Front End Engineering-II

Project Report

Semester-IV (Batch-2022)

DOOB:The Spirit of Digital Agency

Supervised By:

Dr. Raveesh Samkaria

Submitted By:

Rydhampreet Singh Gindra

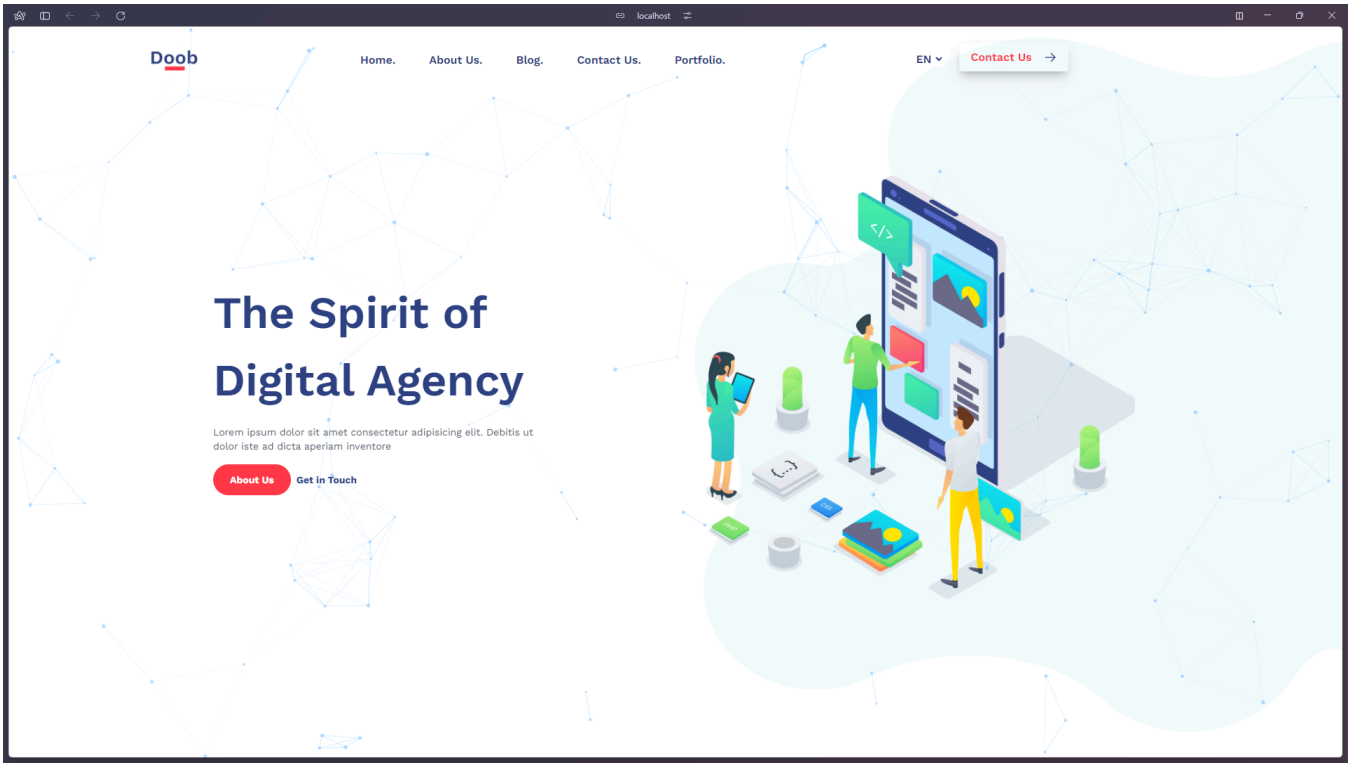
2210990751

G-12

Department of Computer Science and Engineering Chitkara University Institute of Engineering & Technology, Chitkara University, Punjab

Abstract

This user-friendly Digital Agency website “DOOB: The spirit of Digital Agency” empowers you to stay mesmerized in the fluidity of animations and stay impressed with the stunning UI and with the use of strong responsive development principles and the power of React JS, we have received heart-warming results.



INDEX

1. Introduction

The DOOB web application offers a convenient yet efficient platform for knowing about the Digital Agency and find a way to contact it. In today's digital age, where convenience and accessibility are paramount, fluid UI is not a need but a necessity. This introduction provides an overview of the background, objectives, and significance of the “DOOB”.

1.1 Background:

As technology continues to permeate various aspects of our lives, the need for staying at speed is also essential. The “DOOB” web app addresses a modern need i.e. need for a digital agency for various entrepreneurial activities. With the proliferation of web-based applications, users expect mesmerizing UI and top-notch interactivity from a web app.

1.2 Objectives:

The primary objective of the “DOOB” is to provide a user-friendly, fluid interface for individuals to interact with Digital Agencies. This application aims to simplify the process, eliminating the need for users to browse through the internet or rely on external means. Additionally, the “DOOB” web app strives to become better everyday..

1.3 Significance:

The significance of the “DOOB” web app lies in its ability to streamline a complex task that Entrepreneurs encounter regularly. Whether for personal use, professional purposes, or administrative requirements, Digital Agency based tasks are needed. By offering a convenient and reliable solution, the “DOOB” web app enhances efficiency and productivity, saving users time and effort.

2. Problem Statement

In the contemporary digital landscape, users demand websites that offer not only accessibility and convenience but also engaging and fluid user experiences. Despite the proliferation of digital agencies, many still struggle to balance aesthetic appeal with robust functionality and responsiveness. The challenge lies in developing a website that captivates users with seamless animations and a stunning user interface (UI) while ensuring strong responsive development and optimal performance. The DOOB web application addresses this gap by leveraging React JS to create a visually impressive and highly functional platform for users to learn about and contact the Digital Agency. The core problem is to effectively integrate advanced UI/UX principles with responsive web design to create an engaging, user-friendly experience that meets modern expectations for digital agency websites.

3. Software Requirements

a) Integrated Development Environment (IDE):

- Visual Studio Code (VS Code) for code editing and project management.

b) Frontend Technologies:

- ReactJS: ReactJS is a JavaScript library for building user interfaces with reusable components.

c) User Interface (UI) Framework:

- TailwindCSS: Styling language for enhancing the presentation and layout.

d) Version Control:

- Git: Git is a distributed version control system for tracking changes in source code during software development.

e) Deployment:

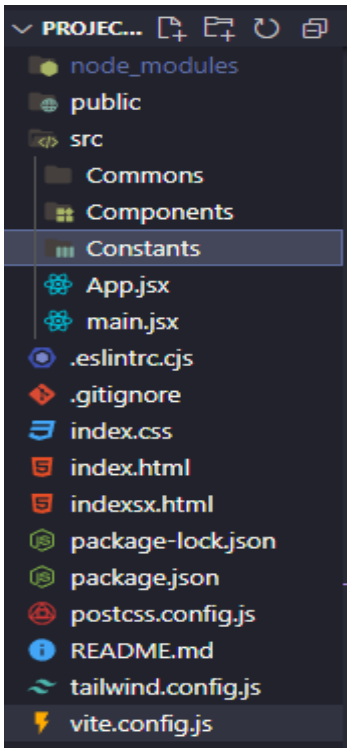
- Vercel: Vercel is a platform for frontend developers to deploy and host web applications.

4. Proposed Design

- **User Interface Design:** Utilize Tailwind and TsParticles for a responsive, visually appealing lay- out for intuitive organization.
- **Frontend Development:** Develop using ReactJS. Utilize React State management, Refs, Component based structuring and Context API
- **User Experience Optimization:** Focus on real-time feedback, interactive elements, and cross-browser compatibility. Ensure responsiveness for varied devices.
- **Documentation and Deployment:** Provide detailed documentation. Deploy on web server with domain. Maintain and update documentation regularly.
- **Availability for the Community:** availablity on Github in a public repository so that community can check it out.

4.1 File Structure

Ensuring proper file and folder structure to maintain consistent file paths and clean structure.



4.2 Components:

- **App Component**

```jsx
import { createContext, useRef } from "react";
import {
  Contact,
  Footer,
  Hero,
  Navbar,
  News,
  ParticlesComponent,
  Projects,
  Reviews,
  Services,
} from "./Components";
export const RefCon = createContext(null);

function App() {
  const HeroRef = useRef(null);
  const ServicesRef = useRef(null);
  const ReviewsRef = useRef(null);
  const ProjectsRef = useRef(null);
  const NewsRef = useRef(null);
  const ContactRef = useRef(null);
  const FooterRef = useRef(null);
  return (
    <RefCon.Provider
      value={{
        HeroRef,
        ServicesRef,
        ReviewsRef,
        ProjectsRef,
        NewsRef,
        ContactRef,
        FooterRef,
      }}
    >
      <div className="flex flex-col items-center justify-center ">
        <ParticlesComponent id="main" />
        <Navbar />
        <Hero />
        <Services />
        <Reviews />
        <Projects />
        <News />
        <Contact />
        <Footer />
      </div>
    </RefCon.Provider>
  );
}

export default App;
```

- **Navbar Component**

```jsx
import { Lang, NavItems } from "../Constants/Constants";
import Logo from "/Nav/Logo.svg";
import NavbarRes from "./NavbarRes";
import Ham from "/Icons/ham.svg"
import Right from "/Icons/Right.svg"
import { useState } from "react";
import {motion} from "framer-motion"
import { RefCon } from "../App";
import { useContext } from "react";

const Navbar = () => {
  const {HomeRef, ServicesRef, NewsRef, ContactRef, FooterRef} = useContext(RefCon)

  const arr = [HomeRef, ServicesRef, NewsRef, ContactRef, FooterRef];

   const [show, isShow] = useState(false)

   const handleClick=(index)=>{

    arr[index].current.scrollIntoView({ behavior: 'smooth' });

   }
  return (
    <motion.div
    initial={{y:-50, opacity:0.4}}
    whileInView={{y:0, opacity:1}}
    transition={{duration:0.3}}
    exit={{y:-50, opacity:0.4}}
    className="max-w-[1500px] h-full w-full max-lg:px-10 py-6 flex items-center justify-between">
      <div id="logo" className="w-1/2">
        <a href="#">
          <img src={Logo} alt="" />
        </a>
      </div>
      <ul id="navItems" className="flex items-center w-full gap-12 max-lg:hidden">
        {NavItems.map((item, index) => {
          return (
            <li
              key={index}
              className="text-[#314584] hover:underline duration-100 cursor-pointer font-semibold hover:font-bold transition-all"
            >
              <p onClick={()=>handleClick(index)} href={item.href}>{item.name}</p>
            </li>
          );
        })}
      </ul>
      <div
        id="LangCont"
        className="flex items-center justify-center w-full h-full gap-5 max-md:hidden max-lg:justify-start"
      >
        <select
          name="lang"
          className="mb-1 hover:cursor-pointer rounded text-[#314584] font-semibold outline-none focus:outline-none"
          id="lang"
        >
          {Lang.map((lang, index) => {
            return (
              <option key={index} className="text-[#314584]" value={lang.name}>
                {lang.value.toUpperCase()}
              </option>
            );
          })}
        </select>
        <button onClick={()=>handleClick(3)} className="px-4 py-2 flex items-center duration-150 gap-4 rounded shadow-lg hover:scale-[1.01] text-[#FF3946] mb-2 font-semibold
over:shadow-black/40 shadow-black/20 ">
          Contact Us <img src={Right} className="w-5 h-5" alt="" />
        </button>
      </div>
          <div className="flex items-center justify-center"><img src={Ham} className="w-16 h-10 cursor-pointer lg:hidden" onClick={()=>isShow(true)} alt="" /></div>

          {show &&  <NavbarRes isShow={isShow} show={show} Logo={Logo} NavItems={NavItems} Lang={Lang}/>}

    </motion.div>
  );
};

export default Navbar;

```

- **Hero Component**

```jsx
import WaveOne from "/Wave1.svg";
import ManPhone from "/Home/ManPhone.svg";
import Button from "../Commons/Button";
import HeroTypeWritter from "../Commons/HeroTypeWritter";
import {motion} from "framer-motion"
import { useContext } from "react";
import { RefCon } from "../App";

const Hero = () => {
  const {Heroref, ServicesRef} = useContext(RefCon)
  const handleClick=(Ref)=>{

    Ref.current.scrollIntoView({ behavior: 'smooth' });

   }
  return (
    <>
      <div className="relative flex items-end justify-end w-full">
        <img
          src={WaveOne}
          className="absolute top-[-5rem] max-lg:top-[1rem] -z-50"
          alt=""
        />
      </div>
      <div ref = {Heroref} className="relative flex items-center justify-center w-full">
        <div className="flex flex-col lg:flex-row max-w-[1440px] items-center justify-center w-full h-full  px-5 pt-5 pb-20">
          <motion.div
          initial={{x:-50, opacity:0.5}}
          whileInView={{x:0, opacity:1}}
          transition={{duration:0.3}}
          exit={{x:-50, opacity:0.5}}
          className="flex flex-col items-center justify-center w-full h-full px-10 py-24 text-center lg:text-left lg:items-start">
            <h1 className="text-[#314584] font-semibold lg:text-[4rem] text-[3.5rem] pb-2">
              The Spirit of <br /> Digital
              <HeroTypeWritter
                words={[" Agency", " Firm"]}
                speed={200}
              />
            </h1>
            <p className="pt-2 pb-4 text-sm text-gray-600">
              Lorem ipsum dolor sit amet consectetur adipisicing elit. Debitis
              ut <br /> dolor iste ad dicta aperiam inventore
            </p>
            <div className="flex gap-2">
              <Button><p onClick={()=>handleClick(ServicesRef)}>About Us</p></Button>
              <button onClick={()=>handleClick(ServicesRef)} className="text-[#314584] text-sm font-bold">
                Get in Touch
              </button>
            </div>
          </motion.div>
          <motion.div
          initial={{x:50, opacity:0.5}}
          whileInView={{x:0, opacity:1}}
          transition={{duration:0.3}}
          exit={{x:50, opacity:0.5}}
          className="flex flex-col items-end justify-center w-full h-full px-10 pt-10 lg:py-24">
            <img src={ManPhone} alt="" />
          </motion.div>
        </div>
      </div>
    </>
  );
};

export default Hero;

```

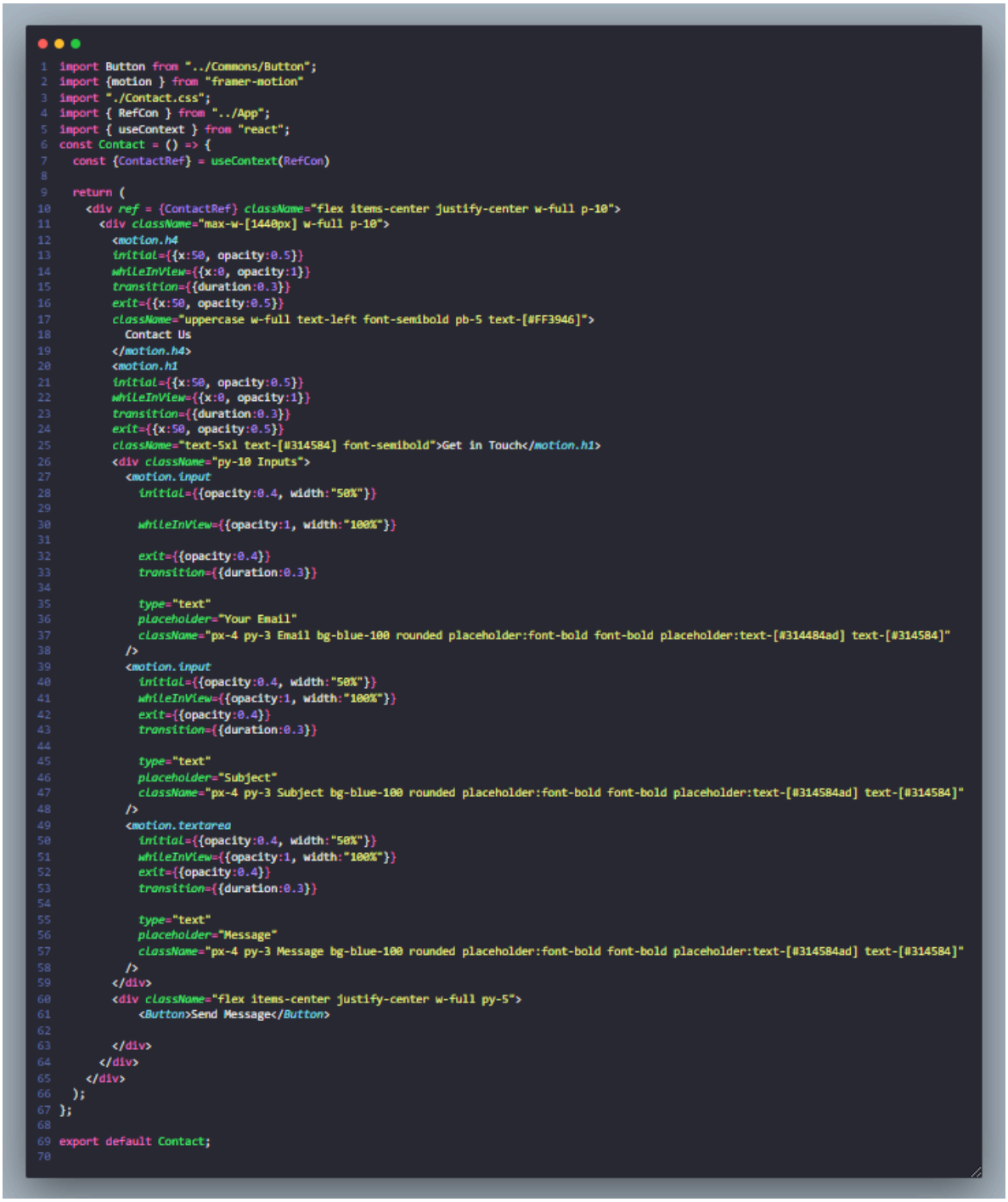
- **Services Component**

```jsx
import Button from "../Commons/Button";
import Card from "./Card";
import Wave2 from "/Wave2.svg";
import { CardContent } from "../Constants/Constants.js";
import { motion } from "framer-motion";
import { RefCon } from "../App.jsx";
import { useContext } from "react";

const Services = () => {

  const {ServicesRef, ContactRef} = useContext(RefCon)
  const handleClick=(ref)=>{

    ref.current.scrollIntoView({ behavior: 'smooth' });

  }
  return (
    <>
      <div  className="relative flex items-start w-full justify-normal">
        <img src={Wave2} className="absolute top-32 -z-50" alt="" />
      </div>
      <div ref={ServicesRef} className="flex items-center justify-center w-full pb-10">
        <div className="max-w-[1440px] grid lg:grid-cols-3 grid-cols-1 w-full h-full p-10">
          <motion.div
            initial={{ x: -50, opacity: 0.4 }}
            whileInView={{ x: 0, opacity: 1, once:true }}
            transition={{ duration: 0.3 }}
            exit={{ x: -50, opacity: 0.4 }}
            className="flex flex-col items-start justify-center w-full h-full px-10 py-24"
          >
            <h1 className="mb-9 max-lg:text-center text-[2.98rem] font-semibold leading-normal  text-[#314584]">
              Anything you need we&apos;ve <br /> got you covered
            </h1>
            <div className="flex items-center justify-center w-full lg:justify-start">
              <Button><p onClick={()=>handleClick(ContactRef)}>Get in Touch</p></Button>
            </div>
          </motion.div>
          {CardContent.map((card, index) => {
            return (
              <Card
                image={card.Image}
                heading={card.Heading}
                key={index}
                content={card.Content}
              />
            );
          })}
        </div>
      </div>
    </>
  );
};

export default Services;

```

- **Projects Component**

```jsx
import ImageGrid from "./ImageGrid"
import Wave4 from "/Wave4.svg"
import {motion} from "framer-motion"
const Projects = () => {
  return (
    <>
    <div className="relative flex items-start justify-start w-full">
        <img src={Wave4} className="absolute top-72 -left-72 -z-50" alt="" />
      </div>
      <div className="flex items-center justify-center w-full py-10">
        <div className="max-w-[1440px] w-full flex flex-col items-center justify-center p-5">
            <motion.div
            initial={{x:-50, opacity:0.5}}
            whileInView={{x:0, opacity:1}}
            transition={{duration:0.3}}
            exit={{x:-50, opacity:0.5}}

            className="w-full p-10">
                <h1 className="text-[#314584] font-semibold text-5xl pb-4 text-left leading-normal">See Some of our <br /> Creative Work</h1>
            </motion.div>
            <ImageGrid/>

        </div>
      </div>

    </>
  )
}

export default Projects
```

- **Contact Component**



```jsx
import Button from "../Commons/Button";
import {motion } from "framer-motion"
import "./Contact.css";
import { RefCon } from "../App";
import { useContext } from "react";
const Contact = () => {
  const {ContactRef} = useContext(RefCon)

  return (
    <div ref = {ContactRef} className="flex items-center justify-center w-full p-10">
      <div className="max-w-[1440px] w-full p-10">
        <motion.h4
        initial={{x:50, opacity:0.5}}
        whileInView={{x:0, opacity:1}}
        transition={{duration:0.3}}
        exit={{x:50, opacity:0.5}}
        className="uppercase w-full text-left font-semibold pb-5 text-[#FF3946]">
          Contact Us
        </motion.h4>
        <motion.h1
        initial={{x:50, opacity:0.5}}
        whileInView={{x:0, opacity:1}}
        transition={{duration:0.3}}
        exit={{x:50, opacity:0.5}}
        className="text-5xl text-[#314584] font-semibold">Get in Touch</motion.h1>
        <div className="py-10 Inputs">
          <motion.input
            initial={{opacity:0.4, width:"50%"}}

            whileInView={{opacity:1, width:"100%"}}

            exit={{opacity:0.4}}
            transition={{duration:0.3}}

            type="text"
            placeholder="Your Email"
            className="px-4 py-3 Email bg-blue-100 rounded placeholder:font-bold font-bold placeholder:text-[#314484ad] text-[#314584]"
          />
          <motion.input
            initial={{opacity:0.4, width:"50%"}}
            whileInView={{opacity:1, width:"100%"}}
            exit={{opacity:0.4}}
            transition={{duration:0.3}}

            type="text"
            placeholder="Subject"
            className="px-4 py-3 Subject bg-blue-100 rounded placeholder:font-bold font-bold placeholder:text-[#314584ad] text-[#314584]"
          />
          <motion.textarea
            initial={{opacity:0.4, width:"50%"}}
            whileInView={{opacity:1, width:"100%"}}
            exit={{opacity:0.4}}
            transition={{duration:0.3}}

            type="text"
            placeholder="Message"
            className="px-4 py-3 Message bg-blue-100 rounded placeholder:font-bold font-bold placeholder:text-[#314584ad] text-[#314584]"
          />
        </div>
        <div className="flex items-center justify-center w-full py-5">
            <Button>Send Message</Button>

        </div>
      </div>
    </div>
  );
};

export default Contact;

```

- **Footer**

```jsx
import { NavItems } from "../Constants/Constants";
import Dribble from "/Socials/Dribble.svg"
import FB from "/Socials/FB.svg"
import Twitter from "/Socials/Twitter.svg"
import {motion} from "framer-motion"
import { RefCon } from "../App";
import { useContext } from "react";
//hey
const Footer = () => {
  const {FooterRef} = useContext(RefCon)

  return (
    <div ref = {FooterRef} className="flex items-center justify-center bg-[#314584] w-full relative z-50 p-10">
      <motion.div
      initial={{opacity:0}}
      whileInView={{opacity:1}}
      transition={{duration:0.3}}
      exit={{opacity:0}}
      className="flex flex-col items-center justify-center max-w-[1440px] w-full">
        <p className="font-semibold text-white uppercase">Doob</p>
        <h3 className="uppercase pt-5 font-semibold text-5xl text-[#FF3946]">Creativity Above</h3>
        <ul id="navItems" className="flex items-center justify-center w-full gap-12 py-10 max-lg:hidden">

        {NavItems.map((item, index) => {
          return (
            <li
              key={index}
              className="font-semibold text-white transition-all duration-100 hover:underline hover:font-bold"
            >
              <a href={item.href}>{item.name}</a>
            </li>
          );
        })}
        </ul>
        <p className="text-xs text-white">&copy; 2019 - Doob, All Right Reserved</p>
        <div className="flex gap-10 py-8 Socials">
          <a href=""><img src={FB} className="h-10"alt="" /></a>
          <a href=""><img src={Twitter} className="h-10" alt="" /></a>
          <a href=""><img src={Dribble} className="h-9" alt="" /></a>

        </div>
      </motion.div>

    </div>
  )
}

export default Footer

```

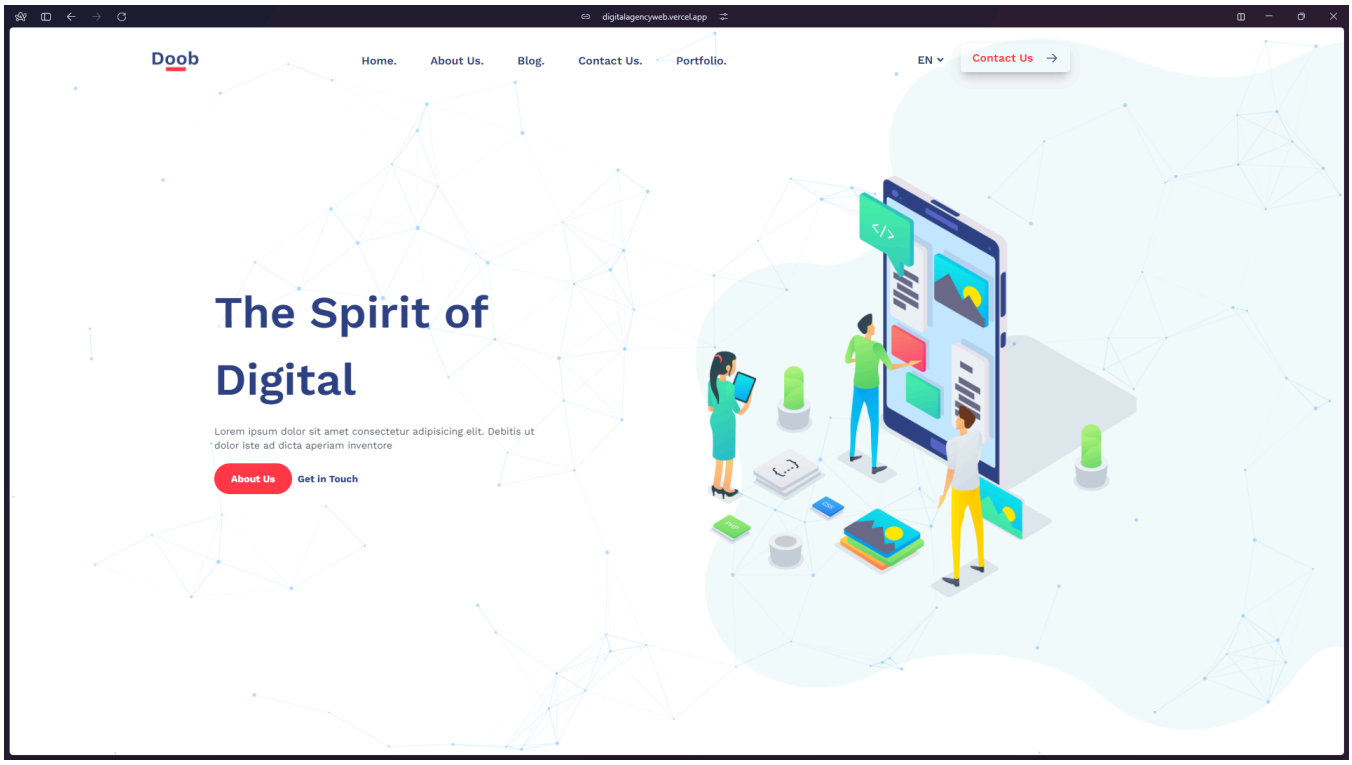
5. Results

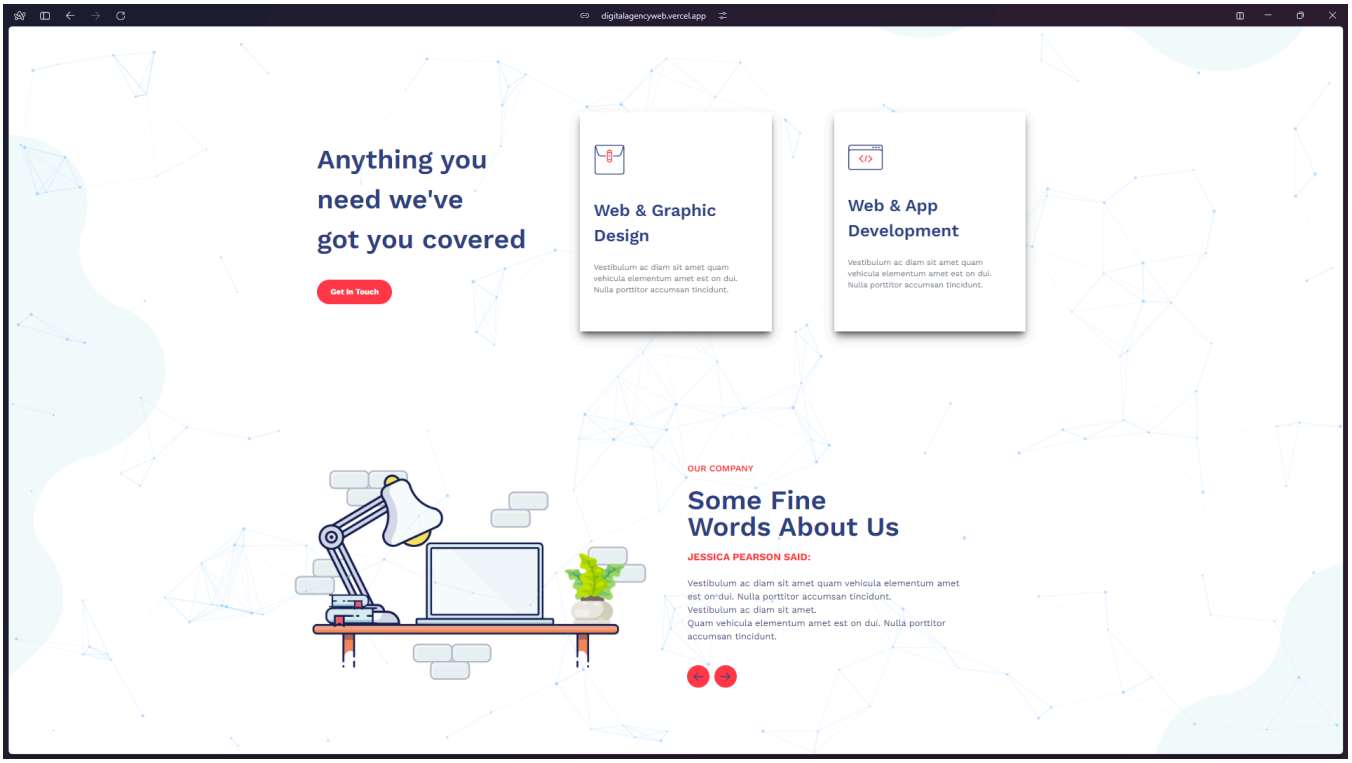
"DOOB" Successfully serves it purpose of easing the life of entrepreneurs.

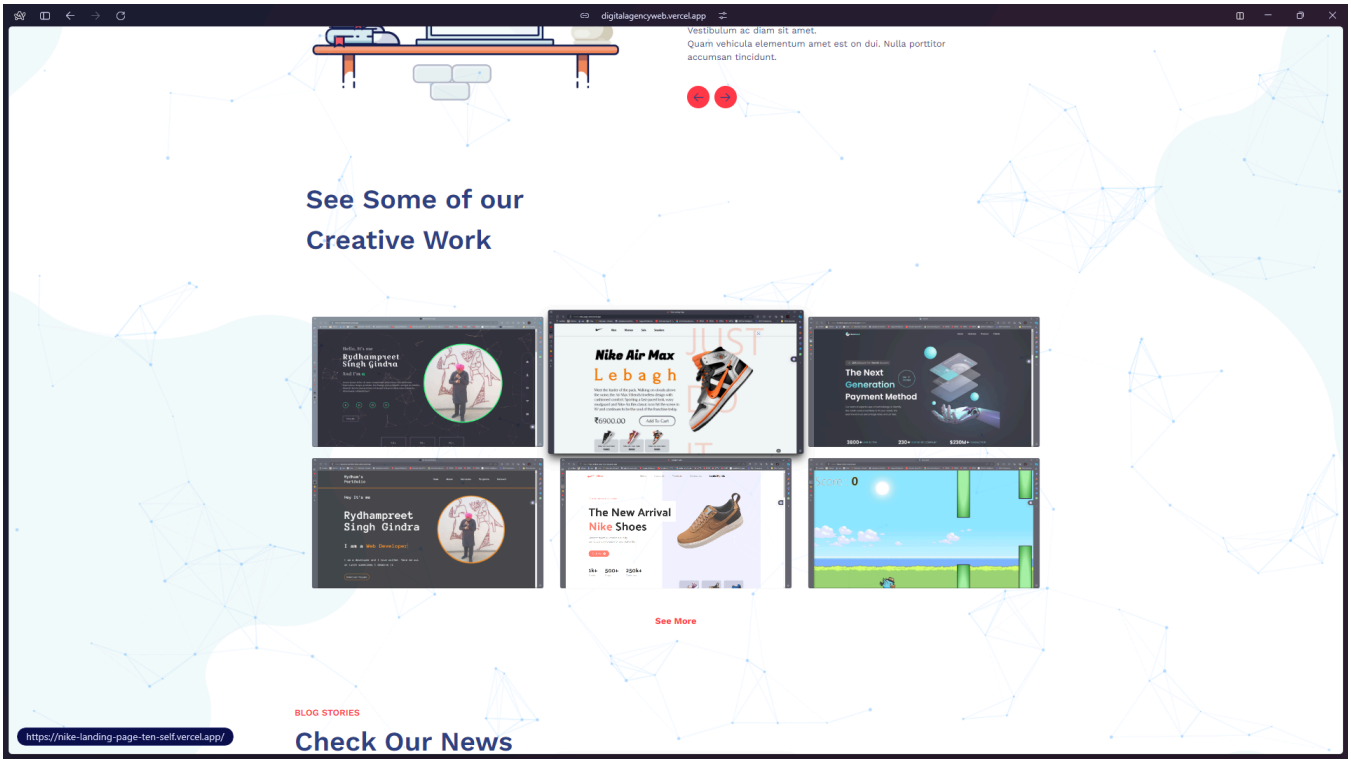
GitHub Repository Link: https://github.com/coolguy-codin-Rydham/ProjectReactCollege.git

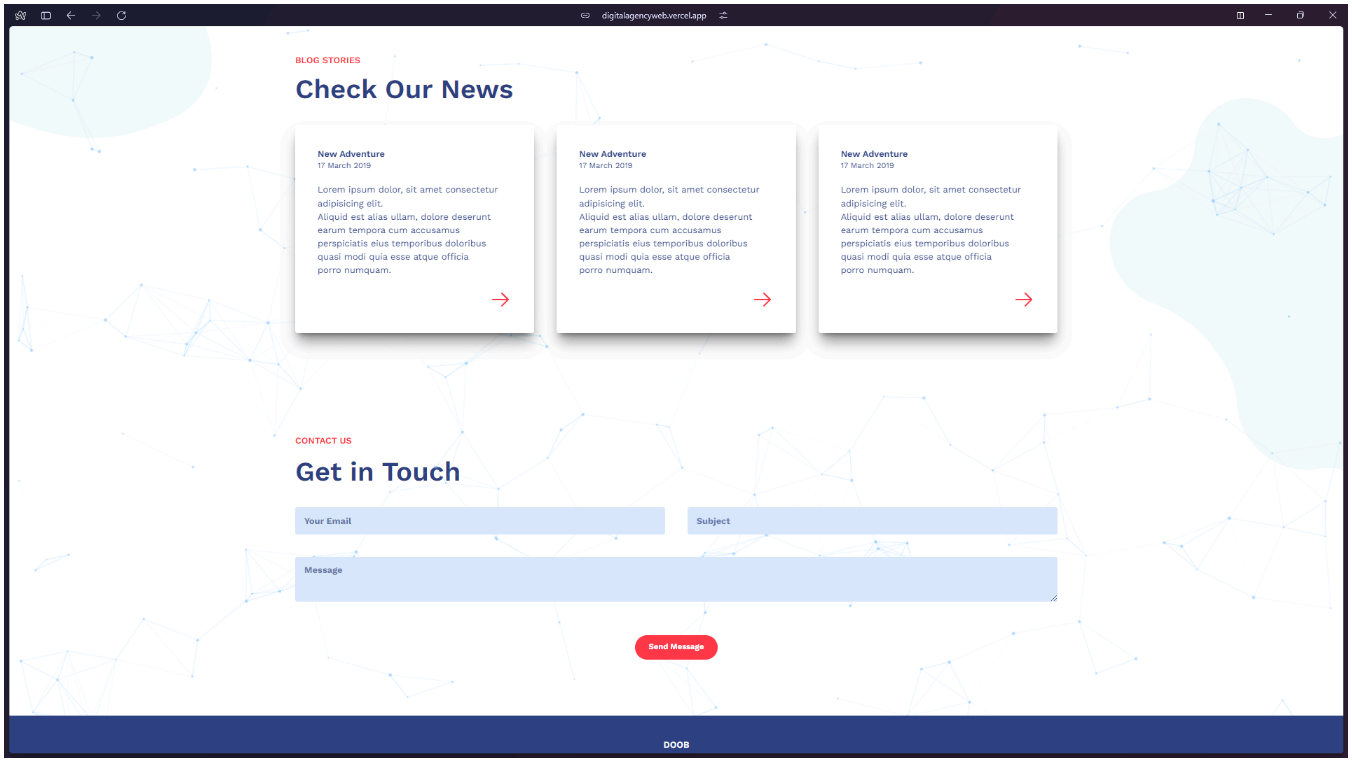
Deployed Via Vercel Link: https://digitalagencyweb.vercel.app/

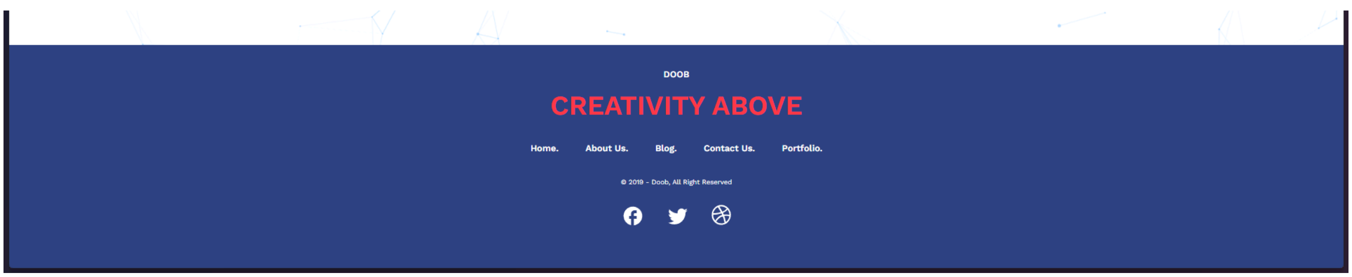
Project Screenshots for all scenarios:




digitalagencyweb.vercel.app
Anything you need we've got you covered
Get In Touch
Web & Graphic Design
Vestibulum ac diam sit amet quam vehicula elementum amet est on dui. Nulla porttitor accumsan tincidunt.
Web & App Development
Vestibulum ac diam sit amet quam vehicula elementum amet est on dui. Nulla porttitor accumsan tincidunt.
OUR COMPANY
Some Fine Words About Us
JESSICA PEARSON SAID:
Vestibulum ac diam sit amet quam vehicula elementum amet est on dui. Nulla porttitor accumsan tincidunt.
Vestibulum ac diam sit amet.
Quam vehicula elementum amet est on dui. Nulla porttitor accumsan tincidunt.


digitalagencyweb.vercel.app
Vestibulum ac diam sit amet.
Quam vehicula elementum amet est on dui. Nulla porttitor accumsan tincidunt.
See Some of our Creative Work
Nike Air Max
Lebagh
The Next Generation Payment Method
Rydhampreet Singh Gindra
The New Arrival
Nike Shoes
See More
BLOG STORIES
Check Our News
https://nike-landing-page-ten-self.vercel.app/


digitalagencyweb.vercel.app
BLOG STORIES
Check Our News
New Adventure
17 March 2019
Lorem ipsum dolor, sit amet consectetur adipisicing elit.
Aliquid est alias ullam, dolore deserunt earum tempora cum accusamus perspiciatis eius temporibus doloribus quasi modi quia esse atque officia porro numquam.
New Adventure
17 March 2019
Lorem ipsum dolor, sit amet consectetur adipisicing elit.
Aliquid est alias ullam, dolore deserunt earum tempora cum accusamus perspiciatis eius temporibus doloribus quasi modi quia esse atque officia porro numquam.
New Adventure
17 March 2019
Lorem ipsum dolor, sit amet consectetur adipisicing elit.
Aliquid est alias ullam, dolore deserunt earum tempora cum accusamus perspiciatis eius temporibus doloribus quasi modi quia esse atque officia porro numquam.
CONTACT US
Get in Touch
Your Email
Subject
Message
Send Message
DOOB


DOOB
CREATIVITY ABOVE
Home.
About Us.
Blog.
Contact Us.
Portfolio.
© 2019 - Doob, All Right Reserved

- **Accuracy:** The “DOOB” web App, implemented in ReactJS is an accurate representation of the power of ReactJS and TailwindCSS.
- **User Experience:** The user interface design, leveraging Tailwind CSS, provides a seamless and visually appealing experience across different devices and screen sizes. Interactive elements and real-time feedback enhance usability and engagement.
- **Documentation and Deployment:** Comprehensive documentation guides users through setup, usage, and troubleshooting. Deployment on a web server with domain access ensures public availability, while regular updates maintain relevance and accuracy.
- **Availability for the Community:** This project is available on Github on my Profile in a public repository so that people can check it out and can learn.

6. References

- **http://react.dev/**
- **https://tailwindcss.com/**
- **https://particles.js.org/**
- **https://www.framer.com/motion/**