Development part 2:

Public Transport Optimization:

Creating a complete public transport optimization system in HTML, CSS, and JavaScript is a complex task that would typically require a back-end server for data processing and storage.

Index.html

<!DOCTYPE html>

<html>

<head>

<title>Public Transport Optimization</title>

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

</head>

<body>

<h1>Public Transport Route Planner</h1>

<div id="input-section">

<label for="start-stop">Start Stop:</label>

<input type="text" id="start-stop" placeholder="Enter start stop">

<br>

<label for="end-stop">End Stop:</label>

<input type="text" id="end-stop" placeholder="Enter end stop">

<br>

<button onclick="planRoute()">Plan Route</button>

</div>

<div id="result-section">

<h2>Optimal Route:</h2>

<p id="optimal-route">No route planned yet</p>

</div>

<script src="script.js"></script>

</body>

</html>

Style.css

body {

font-family: Arial, sans-serif;

}

h1 {

text-align: center;

}

#input-section {

margin: 20px;

padding: 20px;

border: 1px solid #ccc;

border-radius: 5px;

}

label {

font-weight: bold;

}

input {

width: 100%;

padding: 5px;

margin: 5px 0;

}

button {

background-color: #007BFF;

color: white;

padding: 10px 20px;

border: none;

border-radius: 5px;

cursor: pointer;

}

#result-section {

display: none;

margin: 20px;

padding: 20px;

border: 1px solid #ccc;

border-radius: 5px;

}

#optimal-route {

font-weight: bold;

}

Script.js

function planRoute() {

const startStop = document.getElementById('start-stop').value;

const endStop = document.getElementById('end-stop').value;

// In a real system, you would make an API call to get the optimal route.

// For this example, we'll use a simple message.

const message = `Start at ${startStop}, take Route X, change at Stop Y, continue to ${endStop}.`;

const resultSection = document.getElementById('result-section');

const optimalRoute = document.getElementById('optimal-route');

optimalRoute.textContent = message;

resultSection.style.display = 'block';

}

Android App(Java)

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

EditText startStopEditText;

EditText endStopEditText;

Button planRouteButton;

TextView resultTextView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

startStopEditText = findViewById(R.id.startStopEditText);

endStopEditText = findViewById(R.id.endStopEditText);

planRouteButton = findViewById(R.id.planRouteButton);

resultTextView = findViewById(R.id.resultTextView);

planRouteButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String startStop = startStopEditText.getText().toString();

String endStop = endStopEditText.getText().toString();

// In a real system, you would call a route optimization function here.

// For this example, we'll display a simple message.

String message = "Start at " + startStop + ", take Route X, change at Stop Y, continue to " + endStop + ".";

resultTextView.setText(message);

}

});

}

}

TEAM MEMBER:

Jayapriya.S(620821205022)

Sneka.G(620821205053)

Gopika.P(620821205016)

Vaishnavi Devi.R(620821205022)