***EXPERIMENT 2:***

**Design a simple webpage using bootstrap template.**

**<!DOCTYPE html>**

**<html lang="en">**

**//Head Tag**

**<head>**

**<meta charset="UTF-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<title>Week2 Bootstrap</title>**

**//Css style**

**<style>**

**footer {**

**color: white;**

**padding: 15px;**

**bottom: 0;**

**position: absolute;**

**}**

**</style>**

**//Bootstrap CDN Links**

**<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"**

**rel="stylesheet" integrity="sha384-**

**1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"**

**crossorigin="anonymous">**

**<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"**

**integrity="sha384-**

**ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"**

**crossorigin="anonymous">**

**</script></head>**

**<body>**

**<nav class="navbar navbar-expand navbar-dark bg-success">**

**<div class="container-fluid">**

**<a class="navbar-brand" href=""><img src="dora.jfif" alt="dora" width="40"**

**height="40"></a>**

**Unorder list tags and anchor links**

**tags**

**<ul class="navbar-nav me-auto mb-2 mb-lg-0">**

**<li class="nav-item">**

**<a class="nav-link active" href="">About us</a>**

**</li>**

**<li class="nav-item">**

**<a class="nav-link" href="">Contact</a>**

**</li>**

**unorder list tag**

**<li class="nav-item dropdown">**

**<a class="nav-link dropdown-toggle"**

**role="button"**

**id="navbarDropdown"**

**data-bs-toggle="dropdown"**

**href="">Episodes</a>**

**<ul class="dropdown-menu">**

**<li>**

**<a class="dropdown-item" href="">Episode1</a>**

**</li>**

**<li>**

**<a class="dropdown-item" href="">Episode2</a>**

**</li>**

**</ul>**

**</li>**

**</ul>**

**<form class="d-flex">**

**<input class="form-cotrol me-2"**

**type="search"**

**placeholder="search website"/>**

**<button class="btn btn-primary"**

**type="submit">search</button>**

**</button></form>**

**</div>**

**</nav>**

**<div class="container mt-5">**

**<div class="row">**

**<div class="col-sm-4">**

**<h3>Doraemon Biography</h3>**

**</div>**

**<div class="col-sm-8">**

**<h2>Article Title</h2>**

**<h5>Created on 06 march 2024</h5>**

**<p>Doraemon is a male robotic earless cat that travels back in time from the**

**22nd century to aid a preteen boy named Nobita.**

**An official birth certificate for the character gives him a birth date of 3**

**September 2112 and lists his city of residency as Kawasaki, Kanagawa,**

**the city where the manga was created.</p>**

**</div>**

**</div>**

**</div>**

**<footer class="container-fluid text-center bg-success">**

**<p>copy right CMRIT\_2024<br>**

**</footer>**

**//closing body and html tags**

**</body>**

**</html>**

**Exp-7**

**Write middleware to validate JWT and redirect to profile.**

**Steps:-Create package.json file initializing project using command npm(node package**

**manager)**

** npm init (or) npm init –y**

**package.json file created show above fig.1.**

** Installing require modules**

**npm i express nodemon fs path jsonwebtoken**

**Database. Json:**

**[**

**{"name":"jyotsna","work":"Assistant Professor","password":"abc"},**

**{"name":"Sarvesh","work":"Assistant Professor","password":"ambala"},**

**{"name":"ram","work":"Assistant Professor","password":"priya"},**

**{"name":"Lakshmi","work":"Asst Prof","password":"Cmrit@123"}**

**]Server.js:**

**// Import express for creating API's endpoints**

**const express = require("express");**

**const path = require('path');**

**const fs = require("fs");**

**const users = require("./database.json");**

**// Import jwt for API's endpoints authentication**

**const jwt = require("jsonwebtoken");**

**// Creates an Express application, initiate**

**// express top level function**

**const app = express();**

**// A port for serving API's**

**const port = 3000;**

**// Allow json data**

**app.use(express.json());**

**app.get('/',**

**(req, res) => {**

**res.sendFile(\_\_dirname + '/index.html');**

**});**

**app.get('/welcome',**

**(req, res) => {**

**res.sendFile(\_\_dirname + '/welcome.html');**

**});**

**app.get('/signup',**

**(req, res) => {**

**res.sendFile(\_\_dirname + '/signup.html');**

**});**

**app.get('/login',**

**(req, res) => {res.sendFile(\_\_dirname + '/login.html');**

**});**

**var database;**

**var token; app.post("/register", (req, res) => {**

**const name = req.body.name;**

**const password = req.body.password;**

**const work = req.body.work;**

**let isPresent = false;**

**let isPresentIndex = null;**

**fs.readFile("database.json", function(err, data) {**

**if (err) throw err;**

**database = JSON.parse(data);**

**for (let i = 0; i < database.length; i++) {// If data name are matched so check**

**if (database[i].name === name**

**&& database[i].password === password) {**

**isPresent = true;**

**isPresentIndex = i;**

**break;**

**}**

**}**

**if (isPresent) {**

**res.json({**

**signup: false,**

**token: null,**

**error: "Already registered",**

**});**

**} else {**

**let user =**

**{**

**name: name,**

**work: work,**

**password: password,**

**token: token**

**};**

**users.push(user);fs.writeFile(**

**"database.json",**

**JSON.stringify(users),**

**err => {**

**// Checking for errors**

**if (err) throw err;**

**// Success**

**res.json({**

**signup: true,**

**token: "generated",**

**data: "Successfully Registered",**

**});**

**console.log("Done writing");**

**});**

**}**

**});**

**});**

**app.post("/auth", (req, res) => {**

**const name = req.body.name;**

**console.log(name);**

**const password = req.body.password;**

**console.log(password)**

**let isPresent = false;**

**let isPresentIndex = null;**

**// Read database.json file fs.readFile("database.json", function(err, data) {**

**if (err) throw err;**

**database = JSON.parse(data);**

**for (let i = 0; i < database.length; i++) {**

**if (database[i].name === name**

**&& database[i].password === password) {**

**isPresent = true;**

**isPresentIndex = i;**

**break;**

**}**

**}**

**if (isPresent) {**

**const token = jwt.sign(database[isPresentIndex], "secret");**

**res.json({**

**login: true,**

**token: token,**

**data: database[isPresentIndex],**

**});**

**} else {**

**res.json({**

**login: false,**

**error: "please check name and password.",**

**});**

**}**

**});**

**});**

**app.post("/verifyToken", (req, res) => {**

**const token = req.body.token;**

**if (token) {**

**const decode = jwt.verify(token, "secret");**

**res.json({login: true,**

**data: decode,**

**});**

**} else {**

**res.json({**

**login: false,**

**data: "error",**

**});**

**}**

**});**

**app.post('/welcome',(req, res) => {**

**res.redirect("/welcome")**

**});**

**app.post('/login',(req, res) => {**

**res.redirect("/login")**

**});**

**app.post('/signup',(req, res) => {**

**res.redirect("/signup")**

**});**

**// Listen the server**

**app.listen(port, () => {**

**console.log(`Server is running :**

**http://localhost:${port}/`);**

**});**

**Login.html:<!DOCTYPE html>**

**<html>**

**<head>**

**<title>Login Page</title>**

**<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>**

**<script>**

**const form=$("#loginForm");**

**$(document).ready(function() {**

**$("#loginForm").submit((event)=>{**

**event.preventDefault();**

**var name = $("#username").val();**

**var password=$("#pwd").val();**

**$.ajax({**

**url:"/auth",**

**type:"POST",**

**contentType:"application/json",**

**dataType:"json",**

**data: JSON.stringify({**

**name: name,**

**password: password}),**

**success: function(data) {**

**var tokenData=data;**

**if(tokenData.login==true){**

**localStorage.setItem("username",tokenData.data.name);**

**localStorage.setItem("job",tokenData.data.work);**

**if(verifyLogin(tokenData.token))**

**{**

**window.location.href="/welcome";**

**}**

**else{**

**alert("Authentication Failed");**

**}**

**}},**

**error: function(data ){**

**console.log("Something went wrong");**

**}**

**});**

**});**

**});**

**function verifyLogin(token)**

**{**

**let result=true;**

**//verifyToken by making a service call to /verifyToken with get**

**$.ajax({**

**url:"/verifyToken",**

**type:"POST",**

**contentType:"application/json",**

**dataType:"json",**

**data: JSON.stringify({**

**token: token}),**

**success: function(data) {**

**if(data.login==true)**

**{**

**result=true;**

**}**

**else**

**{**

**result=false;**

**}**

**},**

**error: function(data ){**

**console.log("Wrong Token, Not Authenticated.");**

**}**

**});**

**return result;}**

**</script>**

**<body>**

**<center>**

**<h2>Sign In</h2>**

**<form id="loginForm">**

**<label for="username">Username:</label><br>**

**<input type="text" id="username" name="username"><br>**

**<label for="pwd">Password:</label><br>**

**<input type="password" id="pwd" name="pwd"><br><br>**

**<input type="submit" value="Submit">**

**</form>**

**</center>**

**</body>**

**</html>**

**Index.html:**

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<title>YOGO Solutions</title>**

**</head>**

**<body>**

**<center>**

**<a href="/signup">Click here to SignUp</a>**

**<br>**

**<a href="/login">Click here to Login</a>**

**<br>**

**</center>**

**</body>**

**</html>SignUp.html:**

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<title>SignUp</title>**

**<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>**

**<script>**

**$(document).ready(function() {**

**$('#signUpForm').submit(()=>{**

**event.preventDefault();**

**var name = $("#username").val();**

**var password=$("#pwd").val();**

**var work=$("#work").val();**

**//generateToken by making a service call to /register with post**

**$.ajax({**

**url:"/register",**

**type:"POST",**

**contentType:"application/json",**

**dataType:"json",**

**data: JSON.stringify({**

**name: name,**

**password: password,**

**work:work**

**}),**

**success: function(data) {**

**var tokenData=data;**

**if(tokenData.signup==false){**

**$("#msg").html('<h6>You are Already registered with us!!!</h6><br><a**

**id="login-link" href="\login">Login</a>');**

**}**

**else{**

**$("#msg").html('<h6>Successfully registered with us!!!</h6><br><a id="login-link" href="\login">Login</a>');**

**$("login-link").show();**

**//window.location.href="/login";**

**}**

**},**

**error: function(data ){**

**console.log("Something went wrong");**

**}**

**});**

**});**

**});**

**</script>**

**</head>**

**<body>**

**<center>**

**<h2>SignUp</h2>**

**<form id="signUpForm">**

**<label for="username">Username:</label><br>**

**<input type="text" id="username" name="username"><br>**

**<label for="work">Profession:</label><br>**

**<input type="text" id="work" name="work"><br>**

**<label for="pwd">Password:</label><br>**

**<input type="password" id="pwd" name="pwd"><br><br>**

**<input type="submit" value="Submit">**

**</form>**

**<div id="msg"></div>**

**</center>**

**</body>**

**</html>**

**Welcome.html:**

**<!DOCTYPE html>**

**<html lang="en">**

**<head><meta charset="UTF-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<title>Welcome Page</title>**

**<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>**

**<script>**

**var user=localStorage.getItem("username");**

**var job=localStorage.getItem("job");**

**$(document).ready(function(){**

**$("#user").append(user);**

**$("#job").append(job);**

**});**

**</script>**

**</head>**

**<body>**

**<center>**

**<h2>WELCOME TO YOGO SOLUTIONS</h2>**

**<hr>**

**<div style="border-style:solid; border-color: blue; width:400px;**

**height:200px">**

**<h4 id="user">USER: </h4><br>**

**<h4 id="job">PROFESSION: </h4>**

**</div>**

**</center>**

**</body>**

**</html>**

**Exp-11**

**Design Single Page Application with different menu items using react.**

**Layout.js1. import './Layout.css';2. import { Outlet, Link } from "react-router-dom";3.4. const Layout = () => {5. return (6. <>7.8. <nav className="App-header">9. <ul>10. <li className="logo">11. <h1>MyWorld</h1>12. </li>13. <li>14. <Link to="/" className="no">Home</Link>15. </li>16. <li>17. <Link to="/blogs" className="no">Blogs</Link>18. </li>19. <li>20. <Link to="/contact" className="no">Contact</Link>21. </li>22. </ul>23. </nav>24.25. <Outlet />26.27. </>28. )29.};30.31.export default Layout;Layout.css1. .App-header{background-color:aqua;}2. li{3. list-style: none;4. }5. ul {6. display: flex;7. justify-content: flex-start;8. gap: 20px;9. height:50px;10.}11.12..logo{13. line-height:10px ;14.15.}16.17..no{18. text-decoration: none;19. line-height:50px ;20. font-weight: bold;21.22.}23.Home.js1. const Home = () => {2. return <h1>Home</h1>;3. };4.5. export default Home;Contact.js1. const Contact = () => {2. return <h1>Contact Me</h1>3. };4.5. export default Contact;Blogs.js1. const Blogs = () => {2. return <h1>Blog Articles</h1>;3. };4.5. export default Blogs;NoPage.js1. const NoPage = () => {2. return <h1>404</h1>;3. };4.5. export default NoPage;n src folder write code inside index.jsindex.jsimport React from 'react'import ReactDOM from 'react-dom/client';import './index.css';import reportWebVitals from './reportWebVitals';import { BrowserRouter, Routes, Route } from "react-router-dom";import Layout from "./pages/Layout"import Home from "./pages/Home";import Blogs from "./pages/Blogs";import Contact from "./pages/Contact";import NoPage from "./pages/NoPage"const root = ReactDOM.createRoot(document.getElementById('root'));export default function App1() {return (<BrowserRouter><Routes><Route path="/" element={<Layout />}><Route index element={<Home />} /><Route path="blogs" element={<Blogs />} /><Route path="contact" element={<Contact />} /><Route path="\*" element={<NoPage />} /></Route></Routes></BrowserRouter>);}root.render(<React.StrictMode><App1 /></React.StrictMode>);reportWebVitals();**

**Exp-13**

**Design web application with different menu items using Django**

**Step1:Check Python version**

** python –version**

**Step2: Install virtual environment**

** py –m pip install virtualenv**

**Step3: create an virtual environment**

** py –m venv second**

**Step4: activate the environment**

** .\second\scripts\activate**

**Step5: install django package**

** pip install Django**

**Step6: create a project**

** django-admin startproject program13**

**Step 7: move to project folder**

** cd program13**

**Step 8: open project program13 inside create templates folder.**

**Templates folder inside create home.html, contactUs.html and dashboard.html**

**Add the below code**

** home.html**

**1. <!DOCTYPE html>**

**2. <html lang="en">**

**3. <head>**

**4. <meta charset="UTF-8">**

**5. <meta name="viewport" content="width=device-width, initial-scale=1.0">**

**6. <title>Home</title>**

**7. <link**

**href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"**

**rel="stylesheet" integrity="sha384-**

**QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH"**

**crossorigin="anonymous">**

**8. <script**

**src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"**

**integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"**

**crossorigin="anonymous"></script>**

**9. </head>**

**10.<body>**

**11. <h1 class="text-center">Welcome to CMRIT</h1>**

**12. <nav class="navbar navbar-expand-lg bg-body-tertiary">**

**13. <div class="container-fluid">**

**14. <a class="navbar-brand" href="#">CMRIT</a>15. <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data**

**bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria**

**label="Toggle navigation">**

**16. <span class="navbar-toggler-icon"></span>**

**17. </button>**

**18. <div class="collapse navbar-collapse" id="navbarNav">**

**19. <ul class="navbar-nav">**

**20. <li class="nav-item">**

**21. <a class="nav-link active" aria-current="page" href="#">Home</a>**

**22. </li>**

**23. <li class="nav-item">**

**24. <a class="nav-link" href="{% url 'dashboard' %}">Dashboard</a>**

**25. </li>**

**26. <li class="nav-item">**

**27. <a class="nav-link" href="{% url 'contactUs' %}">ContactUs</a>**

**28. </li>**

**29. </ul>**

**30. </div>**

**31. </div>**

**32. </nav>**

**33.**

**34.</body>**

**35.</html>**

** contactUs.html**

**1. <!DOCTYPE html>**

**2. <html lang="en">**

**3. <head>**

**4. <meta charset="UTF-8">**

**5. <meta name="viewport" content="width=device-width, initial-scale=1.0">**

**6. <title>ContactUs</title>**

**7. </head>**

**8. <body>**

**9. <h1>CMR Institue of Technology,Medchal, Hyderabad.</h1>**

**10. <a href="{% url 'home' %}">Back to home</a>**

**11.</body>**

**12.</html>**

** dashboard.html**

**1. <!DOCTYPE html>**

**2. <html lang="en">**

**3. <head>**

**4. <meta charset="UTF-8">**

**5. <meta name="viewport" content="width=device-width, initial-scale=1.0">**

**6. <title>Dashboard</title>**

**7. </head>**

**8. <body>**

**9. <h1>Welcome to CMRIT Dashboard</h1>**

**10. <a href="{% url 'home' %}">Back to home</a>**

**11.</body>**

**12.</html>**

**Step 9: open project program13 subfolder Write below code**

** urls.py1. from django.contrib import admin**

**2. from django.urls import path**

**3. from . import index**

**4. urlpatterns = [**

**5. #path('admin/', admin.site.urls),**

**6. path('',index.home,name='home'),**

**7. path('dashboard/', index.dashboard,name='dashboard'),**

**8. path('contactUs/', index.contactUs,name='contactUs'),**

**9. ]**

** settings.py**

**1. import os**

**2. templates =>DIRS =>os.path.join(BASE\_DIR,’templates’)**

** index.py**

**1. from django.http import HttpResponse**

**2. from django.shortcuts import render**

**3.**

**4. def home(request):**

**5. return render(request,'home.html')**

**6.**

**7. def dashboard(request):**

**8. return render(request,'dashboard.html')**

**9.**

**10.def contactUs(request):**

**11. return render(request,'contactUs.html')**

**Step 10: run the server**

** py manage.py runserver**

**Step 11: open browser, run http://127.0.0.1:8000/**