**Core Functionalities**

## Home Page Front end:

Home page code consist of further sub elements which are given after main home page:

import React from 'react';

import Navbar from '../Navbar/HomeNavbar';

import Hero from './Hero';

import Search from './Search';

import Book from './Book';

import Discover from './Discover';

import About from './About';

const Home = () => {

return (

<>

<Navbar />

<Hero />

<Search />

<Book />

<Discover />

<About />

</>

);

};

export default Home;

* **Nav Bar:**

import React, { useState } from 'react';

import { Link } from 'react-scroll';

import Auth from '../../utils/auth';

// import icons

import { Bars2Icon, XMarkIcon } from '@heroicons/react/24/solid';

const HomeNavbar = () => {

const [nav, setNav] = useState(false);

const handleNav = () => setNav(!nav);

const logout = event => {

event.preventDefault();

Auth.logout();

};

return (

<nav name="home" className="navbar">

<div className={nav ?

'bg-[#f8f8f8de] w-full h-[80px] absolute px-4 flex justify-between items-center transition-all ease-in-out delay-100 duration-300 z-[5]'

:

'w-full h-[80px] absolute px-4 flex justify-between items-center text-white z-[5]'

}>

<div className="logo">

<h2 className="font-semibold uppercase">

<a href="/" className="text-4xl hover:text-teal-200">

Dream Escape

</a>

</h2>

</div>

<ul className="hidden md:flex">

<Link to="home" smooth={true} duration={500}><li>Home</li></Link>

<Link to="search" smooth={true} duration={500}><li>Search</li></Link>

<Link to="book" smooth={true} duration={500}><li>Book</li></Link>

<Link to="discover" smooth={true} duration={500}><li>Discover</li></Link>

<Link to="about" smooth={true} duration={500}><li>Our Story</li></Link>

</ul>

<div className="nav-icons hidden md:flex">

<ul className="hidden md:flex justify-center items-center">

{Auth.loggedIn() ? (

<>

<a href="/blog"><li>Blog</li></a>

<a href="/profile"><li className="mr-2">Profile</li></a>

<a href="/" onClick={logout}><li className="logout-btn">Logout</li></a>

</>

) : (

<>

<a href="/blog"><li>Blog</li></a>

<a href="/login"><li className="login-btn">Login</li></a>

<a href="/signup"><li className="signup-btn">Signup</li></a>

</>

)}

</ul>

</div>

{/\* HAMBURGER MENU \*/}

<div className="hamburger-menu md:hidden hover:cursor-pointer" onClick={handleNav}>

{!nav ? (<Bars2Icon width={30} />) : (<XMarkIcon width={30} />)}

</div>

<div className={nav ? 'mobile-menu active' : 'mobile-menu'}>

<ul className="primary my-4">

<Link to="home" smooth={true} duration={500}><li>Home</li></Link>

<Link to="book" smooth={true} duration={500}><li>Book</li></Link>

<Link to="discover" smooth={true} duration={500}><li>Discover</li></Link>

<Link to="about" smooth={true} duration={500}><li>Our Story</li></Link>

<a href="/blog"><li>Blog</li></a>

<a href="/profile"><li>Profile</li></a>

</ul>

<ul className="secondary my-4">

{Auth.loggedIn() ? (

<>

<a href="/" onClick={logout}><li className="logout-btn">Logout</li></a>

</>

) : (

<>

<a href="/login"><li className="login-btn">Login</li></a>

<a href="/signup"><li className="signup-btn">Signup</li></a>

</>

)}

</ul>

</div>

</div>

</nav>

);

};

export default HomeNavbar;

* **Hero:**

import React from 'react';

import { Link } from 'react-scroll';

import Video from '../../assets/hero.mov';

const Hero = () => {

return (

<>

<div className="hero w-full h-screen relative text-white">

<video

autoPlay

loop

muted

id="video"

className="w-full h-full object-cover z-[-5]"

>

<source src={Video} type="video/mp4" />

</video>

<div className="overlay bg-[#00000016]"></div>

<div className="content w-full h-full m-auto p-4 absolute top-0 flex flex-col justify-center items-center">

<h1 className="text-2xl md:text-5xl">An dream of possibilities</h1>

<h2 className="my-4 text-xl md:text-4xl">The whole world awaits</h2>

<Link to="search" smooth={true} duration={500}>

<button>Get Started</button>

</Link>

</div>

</div>

</>

);

};

export default Hero;

* **Search:**

import React from 'react';

import { MapPin, Phone } from 'react-feather';

import Rating from '@mui/material/Rating'

const PlaceDetails = ({ place, selected, refProp }) => {

if (selected) refProp?.current?.scrollIntoView({ behavior: "smooth", block: "start" })

return (

<div className="w-full card-wrapper">

<div className="h-[350px]">

<img

src={place?.photo ? place.photo.images.larg.url : 'https://images.unsplash.com/photo-1514933651103-005eec06c04b?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxzZWFyY2h8MTh8fHJlc3RhdXJhbnR8ZW58MHx8MHx8&auto=format&fit=crop&w=500&q=60'}

alt={place.name}

/>

</div>

<div className="place-details">

<h2>{place.name}</h2>

<div className="flex justify-between">

<div><Rating value={Number(place.rating)} readOnly /></div>

<div>out of {place.num\_reviews}</div>

</div>

<div className="flex justify-between">

<div>Price</div>

<div>{place.price\_level}</div>

</div>

<div className="flex justify-between">

<div>Ranking</div>

<div>{place.ranking}</div>

</div>

{place?.cuisine?.map(({ name }) => (

<button key={name} label={name} className="py-2 px-1 rounded-full bg-gray-100" />

))}

{place?.address && (

<div>

<MapPin width={18} /> {place.address}

</div>

)}

{place?.phone && (

<div>

<Phone width={18} /> {place.phone}

</div>

)}

</div>

<div className="card-links">

{place?.web\_url && (

<button onClick={() => window.open(place.web\_url, '\_blank')}>

Trip Advisor

</button>

)}

{place?.website && (

<button onClick={() => window.open(place.website, '\_blank')}>

Website

</button>

)}

</div>

</div>

);

};

export default PlaceDetails;

* **Booking:**

import React from 'react';

import Data from './Data/Book-Data';

const Book = () => {

const destinations = Data;

// display return date if `round trip` is selected

const roundTripHandleChange = event => {

document.getElementById('return-date').style.visibility =

event.checked && event.id === 'round-trip' ? 'hidden' : 'visible';

};

// hide return date if `one way` is selected

const oneWayHandleChange = event => {

document.getElementById('return-date').style.visibility =

event.checked && event.id === 'one-way' ? 'visible' : 'hidden';

};

return (

<div name="book" className="book w-full h-full md:h-screen relative p-8 mb-6">

<div className="w-full md:max-w-screen-lg h-full mx-auto flex flex-col justify-center items-center">

<h1>

Spend Less. Travel More.

</h1>

<div className="w-full md:max-w-screen-lg flex flex-row flex-wrap justify-center items-center">

{/\* SEARCH DESTINATIONS \*/}

<div className="w-full md:max-w-screen-md">

<form className="search-form w-full mx-auto grid grid-cols-1 md:grid-cols-3 mb-4">

<div className="col-span-1 md:col-span-3">

<input

type="radio"

id="round-trip"

name="trip-type"

value="round-trip"

onChange={roundTripHandleChange}

defaultChecked

/>

<label for="round-trip" className="pl-2 pr-6">

Round-Trip

</label>

<input

type="radio"

id="one-way"

name="trip-type"

value="one-way"

onChange={oneWayHandleChange}

/>

<label for="one-way" className="pl-2">

One-Way

</label>

</div>

<div className="grid grid-cols-1 md:grid-cols-3 col-span-1 md:col-span-3 gap-4 py-4 place-items-stretch">

<div>

<label className="block">From</label>

<input type="text" placeholder="Origin city or airport" />

</div>

<div>

<label className="block">To</label>

<input

type="text"

placeholder="Destination city or airport"

/>

</div>

<div>

<label className="block">Passengers</label>

<select>

<option value="1">1 Passenger</option>

<option value="1">2 Passengers</option>

<option value="1">3 Passengers</option>

<option value="1">4 Passengers</option>

<option value="1">5 Passengers</option>

<option value="1">6 Passengers</option>

<option value="1">7 Passengers</option>

<option value="1">8 Passengers</option>

<option value="1">9 Passengers</option>

</select>

</div>

</div>

<div className="grid grid-cols-1 md:grid-cols-3 col-span-1 md:col-span-3 gap-4 pb-4 place-items-stretch">

<div id="departure-date" className="w-full">

<label className="block">Departure Date</label>

<input type="date" />

</div>

<div id="return-date" className="w-full">

<label className="block">Return Date</label>

<input type="date" />

</div>

<div className="pt-6">

<button className="primary w-full h-[43px]">Find Your Trip</button>

</div>

</div>

</form>

</div>

{/\* SEARCH DEALS \*/}

<div className="w-full">

<h2 className="mb-8 text-teal-400 text-center">

Don't miss these deals

</h2>

</div>

<div className="images grid grid-cols-1 md:grid-cols-3 gap-2 md:gap-4">

{destinations.map((city) => (

<div className="max-w-sm rounded-md overflow-hidden shadow-lg hover:shadow-slate-400 transition-all ease-in duration-300">

<img src={city.image} alt={city.imgAlt} className="w-full" />

<div className="p-4">

<div className="flex justify-between items-center">

<div className="font-semibold text-xl">{city.name}</div>

<div className="text-teal-400 font-semibold text-xl">

{city.price}

</div>

</div>

<div className="flex justify-between items-center text-gray-500 text-xs">

<div>{city.deal}</div>

<div>per person</div>

</div>

<button className="primary w-full mt-4">View Deal</button>

</div>

</div>

))}

</div>

</div>

</div>

</div>

);

};

export default Book;

* **Discover:**

import React from 'react';

import Data from './Data/Discover-Data';

const Discover = () => {

const destinations = Data;

return (

<div name="discover" className="discover w-full h-full md:h-screen relative p-8">

<div className="w-full md:max-w-screen-lg h-full mx-auto flex flex-col justify-center items-center">

<h1 className='mt-6'>Discover Dreamy Destinations</h1>

<div className="grid grid-cols-1 md:grid-cols-3 gap-2 md:gap-4">

{destinations.map((city) => (

<div className="images relative group max-w-sm rounded-md overflow-hidden opacity-80 hover:opacity-100 transition-all ease-in-out duration-300">

<div className="overlay"></div>

<img src={city.image} alt={city.name} />

<div className="absolute top-0 left-0 h-full w-full flex flex-col justify-center items-center p-2 text-white z-[5]">

<h2 className="text-2xl md:text-xl font-semibold drop-shadow">

{city.name}

</h2>

<div>

<a href="#">Flights</a> &#8226;&nbsp;

<a href="#">Hotels</a> &#8226;&nbsp;

<a href="#">Cars</a>

</div>

</div>

{/\* GRADIENT OVERLAY ON HOVER \*/}

<div className="flex justify-center items-center transition-all ease-in duration-300 opacity-0 bg-gradient-to-t from-transparent via-gray-700 to-opacity-20 group-hover:opacity-60 absolute top-0 left-0 h-full w-full"></div>

</div>

))}

</div>

</div>

</div>

);

};

export default Discover;

* **About:**

import React from "react";

const About = () => {

return (

<div

name="about"

className="about md:bg-about bg-cover bg-top w-full h-full md:h-screen"

>

<div className="w-full md:max-w-screen-lg mx-auto p-8 flex flex-col justify-center z-[2]">

<h1 className="mt-6">Bringing the Dreams to the world</h1>

<p className="my-4 text-center">

Dream Escape was founded in 2024 by a Dream Escape Team

</p>

<div className="flex flex-wrap justify-evenly items-start md:bg-[#ffffff8a] rounded-lg leading-loose">

<div className="w-full md:w-1/2 p-4">

At Dream Escape, we're more than just a booking platform – we're

your partners in exploration, dedicated to unlocking the world's

wonders one journey at a time. Founded on the belief that travel has

the power to enrich lives and broaden horizons, we strive to connect

travelers with authentic experiences that leave a lasting

impression. With a commitment to excellence and a passion for

adventure, we've earned a reputation as a trusted authority in the

travel industry, helping countless explorers turn their wanderlust

into reality.

</div>

<div className="w-full md:w-1/2 p-4">

Driven by a relentless pursuit of excellence, our team of travel

enthusiasts is dedicated to crafting seamless and unforgettable

experiences for every traveler. Whether you're embarking on a solo

adventure, planning a romantic getaway, or organizing a group

expedition, we're here to make your travel dreams come true. From

meticulously curated itineraries to personalized recommendations,

we're with you every step of the way, ensuring your journey is

nothing short of extraordinary. Join us on a journey of discovery

and let's create memories that will last a lifetime.

</div>

</div>

</div>

</div>

);

};

export default About;

* **Footer:**

import React from 'react';

//import icons

import { Facebook, Instagram, Twitter, Youtube, Heart } from 'react-feather';

const Footer = () => {

return (

<footer className="footer mt-auto w-full px-4 py-8 bg-gray-800 text-white">

<div className="w-full">

<div className="upper">

<h3 className="text-teal-200 text-xl font-semibold uppercase">

Dream Escape

</h3>

<div className="social">

<Facebook size={25} className="icon" />

<Instagram size={25} className="icon" />

<Twitter size={25} className="icon" />

<Youtube size={25} className="icon" />

</div>

</div>

<div className="lower">

<div className="left">

<ul>

<li>Privacy & Terms</li>

<li>Policy</li>

<li>Help Center</li>

<li>Partnership</li>

<li>Contact</li>

</ul>

</div>

<div className="right">

<ul>

<li>&copy; 2024 Dream Escape. All rights reserved.</li>

<a

href="https://github.com/demivlkv/trip-tide"

target="\_blank"

className="flex justify-center hover:text-teal-200 cursor-pointer transition-all ease-in duration-500"

>

<li className="inline-flex">

Made with <Heart size={15} className="text-teal-200 mx-1" />{" "}

by Dream Escape Team.

</li>

</a>

</ul>

</div>

</div>

</div>

</footer>

);

};

export default Footer;

## Log In Page:

import React, { useState } from "react";

import { Link } from "react-router-dom";

import { useMutation } from "@apollo/client";

import { LOGIN\_USER } from "../utils/mutations";

import Auth from "../utils/auth";

import Layout from "../components/Layout/Dashboard";

import {

AtSymbolIcon,

LockClosedIcon,

ArrowLongRightIcon,

} from "@heroicons/react/24/solid";

const Login = (props) => {

const [formState, setFormState] = useState({ email: "", password: "" });

const [login, { error }] = useMutation(LOGIN\_USER);

// update state based on form input changes

const handleChange = (event) => {

const { name, value } = event.target;

setFormState({ ...formState, [name]: value });

};

// submit form

const handleFormSubmit = async (event) => {

event.preventDefault();

try {

const { data } = await login({

variables: { ...formState },

});

console.log(data);

Auth.login(data.login.token);

} catch (e) {

console.error(e);

}

};

return (

<Layout>

<div className="w-full h-screen">

<div className="login bg-login bg-fixed bg-cover w-full h-full flex flex-col justify-center items-center px-4 ">

<div className="mb-8 flex flex-col justify-center items-center text-white z-[1]">

<h1>Welcome Back</h1>

<h2>There’s a whole world out there</h2>

</div>

<div className="card">

<form className="w-full max-w-[500px]" onSubmit={handleFormSubmit}>

<div className="relative">

<div className="icons">

<AtSymbolIcon width={20} />

</div>

<input

placeholder="Your email address"

name="email"

type="email"

id="email"

value={formState.email}

onChange={handleChange}

/>

</div>

<div className="relative">

<div className="icons">

<LockClosedIcon width={20} />

</div>

<input

placeholder="Password"

name="password"

type="password"

id="password"

value={formState.password}

onChange={handleChange}

/>

</div>

<button type="submit" className="primary">

Log In

</button>

<div className="w-full flex justify-center items-center text-sm leading-loose">

Don't have an account?

<Link

to="/signup"

className="inline-flex items-center ml-2 text-teal-400 hover:text-teal-200"

>

Sign up{" "}

<ArrowLongRightIcon

width={25}

className="inline-flex items-center ml-1"

/>

</Link>

</div>

</form>

{error && (

<div className="text-sm text-slate-600 italic">Login failed</div>

)}

</div>

<div className="w-full h-full absolute top-0 left-0 bg-gradient-to-r opacity-80 from-teal-100 via-slate-600 to-sky-800 z-0"></div>

</div>

</div>

</Layout>

);

};

export default Login;

## Sign up Page:

import React, { useState } from 'react';

import { Link } from 'react-router-dom';

import { useMutation } from '@apollo/client';

import { ADD\_USER } from '../utils/mutations';

import Auth from '../utils/auth';

import Layout from '../components/Layout/Dashboard';

import { UserIcon, AtSymbolIcon, LockClosedIcon, ArrowLongRightIcon, MapPinIcon, BriefcaseIcon } from '@heroicons/react/24/solid';

const Signup = () => {

const [formState, setFormState] = useState({ username: '', email: '', password: '', location: '', description: '' });

const [addUser, { error }] = useMutation(ADD\_USER);

// update state based on form input changes

const handleChange = (event) => {

const { name, value } = event.target;

setFormState({ ...formState, [name]: value });

};

// submit form

const handleFormSubmit = async (event) => {

event.preventDefault();

try {

// execute addUser mutation & pass in variable data from form

const { data } = await addUser({

variables: { ...formState },

});

console.log(data);

Auth.login(data.addUser.token);

} catch (e) {

console.error(e);

}

};

return (

<Layout>

<div className="w-full h-screen">

<div className="signup bg-signup bg-fixed bg-cover w-full h-full flex flex-col justify-center items-center px-4">

<div className="mb-8 flex flex-col justify-center items-center text-white z-[1]">

<h1>Sign Up</h1>

<h2>Join the travel community</h2>

</div>

<div className="card">

<form className="w-full max-w-[500px]" onSubmit={handleFormSubmit}>

<div className="relative">

<div className="icons">

<UserIcon width={20} />

</div>

<input

placeholder="Enter username"

name="username"

type="username"

id="username"

value={formState.username}

onChange={handleChange}

/>

</div>

<div className="relative">

<div className="icons">

<AtSymbolIcon width={20} />

</div>

<input

placeholder="Enter email address"

name="email"

type="email"

id="email"

value={formState.email}

onChange={handleChange}

/>

</div>

<div className="relative">

<div className="icons">

<LockClosedIcon width={20} />

</div>

<input

placeholder="Password"

name="password"

type="password"

id="password"

value={formState.password}

onChange={handleChange}

/>

</div>

<div className="relative">

<div className="icons">

<MapPinIcon width={20} />

</div>

<input

placeholder="Location"

name="location"

type="location"

id="location"

value={formState.location}

onChange={handleChange}

/>

</div>

<div className="relative">

<div className="icons">

<BriefcaseIcon width={20} />

</div>

<input

placeholder="Tell us a little about yourself"

name="description"

type="description"

id="description"

value={formState.description}

onChange={handleChange}

/>

</div>

<button type="submit" className="primary">

Create Account

</button>

<div className="w-full flex justify-center items-center text-sm leading-loose">

Already a member?

<Link to="/login" className="inline-flex items-center ml-2 text-teal-400 hover:text-teal-200">

Login <ArrowLongRightIcon width={25} className="inline-flex items-center ml-1" />

</Link>

</div>

</form>

{error && <div className="text-sm text-slate-600 italic">Sign up failed</div>}

</div>

<div className="w-full h-full absolute top-0 left-0 bg-gradient-to-r opacity-80 from-teal-100 via-slate-600 to-sky-800 z-0"></div>

</div>

</div>

</Layout>

);

};

export default Signup;

**User Registration:**

* The user registration feature enables individuals to create accounts on the platform, allowing them to access individual services, save preferences and interact with system users.
* Users are provided with a registration form with required information such as username, email address, password, full name, date of birth, contact details, places of interest to set the feed and other optional information.

**Login:**

* Registered users should be able to log in using their username/email/contact number and credentials(password).
* Password reset mechanism should be provided in case user forget their password.

**Customization:**

* Customization feature allows the users the options to customize tour packages according to their preferences.
* Customization option may include selecting additional services such as:
* Accommodation
* Transportation
* Car rental
* Meal plans
* Guided tour
* Recreational Activity
* Taxis
* Lodging
* Room service

**Search and Browsing:**

* The searching feature allow the users to search for tour based on criteria such as location, type, price range, etc.
* Users would be able to browse through the catalog of tours available on the platform.

**Filtering:**

* The filtering feature allows users to filter and narrow down search results or content based on specific criteria, preferences, or criteria.
* It enhances the user experience by helping users quickly find the right information or content that suits their wants and needs.

**Booking:**

* Booking feature will allows to securely book tours through this platform.
* Configuration email will be sent to users upon successful booking.
* Users will have access to their booking history for reference.

**Transparent Pricing:**

* Pricing information for each tour package will be clearly displayed.
* The breakdown of costs, including accommodation, transportation, meals, additional services would be provided.

**Car rental:**

* The car rental feature allows users to browse and rent vehicles for their travel needs, providing flexibility and convenience in transportation options.

**Show Map:**

* Use an interactive map view to explore sightseeing spots, show geographic locations, landmarks, and points of interest.
* Allow users to browse trips based on geography, explore interactive maps to visualize travel routes and destinations. Use an interactive map view to explore sightseeing spots, show geographic locations, landmarks, and points of interest.

**Multi-language:**

* Provide support for multiple languages to handle users from different languages.
* Allow users to choose their preferred language to browse tours, compose and receive communications.

**Admin panel:**

* Admin would be provided for authorized personnel to manage tours, user accounts, bookings, etc.
* Admin would be able to view analytic and reports to gain insights into platform usage and performance.

**Chat with Agent:**

* The chat with agent enables users to engage in real-time conservation with customer support agents or representative for assistance, inquires, or support.

**Help Center:**

* The Help Center feature is a comprehensive knowledge base and self-service platform where users can access relevant information, resources, and instructions on using the platform, troubleshooting common problems, and navigating various services.
* Collection of informative articles, guides, tutorials, and how-to items covering a wide range of topics related to platform usage, features, policies and frequently asked questions (FAQs) will be provided.

**Feedback:**

* Feedback enables users to provide feedback, ideas, and suggestions directly to platform administrators or the customer support team.
* It provides a platform for users to express their views, share their experiences and contribute to the continuous improvement of the platform.