**BIL 332 - Bilgi Güvenliği Dersi**

**2022-2023 Bahar Yarıyılı Final Ödev Çalışması**

**Arama Motoru Web Uygulaması**

**Furkan KARATEKİN**

**1911012034**

**Bilgisayar Mühendisliği**

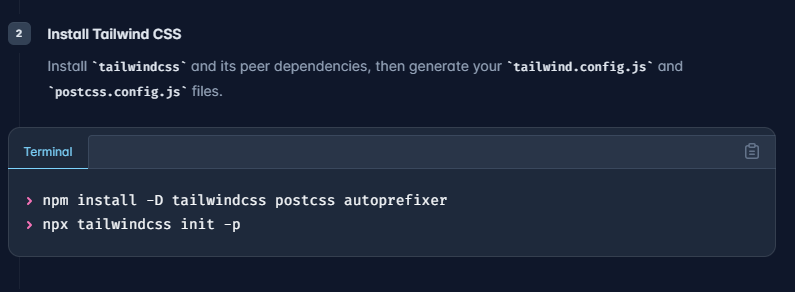
**Isparta – 2023**

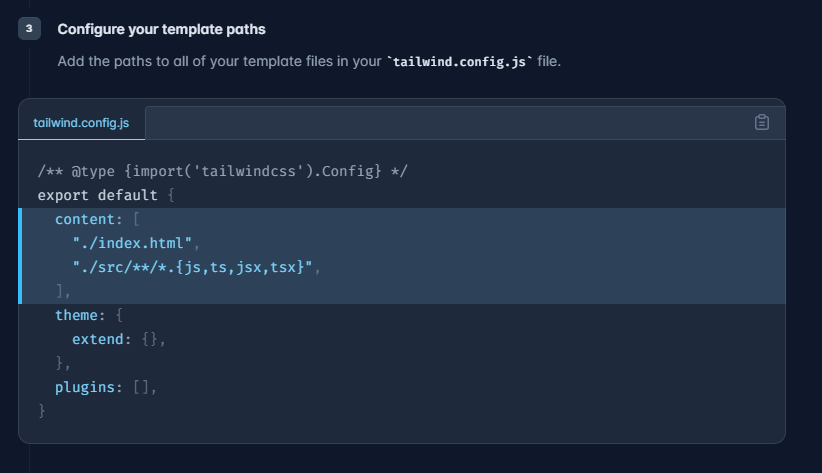
FRONT-END KISMI KURULUM VE GEREKLİ PAKETLERİ YÜKLEME

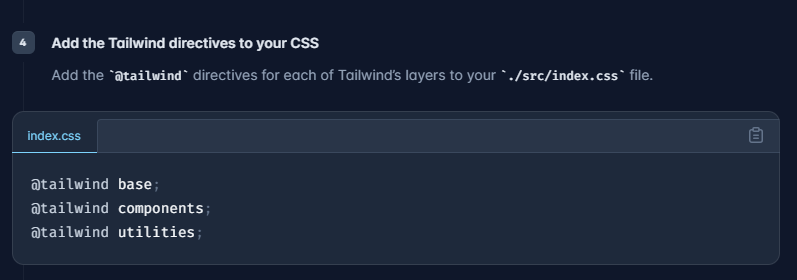


Bu üstteki kodu konsolda yazarak temel kurulum işlemlerine başladık. Artık temel bir React projemiz oluştu

Bundan sonraki adımda ise TailwindCss’i projeme entegre etmem gerekiyordu. Sırası ile aşağıdaki adımları uyguladım.

  
  
Bu kodları konsola yazdıktan sonra projeme entegre etmem gerekliydi.





Artık kurulumumuz hazır.Temel bir ReactJs ve TailwindCss için gerekli ortam oluşturuldu.

FRONT-END TARAFINDA KULLANILAN PAKETLER



"@react-google-maps/api": "^2.19.2"

Bu paketi haritalar kısımı için projeme yükledim. @react-google-maps/api paketi, React tabanlı web uygulamalarında Google Haritalar'ı entegre etmeyi sağlayan bir kütüphanedir. Bu kütüphane, React uygulamalarına dinamik harita bileşenleri eklemek ve bu haritaları kullanarak etkileşimli ve özelleştirilebilir harita deneyimleri oluşturmak için kullanılır.

Bu paketin temel özellikleri şunları içerebilir:

Harita Bileşenleri

Harita Olayları

Marker Yönetimi

Harita Özellikleri ve Kontrolleri

Google Haritalar API Entegrasyonu

"axios": "^1.6.2",

axios paketi, tarayıcı ve Node.js üzerinde HTTP istekleri yapmak için kullanılan bir JavaScript kütüphanesidir.

Temel özellikleri şunlardır:

HTTP İstekleri

Promise Tabanlı

İnterceptor Desteği

JSON Veri Desteği

Hata Yönetimi

"@reduxjs/toolkit": "^1.9.5"

Redux'u daha etkili kullanmayı sağlayan bir araç ve yardımcı fonksiyonlar içeren resmi Redux Toolkit. Tip tanımlamalarını, reducer'ları daha kolay oluşturmayı sağlar, immutable state yönetimi ve diğer Redux görevlerini basitleştirir.

"firebase": "^10.3.1"

Firebase, Google tarafından sunulan bir mobil ve web uygulama geliştirme platformudur. Bu paket, Firebase platformu ile etkileşimde bulunmayı sağlar. Firebase, oturum açma, gerçek zamanlı veritabanı, depolama, kullanıcı yetkilendirmesi, bulut fonksiyonları gibi bir dizi hizmet sunar. Bu paket, bu hizmetlere kolay erişim sağlamak ve Firebase'i projelerinizde kullanmak için kullanılır.

"flowbite-react": "^0.7.2"

Flowbite, React tabanlı bir UI kütüphanesidir.

Modern ve şık tasarıma sahip bileşenleri içerir, projelerde kullanılmak üzere önceden tasarlanmış UI öğeleri sunar.

"formik": "^2.4.5"

Formik, React projelerinde form yönetimini kolaylaştırmak için kullanılan bir kütüphanedir.

Form oluşturmayı, doğrulamayı, değer yönetimini ve form durumunu yönetmeyi sağlar.

"react-icons": "^4.10.1"

React projelerinde kullanılmak üzere çeşitli ikon setlerini içeren bir kütüphanedir.

"react-redux": "^8.1.2"

React uygulamalarında Redux'u entegre etmeyi sağlayan bir pakettir.

Redux'un state yönetimiyle React bileşenlerini birleştirir.

"react-router-dom": "^6.15.0"

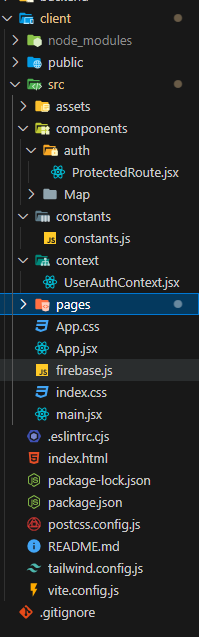
React projelerinde sayfa yönlendirmesi ve gezinme sağlamak için kullanılan bir pakettir.

"redux-persist": "^6.0.0":

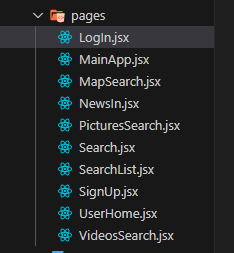
Redux store'unu kalıcı depolama çözümü sağlayan bir pakettir.

Uygulama durduğunda veya yeniden başladığında verilerin korunmasına olanak tanır.

FRONT-END DOSYALARI



Gerekli paketleri ve eklentileri kurduktan sonra projede ilk adım gerekli sayfaları page klasörü altında oluşturdum

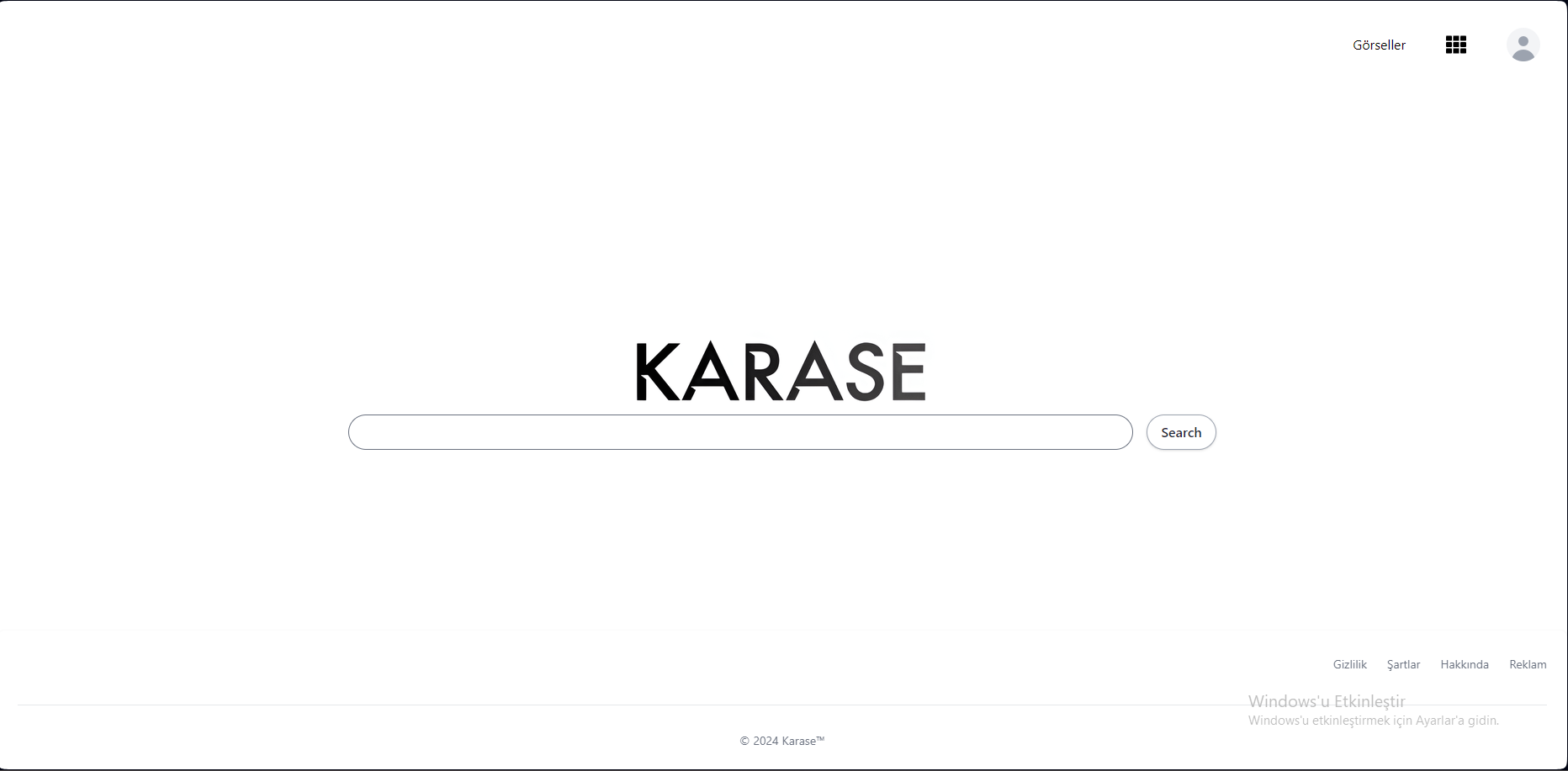


PROJE ROUTER İŞLEMLERİ



PAGES

Arama sayfasında Sizi karşılayan Bölüm



Kodları :

import React, { useState } from "react";

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

import { Avatar } from "flowbite-react";

import { FaTh } from "react-icons/fa";

import { Footer } from "flowbite-react";

const Search = () => {

  const [searchText, setSearchText] = useState("");

  const navigate = useNavigate();

  const handleKeyPress = (event) => {

    if (event.key === "Enter") {

      // Enter tuşuna basıldığında "/list" sayfasına yönlendirme yapılır

      navigate("/list");

    }

  };

  return (

    <div className="App h-screen">

      {/\* en üst sağ kısım \*/}

      <div className="flex flex-wrap gap-2 justify-end mb-80">

        <Link to="pictures">

          <div className="md:mt-10 md:mr-10 mt-5 mr-5 hover:hover:text-gray-400">

            <p>Görseller</p>

          </div>

        </Link>

        <Link to="ucnokta">

          <div>

            <FaTh className=" text-2xl md:mt-10 md:mr-10 mt-5 mr-5 hover:text-gray-400 " />

          </div>

        </Link>

        <Link to="login">

          <div>

            <Avatar

              className="md:mt-8 md:mr-8 mt-4 mr-4  hover:scale-125 transition-transform"

              rounded

            />

          </div>

        </Link>

      </div>

      {/\* google büyük yazan kısım \*/}

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

        <img src="../src/assets/logoSearch.png" alt="" />

      </div>

      {/\*  input ve search buton kısım \*/}

      <div className="flex items-center justify-center mb-8">

        <input

          type="text"

          name="name"

          className="p-2 w-2/3 md:w-1/2 border border-gray-500 rounded-3xl"

          value={searchText}

          onChange={(e) => setSearchText(e.target.value)}

          onKeyDown={handleKeyPress}

        />

        <Link to="list">

          <button className="ml-4 px-4 bg-white hover:bg-gray-100 text-gray-800 font-semibold py-2 px-4 border border-gray-400 rounded-3xl shadow">

            Search

          </button>

        </Link>

      </div>

      {/\* footer \*/}

      <Footer container className="fixed bottom-0">

        <div className="w-full text-center">

          <div className="w-full justify-between sm:flex sm:items-center sm:justify-between">

            <Footer.Brand />

            <Footer.LinkGroup>

              <Footer.Link href="#">Gizlilik</Footer.Link>

              <Footer.Link href="#">Şartlar</Footer.Link>

              <Footer.Link href="#">Hakkında</Footer.Link>

              <Footer.Link href="#">Reklam</Footer.Link>

            </Footer.LinkGroup>

          </div>

          <Footer.Divider />

          <Footer.Copyright href="" by="Karase™" year={2024} />

        </div>

      </Footer>

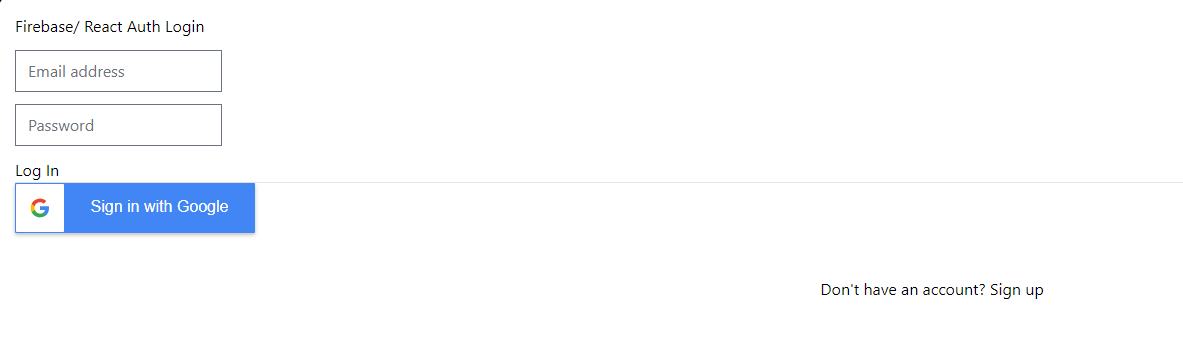
    </div>

  );

};

export default Search;

Kullanıcı Girişi Bölümü



Kullanıcı Girişi Kodları:

import React, { useState } from "react";

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

import { Form, Alert } from "react-bootstrap";

import { Button } from "react-bootstrap";

import GoogleButton from "react-google-button";

import { useUserAuth } from "../context/UserAuthContext";

const Login = () => {

  const [email, setEmail] = useState("");

  const [password, setPassword] = useState("");

  const [error, setError] = useState("");

  const { logIn, googleSignIn } = useUserAuth();

  const navigate = useNavigate();

  const handleSubmit = async (e) => {

    e.preventDefault();

    setError("");

    try {

      await logIn(email, password);

      navigate("/user-home");

    } catch (err) {

      setError(err.message);

    }

  };

  const handleGoogleSignIn = async (e) => {

    e.preventDefault();

    try {

      await googleSignIn();

      navigate("/user-home");

    } catch (error) {

      console.log(error.message);

    }

  };

  return (

    <>

      <div className="p-4 box">

        <h2 className="mb-3">Firebase/ React Auth Login</h2>

        {error && <Alert variant="danger">{error}</Alert>}

        <Form onSubmit={handleSubmit}>

          <Form.Group className="mb-3" controlId="formBasicEmail">

            <Form.Control

              type="email"

              placeholder="Email address"

              onChange={(e) => setEmail(e.target.value)}

            />

          </Form.Group>

          <Form.Group className="mb-3" controlId="formBasicPassword">

            <Form.Control

              type="password"

              placeholder="Password"

              onChange={(e) => setPassword(e.target.value)}

            />

          </Form.Group>

          <div className="d-grid gap-2">

            <Button variant="primary" type="Submit">

              Log In

            </Button>

          </div>

        </Form>

        <hr />

        <div>

          <GoogleButton

            className="g-btn"

            type="dark"

            onClick={handleGoogleSignIn}

          />

        </div>

      </div>

      <div className="p-4 box mt-3 text-center">

        Don't have an account? <Link to="/signup">Sign up</Link>

      </div>

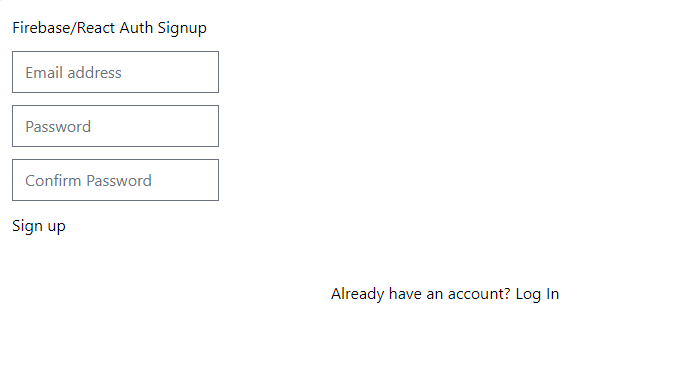
    </>

  );

};

export default Login;

Üye Ol Kısımı:



Üye Ol Kodları:

import React, { useState } from "react";

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

import { Form, Alert } from "react-bootstrap";

import { Button } from "react-bootstrap";

import { useUserAuth } from "../context/UserAuthContext";

const Signup = () => {

  const [email, setEmail] = useState("");

  const [username, setUsername] = useState("");

  const [password, setPassword] = useState("");

  const [confirmPassword, setConfirmPassword] = useState("");

  const [error, setError] = useState("");

  const { signUp } = useUserAuth();

  let navigate = useNavigate();

  const handleSubmit = async (e) => {

    e.preventDefault();

    setError("");

    if (password !== confirmPassword) {

      setError("Passwords do not match.");

      return;

    }

    try {

      await signUp(email, password); // You may want to update this to include the username

      navigate("/login");

    } catch (err) {

      setError(err.message);

    }

  };

  return (

    <>

      <div className="p-4 box">

        <h2 className="mb-3">Firebase/React Auth Signup</h2>

        {error && <Alert variant="danger">{error}</Alert>}

        <Form onSubmit={handleSubmit}>

          <Form.Group className="mb-3" controlId="formBasicEmail">

            <Form.Control

              type="email"

              placeholder="Email address"

              onChange={(e) => setEmail(e.target.value)}

              required

            />

          </Form.Group>

          <Form.Group className="mb-3" controlId="formBasicPassword">

            <Form.Control

              type="password"

              placeholder="Password"

              onChange={(e) => setPassword(e.target.value)}

              required

            />

          </Form.Group>

          <Form.Group className="mb-3" controlId="formBasicConfirmPassword">

            <Form.Control

              type="password"

              placeholder="Confirm Password"

              onChange={(e) => setConfirmPassword(e.target.value)}

              required

            />

          </Form.Group>

          <div className="d-grid gap-2">

            <Button variant="primary" type="submit">

              Sign up

            </Button>

          </div>

        </Form>

      </div>

      <div className="p-4 box mt-3 text-center">

        Already have an account? <Link to="login">Log In</Link>

      </div>

    </>

  );

};

export default Signup;

/\* import React, { useState } from "react";

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

import { Form, Alert } from "react-bootstrap";

import { Button } from "react-bootstrap";

import { useUserAuth } from "../context/UserAuthContext";

const Signup = () => {

  const [email, setEmail] = useState("");

  const [error, setError] = useState("");

  const [password, setPassword] = useState("");

  const { signUp } = useUserAuth();

  let navigate = useNavigate();

  const handleSubmit = async (e) => {

    e.preventDefault();

    setError("");

    try {

      await signUp(email, password);

      navigate("/login");

    } catch (err) {

      setError(err.message);

    }

  };

  return (

    <>

      <div className="p-4 box">

        <h2 className="mb-3">Firebase/ React Auth Signup</h2>

        {error && <Alert variant="danger">{error}</Alert>}

        <Form onSubmit={handleSubmit}>

          <Form.Group className="mb-3" controlId="formBasicEmail">

            <Form.Control

              type="email"

              placeholder="Email address"

              onChange={(e) => setEmail(e.target.value)}

            />

          </Form.Group>

          <Form.Group className="mb-3" controlId="formBasicPassword">

            <Form.Control

              type="password"

              placeholder="Password"

              onChange={(e) => setPassword(e.target.value)}

            />

          </Form.Group>

          <div className="d-grid gap-2">

            <Button variant="primary" type="Submit">

              Sign up

            </Button>

          </div>

        </Form>

      </div>

      <div className="p-4 box mt-3 text-center">

        Already have an account? <Link to="login">Log In</Link>

      </div>

    </>

  );

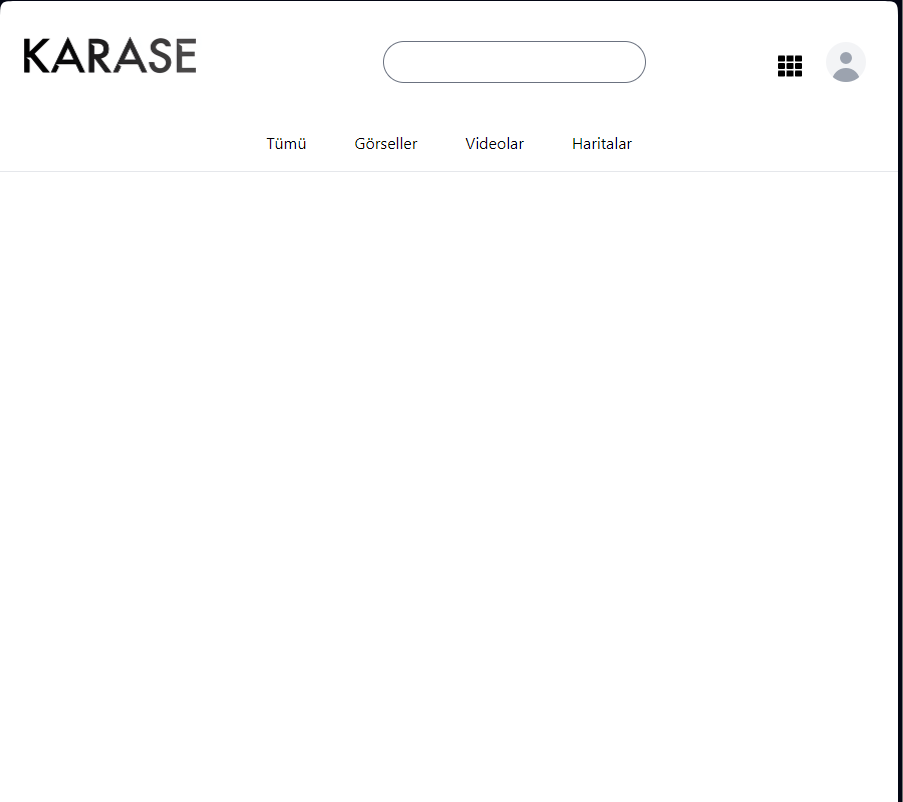
};

export default Signup;

 \*/

Arama Yapıldığında Listelenen Yer:

Veritabanı bağlı olmadığından dolayı şu anda listelenmemektedir veritabanı bağlanıldığı zaman listelenmektedir.



Arama Yapılan Listenin Kodları:

import React, { useState, useEffect } from "react";

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

import { Avatar } from "flowbite-react";

import { FaTh } from "react-icons/fa";

import axios from "axios";

const SearchList = () => {

  const [searchList, setSearchList] = useState([]);

  useEffect(() => {

    axios.get("http://localhost:8080/api/search").then((response) => {

      setSearchList(response.data);

    });

  }, []);

  const [searchText, setSearchText] = useState("");

  const navigate = useNavigate();

  const handleKeyPress = (event) => {

    if (event.key === "Enter") {

      // Enter tuşuna basıldığında "/list" sayfasına yönlendirme yapılır

      navigate("/list");

    }

  };

  return (

    <div>

      {/\*        Navbar \*/}

      <div className="sticky top-0 z-50 bg-white ">

        <div className="flex justify-between items-center mt-4 mb-5 px-5 py-3">

          <Link to="/">

            <div className="flex-none">

              <img

                className="w-1/2"

                src="../src/assets/logoSearch.png"

                alt=""

              />

            </div>

          </Link>

          <div className="flex-grow">

            <input

              type="text"

              name="name"

              className="w-2/3 mt-3 border border-gray-500 rounded-3xl"

              value={searchText}

              onChange={(e) => setSearchText(e.target.value)}

              onKeyDown={handleKeyPress}

            />

          </div>

          <div className="flex">

            <Link to="/ucnokta">

              <div>

                <FaTh className="text-2xl md:mt-6 md:mr-6 mt-2 mr-2 hover:text-gray-400" />

              </div>

            </Link>

            <Link to="/login">

              <div>

                <Avatar

                  className="md:mt-3 md:mr-3 mt-2 mr-2 hover:scale-125 transition-transform"

                  rounded

                />

              </div>

            </Link>

          </div>

        </div>

      </div>

      {/\* video görsek cart curt yazısı  \*/}

      <div className="flex justify-center  items-center p-4">

        <div>

          <Link to="/list">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Tümü

            </a>

          </Link>

          <Link to="/pictures">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Görseller

            </a>

          </Link>

          <Link to="/videos">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Videolar

            </a>

          </Link>

          <Link to="/maps">

            <a href="#" className="hover:underline hover:text-gray-400">

              Haritalar

            </a>

          </Link>

        </div>

      </div>

      <hr className="" />

      {/\*  Arama Sonuçları \*/}

      <div>

        {searchList.map((se) => (

          <div key={se.id} className="md:w-1/2 mx-3 ml-60 mt-7 mb-10">

            <Link to="/newsin">

              <div className="flex">

                <div className="rounded-full overflow-hidden w-9 h-9 hover:scale-110 transition-transform">

                  <img

                    src={se.image}

                    alt={se.image}

                    className="w-full h-full object-cover"

                  />

                </div>

                <div className="ml-3 text-sm hover:underline">

                  <h4>{se.shortURL}</h4>

                  <h5>{se.longURL}</h5>

                </div>

              </div>

            </Link>

            <div className="mt-4">

              <Link to="/newsin">

                <h3 className="text-blue-500 font-bold text-xl hover:underline">

                  {se.title}

                </h3>

              </Link>

              <p className="text-sm mt-2">{se.description}</p>

              <div className="mt-2 text-blue-400">

                <Link to="/newsin">

                  <a className="mr-3" href="">

                    {se.metaTag}

                  </a>

                </Link>

              </div>

            </div>

          </div>

        ))}

      </div>

      {/\* pagination \*/}

      <div></div>

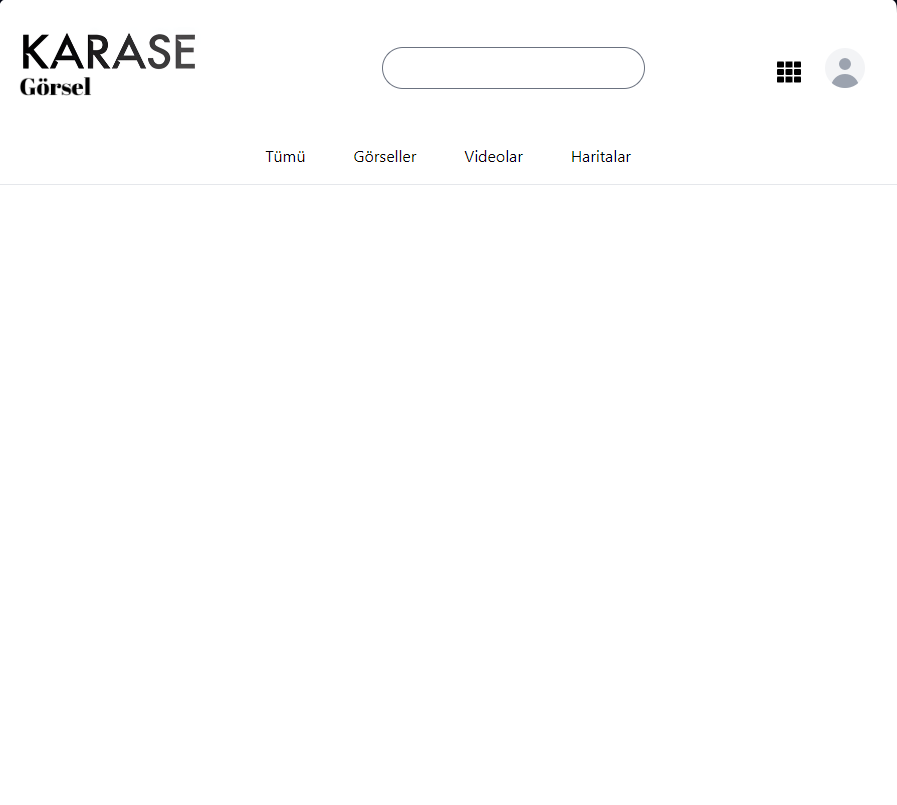
    </div>

  );

};

export default SearchList;

Arama Sonucu İçin Görseller:



Arama Sonucu İçin Görseller:

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

import { Avatar } from "flowbite-react";

import { FaTh } from "react-icons/fa";

import React, {useState,useEffect} from "react";

import axios from "axios";

const MetinKisalt = ({ metin, maksKarakter }) => {

  if (metin.length <= maksKarakter) {

    return <p>{metin}</p>;

  }

  const kisaltilmisMetin = metin.slice(0, maksKarakter) + "...";

  return <p>{kisaltilmisMetin}</p>;

};

const PicturesSearch = () => {

  const [searchText, setSearchText] = useState("");

  const navigate = useNavigate();

  const handleKeyPress = (event) => {

    if (event.key === "Enter") {

      // Enter tuşuna basıldığında "/list" sayfasına yönlendirme yapılır

      navigate("/list");

    }

  };

    //modelopen ise bir görsele tıkladığımız zaman ekranında sağında açılan page işlevi için kullanılan bir fonksiyon olacak.

  const [modalOpen, setModalOpen] = useState(false);

  const openModal = () => {

    setModalOpen(true);

  };

  const closeModal = () => {

    setModalOpen(false);

  };

  //db'den çektiğimz verileri api ile almak için kullanırız.

  const [searchPic, setSearchPic] = useState([])

    useEffect(() => {

      axios.get("http://localhost:8080/api/pic").then((response) => {

        setSearchPic(response.data);

      });

    }, []);

  return (

    <div>

      {/\*        Navbar \*/}

      <div className="sticky top-0 z-50 bg-white ">

        <div className="flex justify-between items-center mt-4 mb-5 px-5 py-3">

          <Link to="/">

            <div className="flex-none g">

              <img

                className="w-1/2"

                src="../src/assets/logoSearch.png"

                alt=""

              />

              <img className="w-1/5" src="../src/assets/görsel.png" alt="" />

            </div>

            <span></span>

          </Link>

          <div className="flex-grow">

            <input

              type="text"

              name="name"

              className="w-2/3 mt-3 border border-gray-500 rounded-3xl"

              value={searchText}

              onChange={(e) => setSearchText(e.target.value)}

              onKeyDown={handleKeyPress}

            />

          </div>

          <div className="flex">

            <Link to="/ucnokta">

              <div>

                <FaTh className="text-2xl md:mt-6 md:mr-6 mt-2 mr-2 hover:text-gray-400" />

              </div>

            </Link>

            <Link to="/login">

              <div>

                <Avatar

                  className="md:mt-3 md:mr-3 mt-2 mr-2 hover:scale-125 transition-transform"

                  rounded

                />

              </div>

            </Link>

          </div>

        </div>

      </div>

      {/\* video görsek cart curt yazısı  \*/}

      <div className="flex justify-center  items-center p-4">

        <div>

          <Link to="/list">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Tümü

            </a>

          </Link>

          <Link to="/pictures">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Görseller

            </a>

          </Link>

          <Link to="/videos">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Videolar

            </a>

          </Link>

          <Link to="/maps">

            <a href="#" className="hover:underline hover:text-gray-400">

              Haritalar

            </a>

          </Link>

        </div>

      </div>

      <hr className="" />

      {/\* Görsllerin olduğu kısım \*/}

      <div>

        {searchPic.map((sp) => (

          <div className="flex flex-wrap justify-center">

            <div className="w-88 h-48 mx-2 my-10 mb-12">

              <img

                src={sp.picture}

                alt=""

                className="w-full h-full object-cover rounded-md"

                onClick={openModal}

              />

              <div className="flex mt-1">

                <div className="rounded-full overflow-hidden w-5 h-5 hover:scale-110 transition-transform">

                  <img

                    src={sp.logo}

                    alt=""

                    className="w-full h-full object-cover"

                    onClick={openModal}

                  />

                </div>

                <h3 className="ml-1 font-thin" onClick={openModal}>

                  {sp.picTitle}

                </h3>

              </div>

              <p onClick={openModal}>

                <MetinKisalt

                  metin={sp.picDescription}

                  maksKarakter={48}

                />

              </p>

            </div>

          </div>

        ))}

      </div>

      {modalOpen && (

        //Buraya aşağıya göre baka baka bir tane div oluştur.

        <div className="fixed top-0 right-0 h-screen w-1/2 flex items-center justify-end bg-white ">

          <div className="bg-whit e p-8 rounded-lg">

            <h2 className="text-2xl font-bold mb-4">Detaylı İçerik</h2>

            <p>

              Bu alanda blog gönderisinin detaylı içeriği yer alabilir. Detaylı

              içerik burada...

            </p>

            <button

              className="mt-4 bg-blue-500 hover:bg-blue-700 text-white py-2 px-4 rounded"

              onClick={closeModal}

            >

              Kapat

            </button>

          </div>

        </div>

      )}

    </div>

  );

};

export default PicturesSearch;

<p>

  <MetinKisalt

    metin="Lorem ipsum dolor sit amet consectetur adipisicing elit. Expedita commodi eum natus quia provident enim. Modi ipsa ducimus atque et velit labore nemo aspernatur officia at! Modi quaerat unde distinctio."

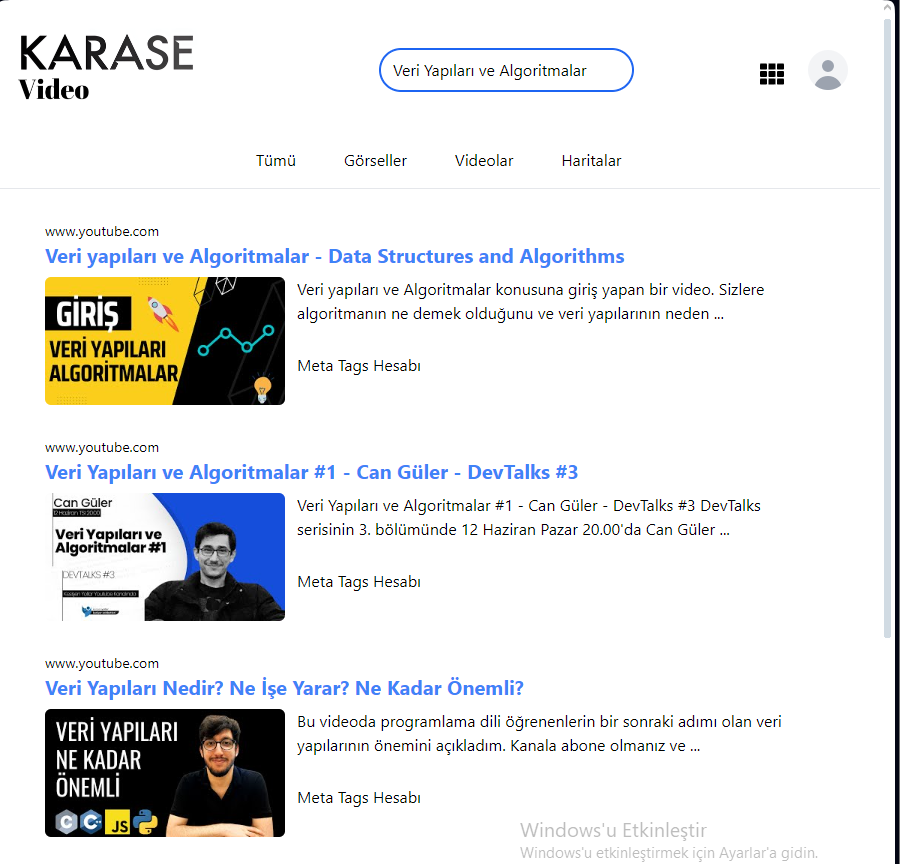
    maksKarakter={60}

  />

</p>;

Arama Sonucu İçin Videolar Kısımı:

Burada Youtube Api kullandım.



Videolar İçin Olan Kısımın Kodları:

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

import { Avatar } from "flowbite-react";

import { FaTh } from "react-icons/fa";

import React, { useState, useEffect } from "react";

const MetinKisalt = ({ metin, maksKarakter }) => {

  if (metin.length <= maksKarakter) {

    return <p>{metin}</p>;

  }

  const kisaltilmisMetin = metin.slice(0, maksKarakter) + "...";

  return <p>{kisaltilmisMetin}</p>;

};

const VideosSearch = () => {

  const [searchText, setSearchText] = useState("");

  const navigate = useNavigate();

  const handleKeyPress = (event) => {

    if (event.key === "Enter") {

      // Enter tuşuna basıldığında "/list" sayfasına yönlendirme yapılır

      navigate("/list");

    }

  };

  //Youtube Api için kral uzun kod bloğu he

  const [searchTerm, setSearchTerm] = useState("");

  const [searchResults, setSearchResults] = useState([]);

  const [buttonClicked, setButtonClicked] = useState(false);

  const apiKey = "AIzaSyC6LFAXBW\_B-Dj2xn\_JJbnqSoNVg2oybe0";

  useEffect(() => {

    const fetchData = async () => {

      try {

        if (!searchTerm || !buttonClicked) {

          setSearchResults([]);

          return;

        }

        const response = await fetch(

          `https://www.googleapis.com/youtube/v3/search?q=${searchTerm}&key=${apiKey}&part=snippet&type=video`

        );

        const data = await response.json();

        if (data.items) {

          setSearchResults(data.items);

        } else {

          setSearchResults([]);

        }

      } catch (error) {

        console.error("Error fetching data:", error);

      }

    };

    fetchData();

  }, [searchTerm, apiKey, buttonClicked]);

  const handleSearchButtonClick = () => {

    setButtonClicked(true);

  };

  const handleKeyPressVideos = (e) => {

    if (e.key === "Enter") {

      setButtonClicked(true);

    }

  };

  const openVideoInNewTab = (videoId) => {

    window.open(`https://www.youtube.com/watch?v=${videoId}`, "\_blank");

  };

  return (

    <div>

      {/\*        Navbar \*/}

      <div className="sticky top-0 z-50 bg-white ">

        <div className="flex justify-between items-center mt-4 mb-5 px-5 py-3">

          <Link to="/">

            <div className="flex-none ">

              <img

                className="w-1/2"

                src="../src/assets/logoSearch.png"

                alt=""

              />

              <img className="w-1/5" src="../src/assets/video.png" alt="" />

            </div>

            <span></span>

          </Link>

          <div className="flex-grow">

            {/\*             <input

              type="text"

              name="name"

              className="w-2/3 mt-3 border border-gray-500 rounded-3xl"

              value={searchText}

              onChange={(e) => setSearchText(e.target.value)}

              onKeyDown={handleKeyPressVideos}

            /> \*/}

            <input

              type="text"

              value={searchTerm}

              onChange={(e) => setSearchTerm(e.target.value)}

              onKeyDown={handleKeyPressVideos}

              placeholder=""

              className="w-2/3 mt-3 border border-gray-500 rounded-3xl"

            />

          </div>

          <div className="flex">

            <Link to="/ucnokta">

              <div>

                <FaTh className="text-2xl md:mt-6 md:mr-6 mt-2 mr-2 hover:text-gray-400" />

              </div>

            </Link>

            <Link to="/login">

              <div>

                <Avatar

                  className="md:mt-3 md:mr-3 mt-2 mr-2 hover:scale-125 transition-transform"

                  rounded

                />

              </div>

            </Link>

          </div>

        </div>

      </div>

      {/\* video görsek cart curt yazısı  \*/}

      <div className="flex justify-center  items-center p-4">

        <div>

          <Link to="/list">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Tümü

            </a>

          </Link>

          <Link to="/pictures">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Görseller

            </a>

          </Link>

          <Link to="/videos">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Videolar

            </a>

          </Link>

          <Link to="/maps">

            <a href="#" className="hover:underline hover:text-gray-400">

              Haritalar

            </a>

          </Link>

        </div>

      </div>

      <hr className="" />

      {/\* videoların gösterileceği yer \*/}

      <div>

        {searchResults.slice(0, 5).map((result) => (

          <div key={result.id.videoId} className="flex flex-wrap justify-start">

            <div className="mx-12 lg:mx-48 mt-8 ">

              <div className="text-sm">

                <p>www.youtube.com</p>

              </div>

              <div>

                <h4 className="text-blue-500 font-bold text-xl hover:underline">

                  {result.snippet.title}

                </h4>

              </div>

              <div className="flex mt-2">

                <div className="flex-none w-52 h-32 md:w-60 md:h-32">

                  <img

                    className="w-full h-full object-cover rounded-md"

                    src={result.snippet.thumbnails.medium.url}

                    alt=""

                  />

                </div>

                <div className="mx-3">

                  <div>

                    <p>

                      <MetinKisalt

                        metin={result.snippet.description}

                        maksKarakter={130}

                      />

                    </p>

                  </div>

                  <div className="my-7">Meta Tags Hesabı</div>

                </div>

              </div>

            </div>

          </div>

        ))}

      </div>

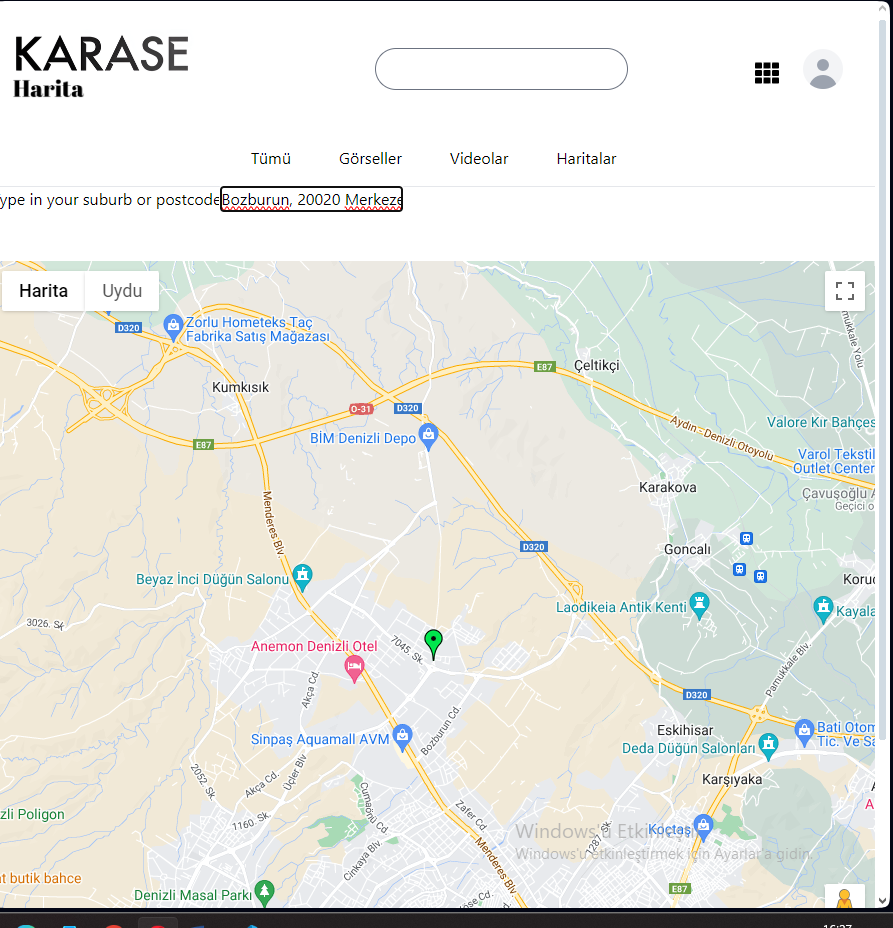
    </div>

  );

};

export default VideosSearch;

Arama Motorunda Haritalar İçin Olan Kısım:



Haritalar İçin Olan Kısımın Kodları

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

import { Avatar } from "flowbite-react";

import { FaTh } from "react-icons/fa";

import React, { useState } from "react";

import GooglePlacesApi from "../components/Map/GooglePlacesApi";

import Map from "../components/Map/Map";

const MapSearch = () => {

  const [searchText, setSearchText] = useState("");

  const navigate = useNavigate();

  const handleKeyPress = (event) => {

    if (event.key === "Enter") {

      // Enter tuşuna basıldığında "/list" sayfasına yönlendirme yapılır

      navigate("/list");

    }

  };

  const [selectedLocation, setSelectedLocation] = useState({

    lat: 39.9334,

    lng: 32.8597,

  });

  return (

    <div>

      {/\*        Navbar \*/}

      <div className="sticky top-0 z-50 bg-white ">

        <div className="flex justify-between items-center mt-4 mb-5 px-5 py-3">

          <Link to="/">

            <div className="flex-none g">

              <img

                className="w-1/2"

                src="../src/assets/logoSearch.png"

                alt=""

              />

              <img className="w-1/5" src="../src/assets/harita.png" alt="" />

            </div>

            <span></span>

          </Link>

          <div className="flex-grow">

            <input

              type="text"

              name="name"

              className="w-2/3 mt-3 border border-gray-500 rounded-3xl"

              value={searchText}

              onChange={(e) => setSearchText(e.target.value)}

              onKeyDown={handleKeyPress}

            />

          </div>

          <div className="flex">

            <Link to="/ucnokta">

              <div>

                <FaTh className="text-2xl md:mt-6 md:mr-6 mt-2 mr-2 hover:text-gray-400" />

              </div>

            </Link>

            <Link to="/login">

              <div>

                <Avatar

                  className="md:mt-3 md:mr-3 mt-2 mr-2 hover:scale-125 transition-transform"

                  rounded

                />

              </div>

            </Link>

          </div>

        </div>

      </div>

      {/\* video görsek cart curt yazısı  \*/}

      <div className="flex justify-center  items-center p-4">

        <div>

          <Link to="/list">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Tümü

            </a>

          </Link>

          <Link to="/pictures">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Görseller

            </a>

          </Link>

          <Link to="/videos">

            <a href="#" className="mr-12 hover:underline hover:text-gray-400">

              Videolar

            </a>

          </Link>

          <Link to="/maps">

            <a href="#" className="hover:underline hover:text-gray-400">

              Haritalar

            </a>

          </Link>

        </div>

      </div>

      <hr className="" />

      {/\* map gözükecek yer \*/}

      <div>

        <div style={{ height: "100vh", width: "100%" }}>

          <GooglePlacesApi setSelectedLocation={setSelectedLocation} />

          <Map selectedLocation={selectedLocation} />

        </div>

      </div>

    </div>

  );

};

export default MapSearch;