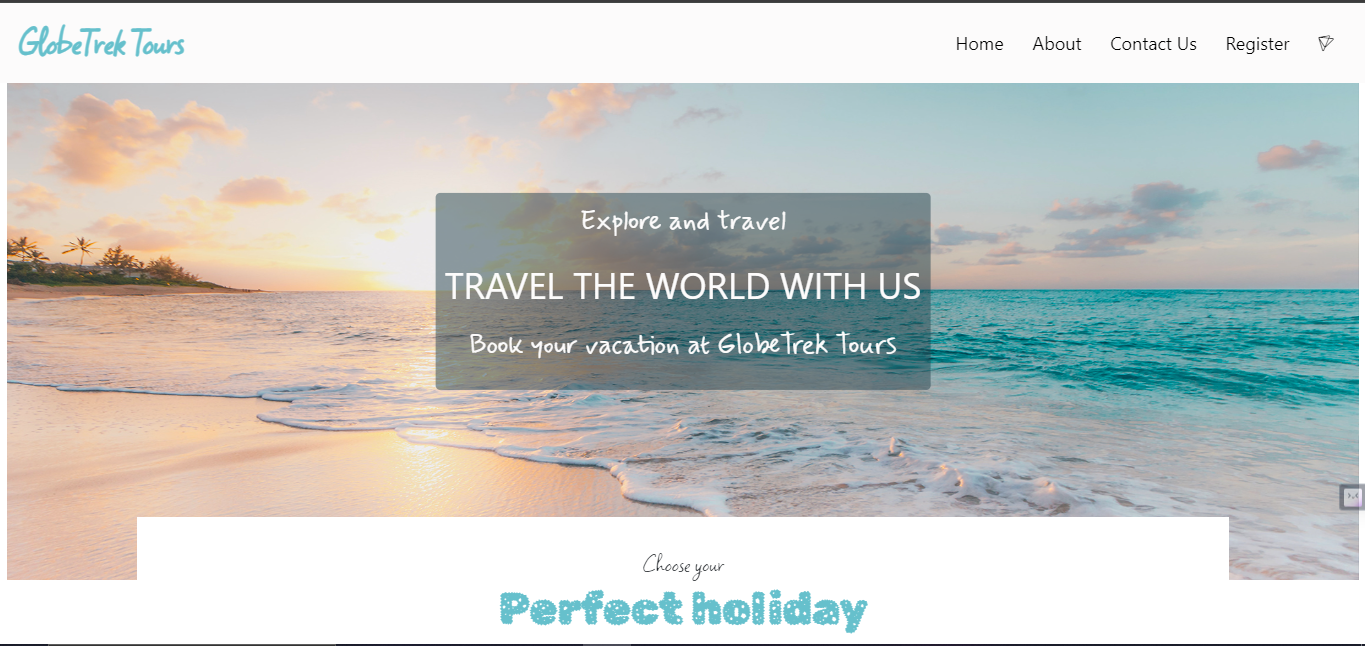
});

}}

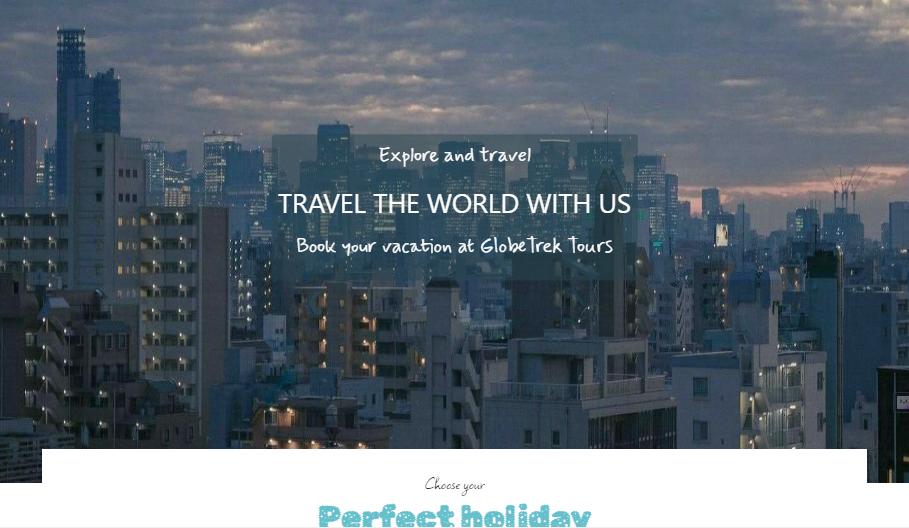
OUTPUT SCREENSHOT

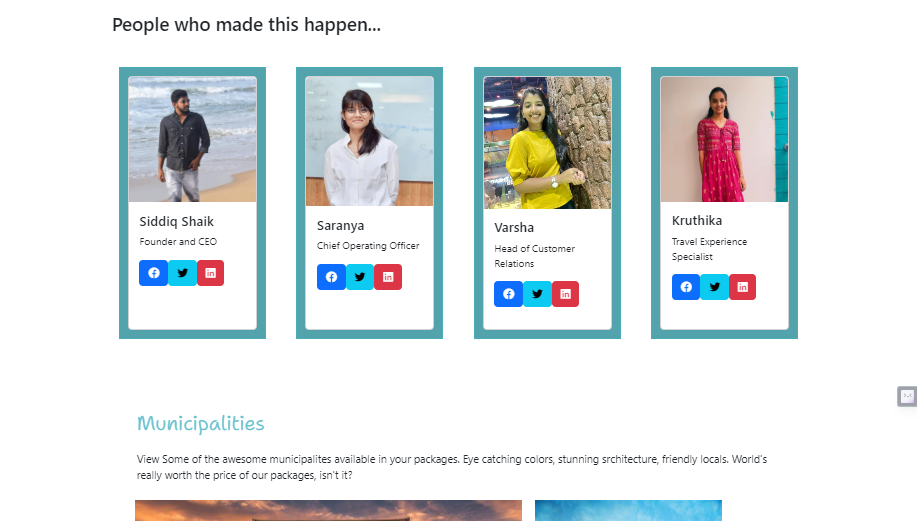


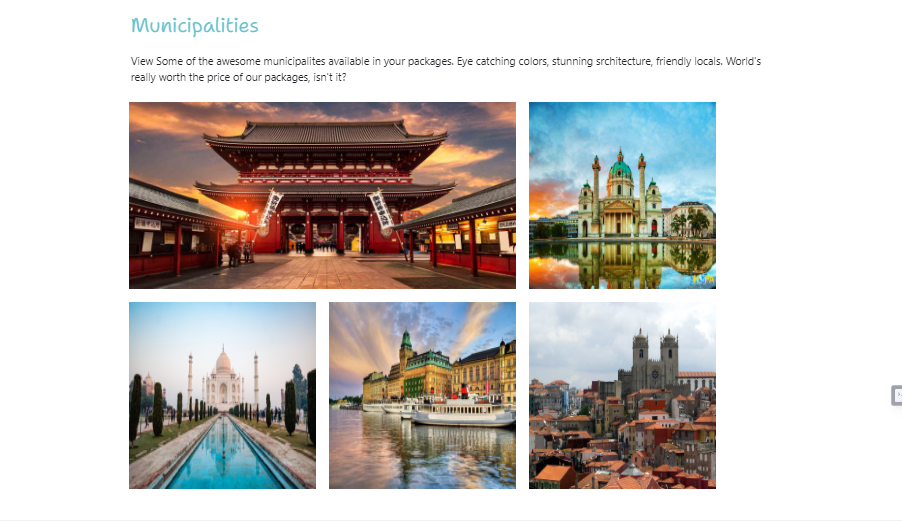
Login and Register page

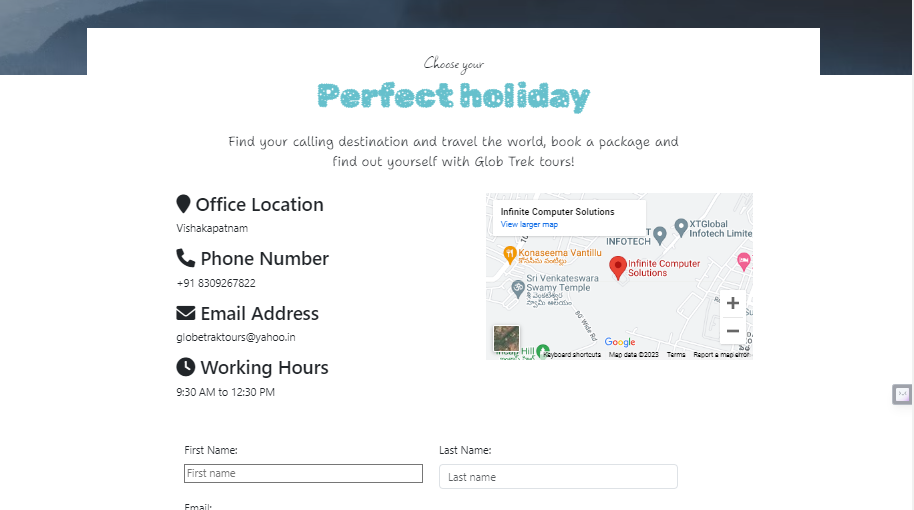


Home Page

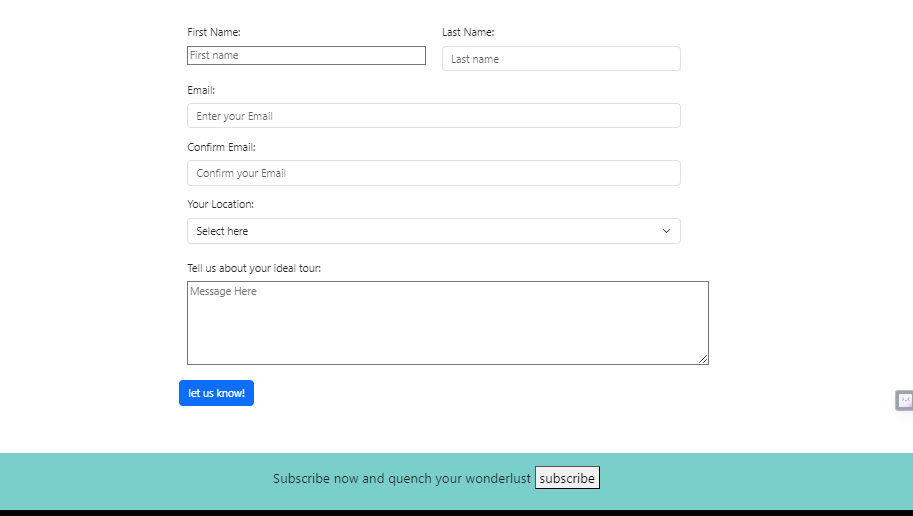
 About Us Page

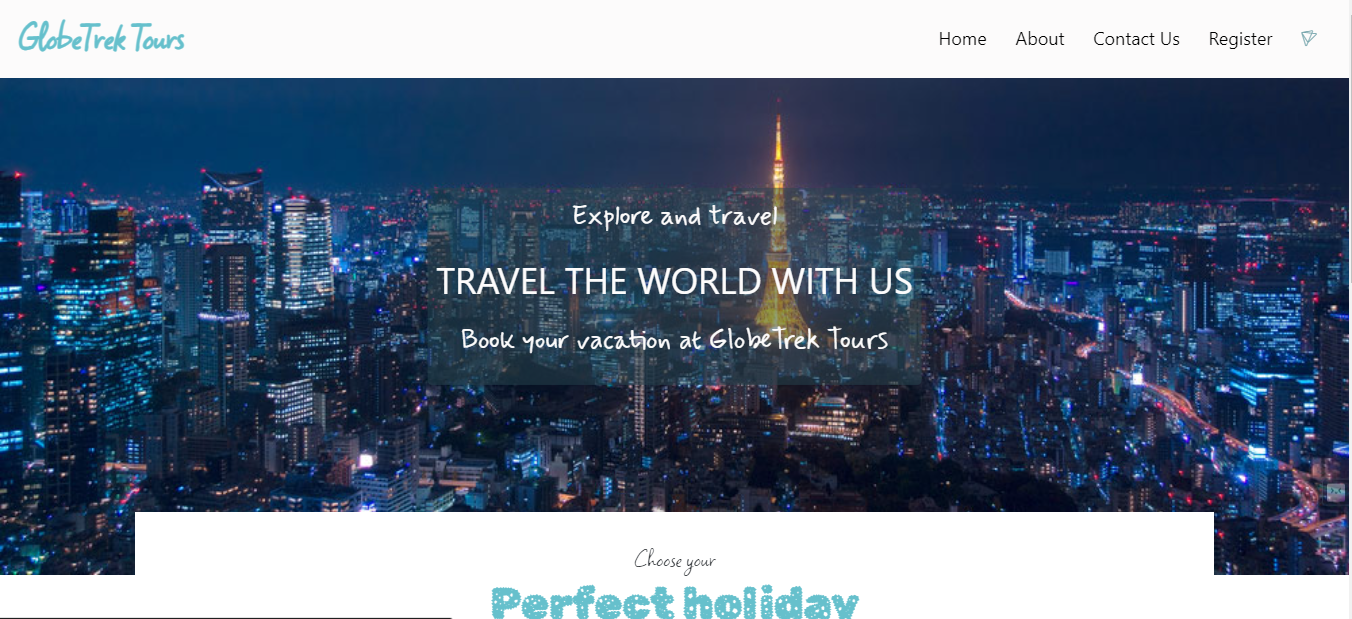


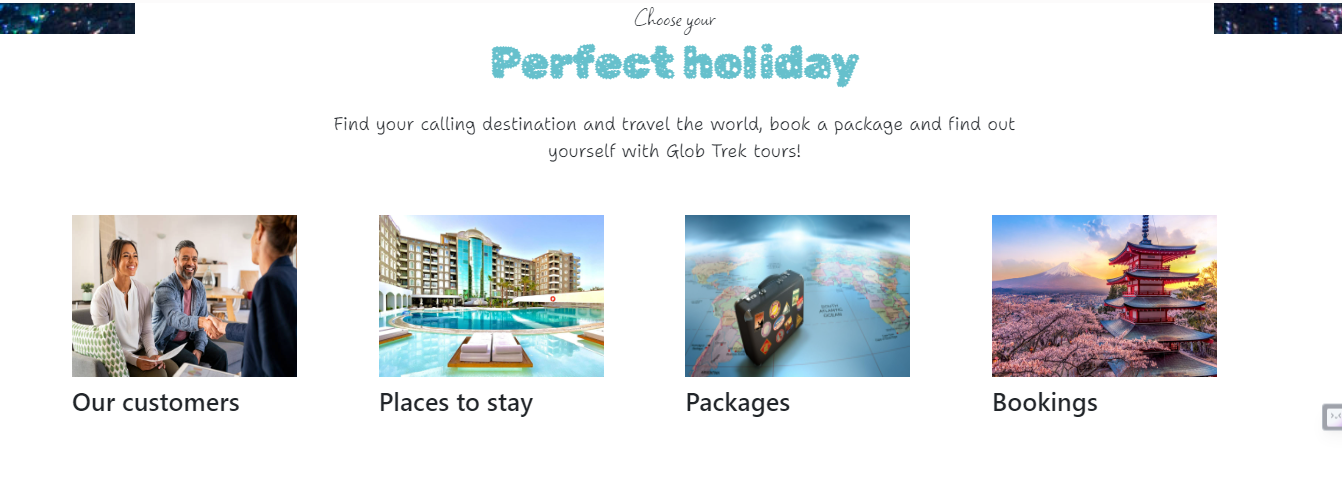




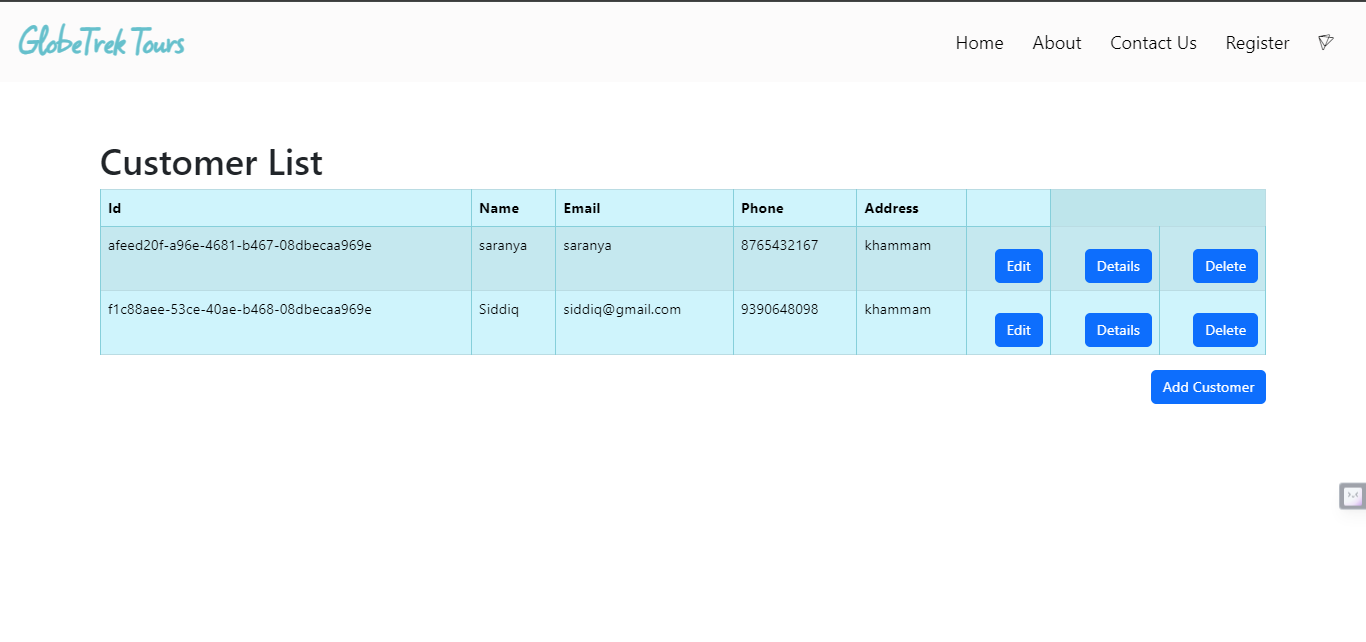
Contact US



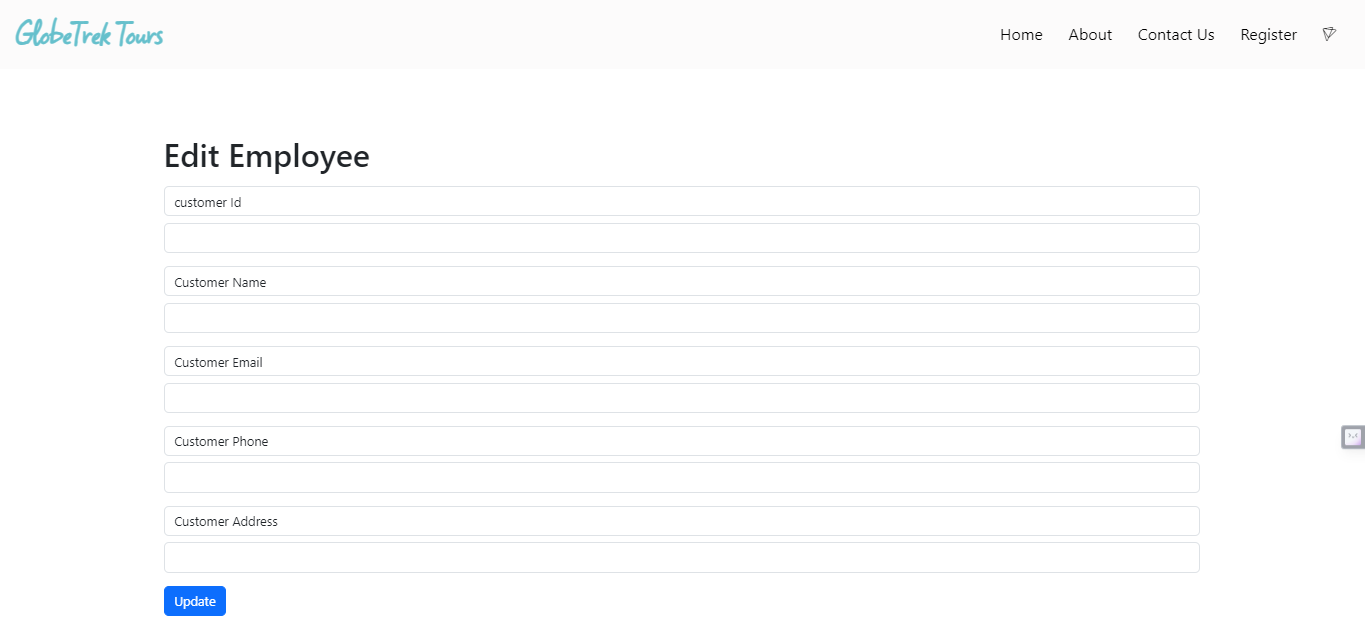




Admin Page



Backend Pages



CONCLUSION

In summary, the Travel Agency Management System represents a pivotal evolution in the landscape of travel industry operations. This sophisticated system seamlessly integrates a myriad of functionalities to orchestrate a harmonious symphony of booking management, itinerary customization, and robust communication channels. As travel agencies navigate the dynamic and competitive realm of travel, this comprehensive solution emerges as a linchpin for success.

By leveraging cutting-edge technology, the system not only streamlines internal processes but also enhances the overall customer experience. The automation of routine tasks not only reduces the margin for error but also frees up valuable time for travel agents to focus on personalized service and strategic decision-making. Real-time updates and information accessibility empower both agents and customers, fostering a sense of transparency and reliability.

Moreover, the Travel Agency Management System is not merely a tool for administrative efficiency; it is a catalyst for growth. Through data analytics and reporting features, agencies can glean valuable insights into customer preferences and market trends, enabling them to tailor their services and stay ahead of the curve.

In the ever-evolving landscape of the travel industry, adaptability is key. The Travel Agency Management System stands as a testament to the transformative power of technology in reshaping traditional paradigms. As it continues to evolve, it promises to be a cornerstone for travel agencies seeking not only to survive but to thrive in an environment that demands agility, innovation, and a steadfast commitment to customer satisfaction.