BS Computer Science

Hitec University, Taxila

Project Title

Preventive Healthcare Guide - A Static Informational Website

Applications Of Information and Communication Technologies Lab

Semester: 1st Semester

Section: A

Submitted By: Hafsa Shahid

Submitted To: Muhammad Khalid

Roll No: 25-CS-181

Table of Contents

1. Executive Summary

This project involves the design and development of a static website titled **"Preventive Healthcare Guide"** using **HTML and CSS**.

The website aims to educate users about preventive healthcare practices such as healthy lifestyle habits, regular medical checkups, and disease prevention tips. The project focuses on proper web page structure, clean styling, and responsive layout design. This website demonstrates fundamental front-end development skills learned during the AICT lab.

2. Introduction

2.1 Project Background

Preventive healthcare plays an essential role in maintaining overall well-being and reducing the risk of serious diseases. With increasing health awareness, an informational platform can help people adopt healthier habits before illness occurs.

2.2 Objective

The main objective of this project is:

- To design a multi-page static website using HTML and CSS
- To present healthcare information in a clear and organized manner
- To practice layout design, navigation, forms, and responsiveness

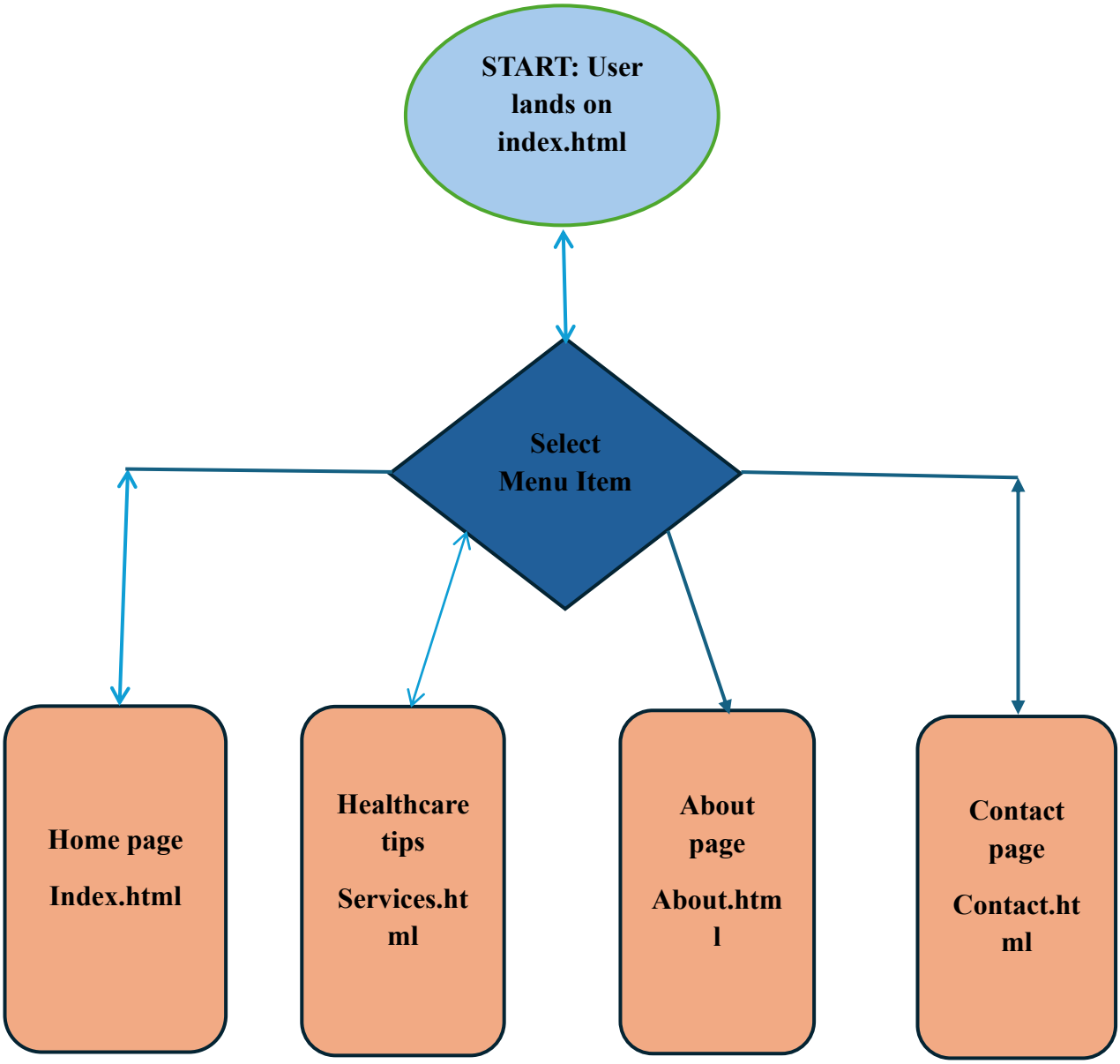
2.3 Scope

This project is limited to:

- Static web pages
- Informational content only
- No backend or JavaScript functionality

3. System Design

3.1 Flowchart



4. Implementation

4.1 Technologies Used

The project was developed using a standard web technology stack to ensure cross-browser compatibility and responsive performance:

- **HTML5:** Used to create the semantic structure of the website, including headers, navigation menus, interactive sections, and the healthcare data table.
- **CSS3 (Flexbox):** Employed for the entire visual layout. Flexbox was specifically used to create the dynamic 4-column grid and the centered navigation bar. It ensures that elements align perfectly regardless of screen size.
- **JavaScript (ES6):** Utilized to add functional logic. Specifically, it handles the Document Object Model (DOM) manipulation for the "click-to-reveal" info boxes and provides client-side validation for the contact form.
- **Optimized Media:** A combination of **JPG** and **WebP** formats was used. WebP was selected for the contact and service pages to provide high-quality visuals with smaller file sizes for faster loading.

4.2 Code Structure and Organization

The project is organized into a clean, flat directory structure to ensure easy maintenance and clear linking between pages.

File Organization:

- **`index.html`**: The homepage and entry point of the application. It contains the hero banner and the primary welcome message.
- **`AICT Lab project about.html`**: Contains the interactive Flexbox layout where users can click boxes to learn about preventive care topics.
- **`AICT Lab project services.html`**: Houses the tabular data representing healthcare habits and their benefits.
- **`AICT Lab project contact.html`**: Features the user feedback form and contact information.
- **`script.js`**: A centralized JavaScript file that contains the `toggleInfo()` function (used by the flex items) and the `validateForm()` logic.
- **Internal CSS:** Styling is handled within `<style>` tags in the head of each document to ensure page-specific layouts (like the Table vs. the Flexbox) load correctly without external dependencies.
- **`AICT Lab project/` Folder**: A dedicated directory for all image assets to keep the root folder organized.

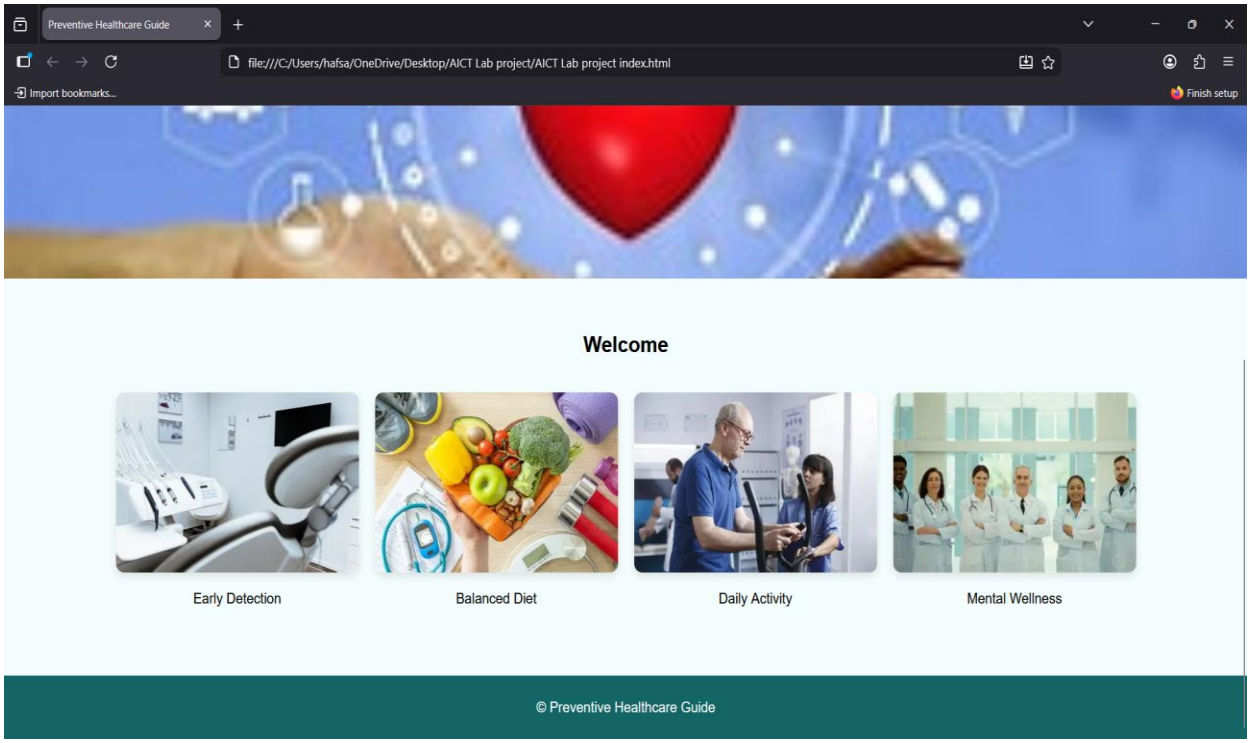
4.3 Implementation Highlights

- **Responsiveness:** Implementation of `@media` queries ensures that when the screen width is less than 600px, the navigation and flex-containers switch from a horizontal row to a vertical stack.

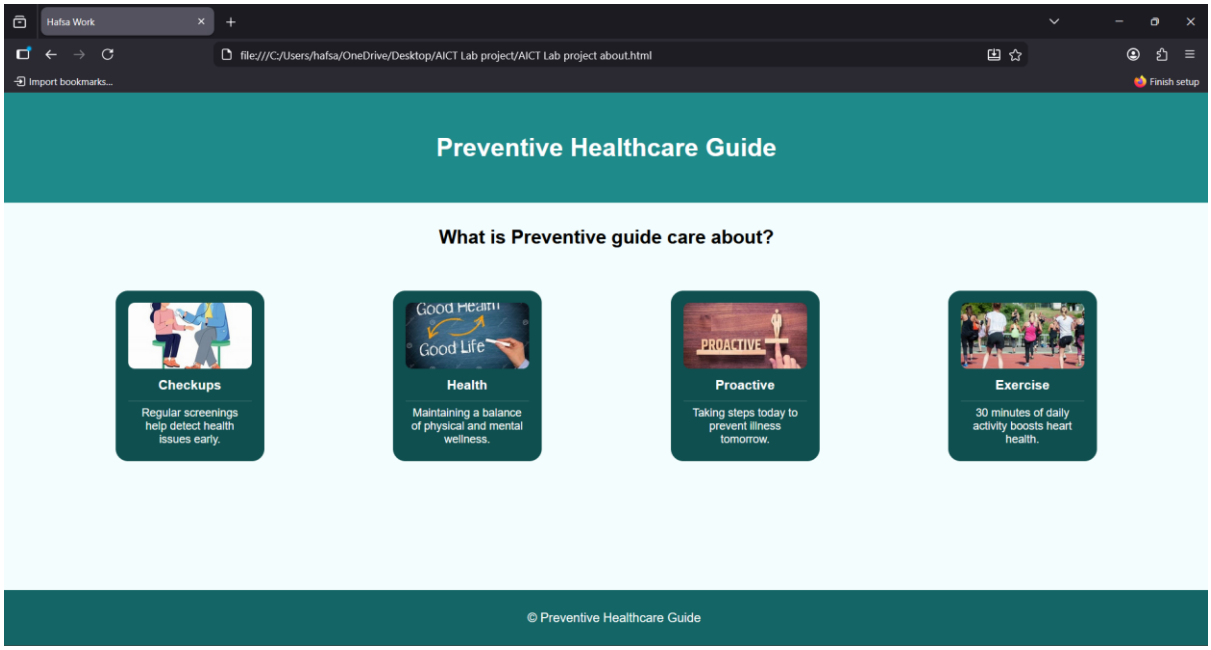
5. Results and Challenges

5.1 Images of the final website:

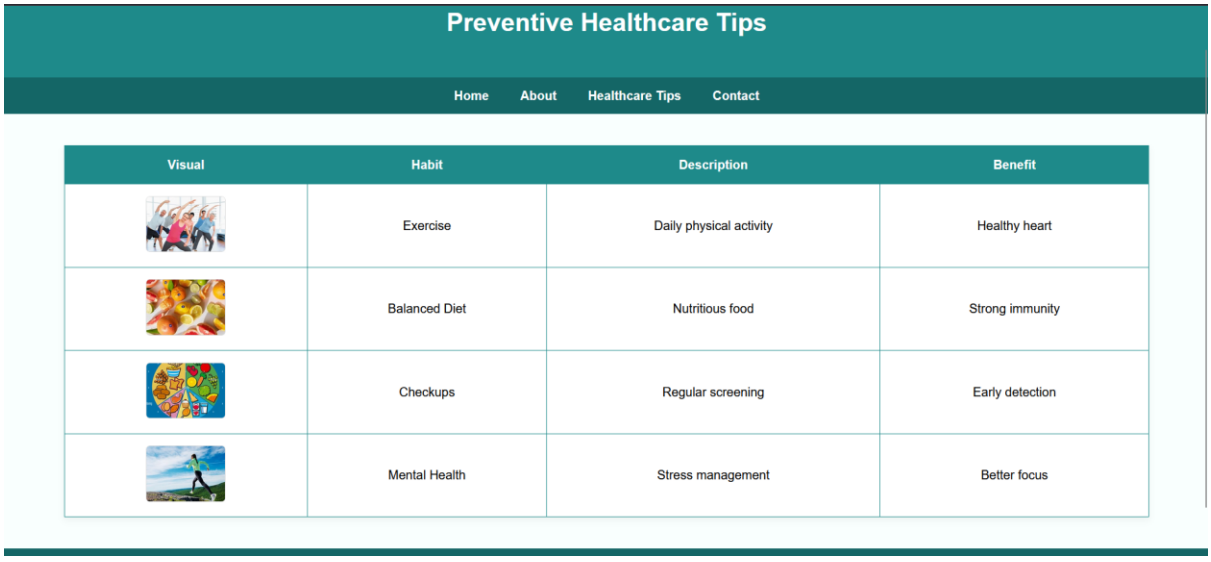
Index page:



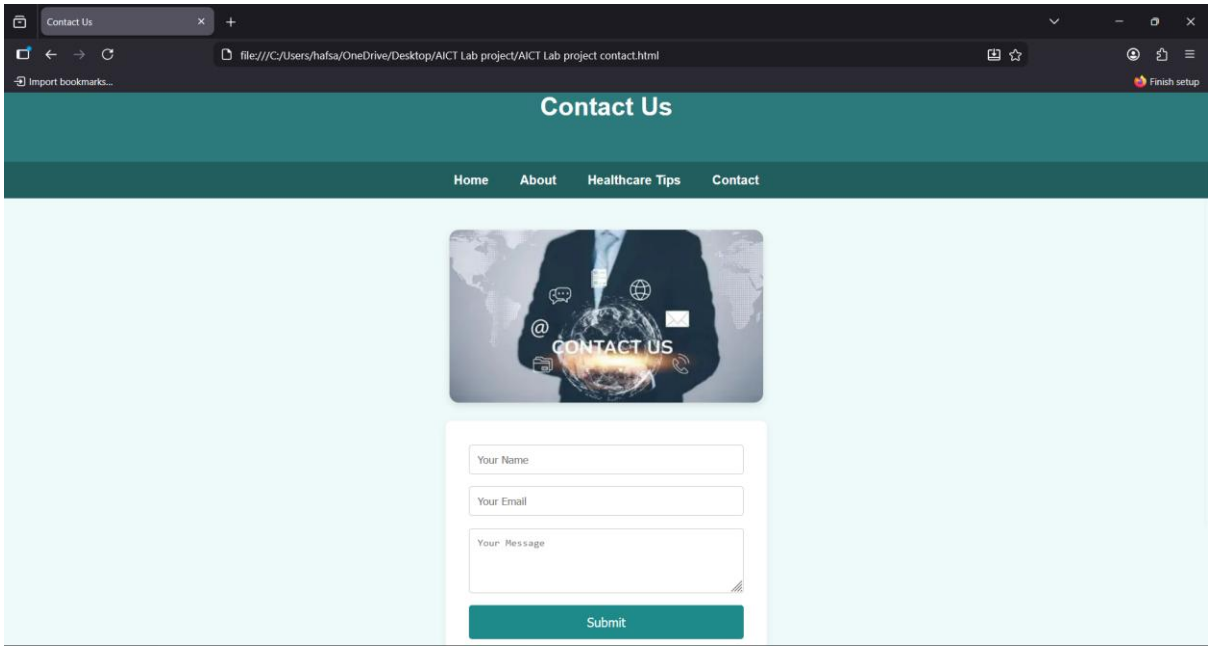
About page: It is a responsive design.



Healthcare tips page:



Contact Us page:



5.2 The challenges

File Path Discrepancies: The browser could not find images due to spaces in folder names and incorrect relative paths.

Extension Mismatch: Errors occurred when the code called for .jpg but the actual files were, webp or had capitalized extensions like .JPG.

Case Sensitivity: Issues arose because web servers often treat Image.jpg and image.jpg as two completely different files.

Responsive Layout Issues: Content originally designed for desktop would overflow or look cramped on mobile screens before adding Media Queries.

JavaScript Targeting: Ensuring that clicking one "flex-item" only opened that specific box rather than all of them at once.

Fixed Height Constraints: Hard-coded box heights caused text to overlap or be cut off when the interactive info was revealed.

6. Appendices

GitHub link: [25-cs-181-haf/Applications-of-Information-and-Communication-Technologies: Website design by a programmer.]