C:\Users\janani\Desktop\RGU.PNG

Informatics Institute of Technology

**Department of Computing**

Bsc (Hons) Artificial Intelligence and Data Science

**Module: CM1605 Web Technology**

**Module Coordinator: Ms. Janani Harischandra**

Group Coursework Report

Tutorial Group: Group B

Coursework Group: Group 6

Student Details: Runisi Nikoya Samaranayake – 2237028

Esandu Obadaaracchchi - 2237041

Hammaad Rizwan - 2237928

Lakidu Haputhanthri - 2237037

CONTENTS

1. Introduction …………………………………………………………………………………………………………………………………
2. Research on Numbering Systems …………………………………………………………………………………………………
3. Technical Discussion …………………………………………………………………………………………………………………….
4. Discussion of UX/UI principles/Applications/Justifications ……………………………………………………………

4.1 Navigation Techniques …………………………………………………………………………………………………………..

4.2 Color balance/selection/consistency ………………………………………………………………………………………

4.3 Color Contrast Test …………………………………………………………………………………………………………………

4.4 Typography / consistency ………………………………………………………………………………………………………

4.5 Accessibility ……………………………………………………………………………………………………………………………

4.6 Accessibility Test …………………………………………………………………………………………………………………….

4.7 Site Diagram ……………………………………………………………………………………………………………………………

05. Self-Reflection ……………………………………………………………………………………………………………………………….

06. References …………………………………………………………………………………………………………………………………….

01.introduction

1. Runisi Nikoya Samaranayake – 2237028 – Student Number 1
2. Esandu Obadaaracchchi – 2237041 – Student Number 2
3. Lakidu Haputhanthri – 2237037 – Student Number 3
4. Hammaad Rizwan – 2237928 – Student Number 4

02. Research on existing systems

* MyFitnessPal - MyFitnessPal is a free app that allows users to track their calorie intake, exercise, and weight loss progress. It has a vast database of foods, which allows users to easily log their food intake and get a more accurate estimate of their calorie intake. It also integrates with several fitness tracking devices and apps, such as Fitbit and Apple Health. MyFitnessPal provides personalized recommendations for calorie intake and nutrient goals based on the user's weight, height, and activity level.
* Lose It - Lose It! is another free app that allows users to track their food intake, exercise, and weight loss progress. It also has a large database of foods and allows users to scan barcodes for quick and easy logging. The app also has a feature that connects users with a community of people who are also working towards their weight loss goals. Lose It! also provides personalized recommendations for calorie intake and nutrient goals based on the user's weight, height, and activity level.

03. TECHNICAL DISCUSSION

STUDENT 1 – RUNISI NIKOYA SAMARANAYAKE

PRESENTATION PAGE

**HTML CODE**

<!DOCTYPE html>

<html>

<head>

    <title>Page Title</title>

    <link rel="stylesheet" type="text/css" href="presentation.css">

</head>

<body onload="startTimer()">

    <div class="background-image">

    </div>

    <script>

        function startTimer() {

            setTimeout(function() {

                window.location.href = "trial1.html";

            }, 5000); // 5000 milliseconds = 5 seconds

        }

    </script>

</body>

</html>

**EXPLANATION**

This code is a basic HTML web page that has a background image and a timer that redirects the user to another web page after five seconds.

The HTML code includes a head section where the page title and the link to an external stylesheet file are defined. The body section contains a div tag with the class "background-image" that defines the background image of the web page.

In the body section, the "onload" event is used to trigger the "startTimer()" function when the web page finishes loading. The "startTimer()" function uses the JavaScript "setTimeout" method to delay the redirection of the user to another web page. In this case, after five seconds, the user will be redirected to the Main web page.

The CSS code defines the "background-image" class, which sets the background image for the web page, resizes it to fit the page, and sets its height and width.

SIGN-UP FORM

**HTML CODE**

<!DOCTYPE html>

<html>

    <head>

        <meta charset="UTF-8">

        <link rel = "stylesheet" type = "text/css" href = "coursework (1).css">

    </head>

    <body>

    <header>

        <div class="logo">

            <img class="logo-pic" src="logo.png" alt="logo">

        </div>

        <div class="heading">

        </div>

        <div class="button-section">

          <button class="navi-buttons">

            <a class="button-text" href="#laki add link store" target="\_blank">Online store</a>

            </button>

            <button class="navi-buttons">

              <a class="button-text" href="#hammaad add link mylist" target="\_blank">My list </a>

            </button>

            <button class="navi-buttons">

              <a class="button-text" href="galleryyyy.html" target="\_blank">Healthy foods</a>

            </button>

            <button class="navi-buttons">

                <a class="button-text" href="coursework (1).html" target="\_blank">Sign Up</a>

            </button>

        </div>

    </header>

        <div class = "img"></div>

    <div class="Background-image"></div>

        <br>

        <br>

    <div class="back-ground">

        <h1>SIGN UP</h1>

        <form id="signup-form">

          <div class="form-group">

            <label for="name">NAME:</label>

            <input type="text" id="name" name="name" >

          </div>

          <div class="form-group">

            <label for="DATE OF BIRTH"> DATE OF BIRTH:</label>

            <input type="date" id="date" name="date" >

            <label for="gender">GENDER:</label>

            <select id="gender" name="gender" >

                <option value="" disabled selected></option>

                <option value="male">Male</option>

                <option value="female">Female</option>

                <option value="other">Other</option>

            </select>

          </div>

          <div class="form-group">

            <label for="email">EMAIL:</label>

            <input type="email" id="email" name="email" >

          </div>

          <button type="submit" hre>NEXT</button>

          <div class="login-container">

          <p>ALREADY A MEMBER?</p>

            <a href="logo.html" target="\_blank">LOGIN</a>

          </div>

        </form>

        <div id="subpage1">

            <form id="subpage1-form">

                <div class="form-group">

                  <label for="physical-activity">PHYSICAL ACTIVITY:</label>

                  <input type="text" id="physical-activity" name="physical-activity" list="search">

                  <datalist id="search">

                    <option value="Cardio-Vascular">

                        <option value="Strength Training">

                        <option value="Yoga and Pilates">

                        <option value="Sports">

                        <option value="Outdoor activities">

                        <option value="Dancing">

                        <option value="Martial arts">

                    </option>

                  </datalist>

                </div>

                <div class="form-group">

                  <label for="food-intake">FOOD INTAKE:</label>

                  <input type="text" id="food-intake" name="food-intake" list="search-list">

                  <datalist id="search-list">

                    <option value="Vegetarian">

                        <option value="Low-Carb">

                        <option value="Fast-food">

                        <option value="Paleo">

                        <option value="Flexitarian">

                        <option value="Mediterranean">

                        <option value="Gluten-free">

                    </option>

                  </datalist>

                </div>

                <div class="form-group">

                  <label for="workout-time">TIME OF WORKOUTS:</label>

                  <input type="time" id="workout-time" name="workout-time">

                </div>

                <div class="form-group">

                  <label for="meals">NO OF MEALS PER DAY:</label>

                  <input type="text" id="meals" name="meals" list="search-menu">

                  <datalist id="search-menu">

                    <option value="1">

                        <option value="2">

                        <option value="3">

                        <option value="4">

                    </option>

                  </datalist>

                </div>

                <button type="submit">Next</button>

                <div class="login-container">

                    <p>ALREADY A MEMBER?</p>

                    <a href="#" target="\_blank">LOGIN</a>

                </div>

            </form>

        </div>

**JAVASCRIPT CODE**

<script>

            // get form elements

            const signupForm = document.getElementById("signup-form");

            const subpage1Form = document.getElementById("subpage1-form");

            // get form buttons

            const nextBtn1 = document.querySelector("#signup-form button[type='submit']");

            const nextBtn2 = document.querySelector("#subpage1-form button[type='submit']");

            // get form pages

            const page1 = document.getElementById("signup-form");

            const page2 = document.getElementById("subpage1");

            // listen for form submissions

            signupForm.addEventListener("submit", (e) => {

            // prevent form from submitting

            e.preventDefault();

            // validate form fields

            const name = signupForm.elements["name"].value;

            const fname = name.split(" ")[0];

            const lname = name.split(" ")[1];

            const date = signupForm.elements["date"].value;

            const gender = signupForm.elements["gender"].value;

            const email = signupForm.elements["email"].value;

            if (fname === "" || lname===""|| date === "" || gender === "" || email === "") {

                alert("Please fill out all required fields");

                return;

            }

            const nameRegex = /^[A-Za-z]+$/;

            if (!nameRegex.test(fname) || !nameRegex.test(lname)) {

                alert("Name should contain only letters");

                return;

            }

            if (name.split(" ").length>2||name.split(" ").length<2){

                alert("Enter the full name(first name and last name)");

                return;

            }

            if (fname === "" || lname===""|| date === "" || gender === "" || email === "") {

                alert("Please fill out all required fields");

                return;

            }

            // show subpage 1

            page1.style.display = "none";

            page2.style.display = "block";

            });

            subpage1Form.addEventListener("submit", (e) => {

            // prevent form from submitting

            e.preventDefault();

            // validate form fields

            const physicalActivity = subpage1Form.elements["physical-activity"].value;

            const foodIntake = subpage1Form.elements["food-intake"].value;

            const workoutTime = subpage1Form.elements["workout-time"].value;

            const meals = subpage1Form.elements["meals"].value;

            if (physicalActivity === "" || foodIntake === "" || workoutTime === "" || meals === "") {

                alert("Please fill out all required fields");

                return;

            }

            // submit form data

            alert("Dear "+ signupForm.elements["name"].value.split(" ")[0]+" thank you for using Fitzen, The results will be shown in a while!");

            });

        </script>

This code creates a basic HTML page that includes a form with two pages. The first page asks for the user's name, date of birth, gender, and email address. The user can click a "NEXT" button to move to the second page, which asks for information about the user's physical activity, food intake, workout time, and number of meals per day.

The form is designed using HTML and styled using CSS. It includes a logo and navigation bar at the top, as well as a background image.

The code uses JavaScript to listen for form submissions and to switch between the two form pages when the "NEXT" button is clicked. It gets the form elements and buttons by their IDs, and it gets the form pages by their HTML element IDs. When the "NEXT" button is clicked, the code hides the current form page and shows the next form page.

The code also includes some basic CSS styles to format the page and the form. It uses a logo image, a background image, and some buttons to create a simple navigation menu.

**EXPLANTION OF THE JAVASCRIPT CODE**

This code is responsible for implementing a two-page form that collects user information related to fitness and health. The code uses JavaScript to handle form submissions, validate user input, and display error messages when necessary. . The code includes event listeners that wait for the form submissions and a conditional display that shows the next page of the form only if the previous page has been successfully submitted.

The first part of the code gets references to various elements in the HTML document using the **getElementById()** and **querySelector()** methods. These elements include the two form pages (**signup-form** and **subpage1**), the form buttons, and the form fields.

Next, event listeners are added to the forms to wait for form submissions. When the user clicks the "Next" button on the first page, the event listener for **signupForm** hides the first page and shows the second page using the **style.display** property.

The code then adds event listeners to the two forms that listen for the **submit** event. When the user submits the form, the code prevents the default form submission behavior using **preventDefault().**The event listeners prevent the default behavior of form submissions using **preventDefault()**. This is done to prevent the forms from being submitted and the page from reloading before the necessary actions are taken.

The first form is validated by checking if all required fields are filled in using an if statement. The fields are then validated for specific criteria such as name containing only letters, and name consisting of first name and last name. If any validation fails, an error message is displayed using the alert() method. If all validation passes, the first form is hidden, and the second form is shown using style.display property.

The second form is also validated for all required fields. If validation passes, an alert message is displayed thanking the user for using the form, and the data is submitted.

GALLERY PAGE

**HTML CODE**

<!DOCTYPE html>

<html>

<head>

    <title>HEALTHY FOODS</title>

    <link rel="stylesheet" href="gaall.css">

</head>

<body>

    <header>

        <div class="logo">

            <img class="logo-pic" src="logo.png" alt="logo">

        </div>

        <div class="heading">

        </div>

        <div class="button-section">

          <button class="navi-buttons">

            <a class="button-text" href="#laki add link store" target="\_blank">Online store</a>

            </button>

            <button class="navi-buttons">

              <a class="button-text" href="#hammaad add link mylist" target="\_blank">My list </a>

            </button>

            <button class="navi-buttons">

              <a class="button-text" href="galleryyyy.html" target="\_blank">Healthy foods</a>

            </button>

            <button class="navi-buttons">

            <a class="button-text" href="coursework (1).html" target="\_blank">Sign Up</a>

            </button>

        </div>

    </header>

    <h1 id="head">HEALTHY FOODS</h1>

    <div class="gallery">

        <img class="thumbnail" src="pic.jpg" alt="Grilled chicken with vegetables is a healthy and delicious dish that features juicy and flavorful chicken breasts alongside a colorful variety of grilled vegetables. The chicken is often marinated in a blend of herbs, spices, and citrus juices to infuse it with extra flavor, while the vegetables are brushed with olive oil and seasoned with salt and pepper before being grilled until they are tender and slightly charred. This dish is a perfect option for those who want to enjoy a satisfying meal that is packed with nutrients and bursting with flavor. It's a great way to get in your daily dose of veggies and protein in one delicious dish!">

        <img class="thumbnail" src="site-smoothie.jpg" alt="Green smoothie bowl is a nutritious and refreshing breakfast or snack option made by blending together a variety of fresh fruits and vegetables with almond milk or coconut water, and topping it with delicious ingredients like sliced fruits, nuts, and seeds. This meal is packed with essential vitamins, minerals, and antioxidants, making it a great way to boost your overall health and wellbeing. Green smoothie bowls are easy to customize and enjoy at any time of the day, making them a delicious and healthy addition to your diet.">

        <img class="thumbnail" src="pic2.jpg" alt="Quinoa salad with avocado and mango is a healthy and flavorful meal made by combining cooked quinoa with diced mango, avocado, red onion, and cilantro. A tangy dressing made from lime juice, olive oil, honey, and salt adds the perfect balance of sweet and savory flavors. This salad is a great source of protein, fiber, and healthy fats, making it a perfect option for those who are looking to eat clean or add more plant-based protein to their diet. It is easy to customize and can be enjoyed as a main course or side dish, making it a versatile and delicious addition to any meal.">

        <img class="thumbnail" src="pic3.jpg" alt="Roasted sweet potato and black bean tacos are a delicious and healthy meal made by roasting diced sweet potatoes until tender and combining them with black beans, red onion, and flavorful spices. Topped with fresh cilantro, crumbled feta cheese, and a squeeze of lime juice, these tacos are packed with essential nutrients such as fiber, vitamins, and protein. This dish is easy to customize with your favorite toppings and can be enjoyed as a satisfying main course or as a fun and flavorful appetizer. The combination of sweet and savory flavors makes these tacos a delicious and nutritious addition to your meal repertoire.">

        <img class="thumbnail" src="pc4.jpg" alt="Banana oat pancakes are a delicious and nutritious breakfast option made by blending together rolled oats, banana, eggs, and milk. The batter is then cooked on a griddle until golden brown, creating fluffy and satisfying pancakes that are packed with fiber, protein, and essential vitamins. These pancakes are easy to make and can be customized with your favorite toppings such as fresh fruit, honey, or nut butter. They are a perfect meal for those who are looking for a healthy and satisfying breakfast that will keep them full and energized throughout the day. Try making banana oat pancakes for your next breakfast and enjoy a delicious and nutritious start to your day!">

    </div>

    <div class="large-image">

        <img id="main-image" src="" alt="">

        <p class="description" id="image-description"></p>

    </div>

    <script>

        // Get all the thumbnail images

        const thumbnails = document.querySelectorAll('.thumbnail');

        // Get the main image and image description elements

        const mainImage = document.getElementById('main-image');

        const imageDescription = document.getElementById('image-description');

        // Set up an event listener for each thumbnail

        thumbnails.forEach(thumbnail => {

            thumbnail.addEventListener('click', () => {

                // Remove the selected class from all thumbnails

                thumbnails.forEach(t => t.classList.remove('selected'));

                // Add the selected class to the clicked thumbnail

                thumbnail.classList.add('selected');

                // Set the main image source and alt text to the clicked thumbnail's attributes

                mainImage.src = thumbnail.src;

                mainImage.alt = thumbnail.alt;

                // Set the image description to the clicked thumbnail's alt text

                imageDescription.textContent = thumbnail.alt;

            });

        });

    </script>

</body>

</html>

**EXPLANATION**

This is a webpage displaying a gallery of healthy food images. The gallery is interactive and allows the user to click on a thumbnail image to display a larger version of the image and a description.The webpage contains a header with a logo and navigation buttons to an online store, a user's list, healthy foods gallery, and a sign-up page. The main content of the webpage contains a title, an h1 heading that says "HEALTHY FOODS", and a div with a class of "gallery" that contains five images of healthy food each with a brief description.. Each image has a class of "thumbnail" and an alt attribute that describes the image.

The document also contains a div with a class of "large-image" that displays a larger version of the selected thumbnail image and its description. The script at the end of the document listens for a click event on each thumbnail image and updates the larger image and its description accordingly.

The webpage uses HTML, CSS, and JavaScript to create the layout, style, and functionality of the webpage. The CSS file is linked to the HTML file to apply styles to the webpage, including font styles, color schemes, and layout styles. The JavaScript code is used to add functionality to the webpage by implementing a thumbnail image gallery feature. When a thumbnail image is clicked, the selected image is displayed in a larger size along with its corresponding description.

**EXPLANTION OF THE JAVASCRIPT CODE**

This JavaScript code selects all the thumbnail images on a web page using the "querySelectorAll" method, and stores them in a variable called "thumbnails". It also selects two other elements, the main image and the image description, and stores them in the "mainImage" and "imageDescription" variables, respectively.

Next, the code sets up an event listener for each thumbnail using a "forEach" loop. When a thumbnail is clicked, the code removes the "selected" class from all thumbnails, adds the "selected" class to the clicked thumbnail, and sets the source and alt text of the main image to the clicked thumbnail's source and alt attributes, respectively. It also sets the text content of the image description element to the alt text of the clicked thumbnail.

This functionality allows users to click on a thumbnail image and see a larger version of the image in the main image area of the webpage, with a corresponding description of the image shown below it.

STUDENT 2 - Esandu Obadaaracchchi