**REPÚBLICA BOLIVARIANA DE VENEZUELA**

**UNIVERSIDAD RAFAEL BELLOSO CHACÍN**

**FACULTAD DE INGENIERIA**

**ESCUELA DE INFORMATICA**

**CÁTEDRA: SIMULACIÓN DE SISTEMAS**

**SECCIÓN: N1113**

****

**INFORME TECNICO DE APP DE SIMULACIÓN**

**PRESENTADO POR:**

**Esteban Galban, 30274211.**

**Diego Jordan, 29939977.**

**Valeria Velásquez, 27412794.**

**Maracaibo, 26 de Julio de 2024**

**INTRODUCCIÓN:**

El objetivo de este informe es presentar la simulación de tráfico para la vía rápida Circunvalación 1 en Maracaibo, Estado Zulia. Dicha simulación busca analizar el flujo vehicular y los posibles retrasos en función de diferentes parámetros y tiempos, proporcionando así una herramienta para optimizar la gestión del tráfico en esta importante arteria vial. El sistema se compone de una interfaz de usuario en HTML y un conjunto de scripts en JavaScript para la lógica de la simulación.

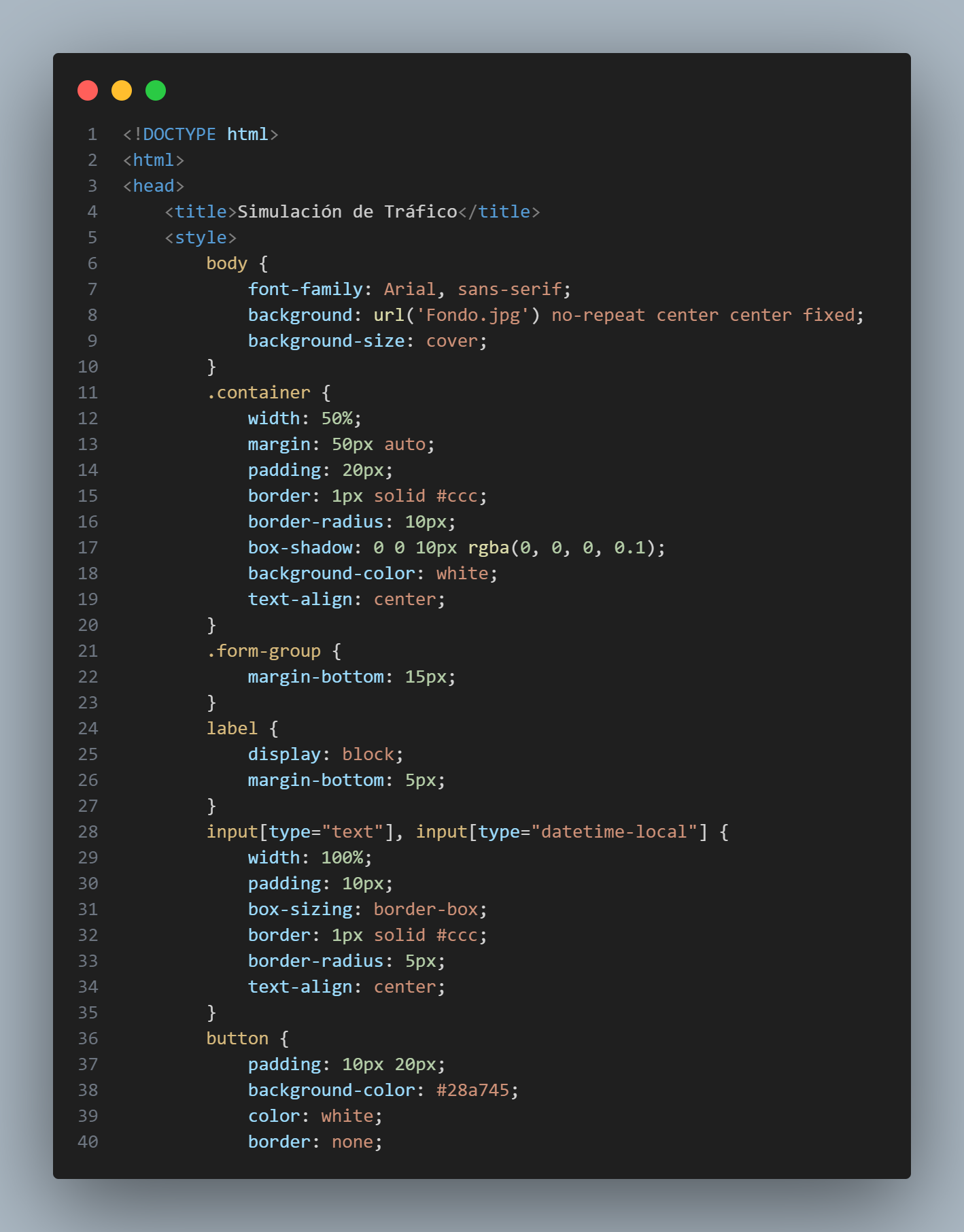
**DESARROLLO:**

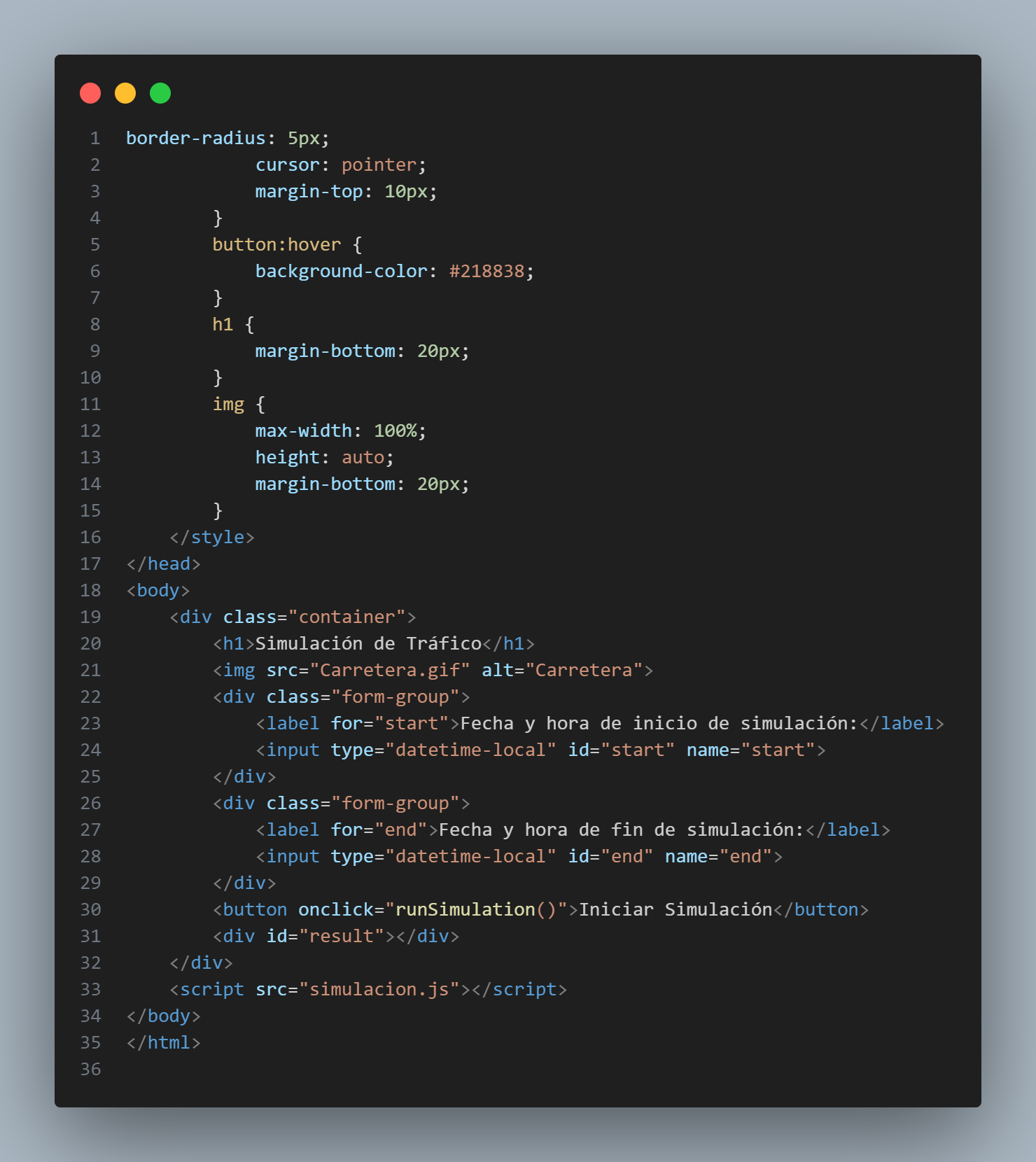
**1. Código HTML**

**1.1 Descripción General**

El código HTML proporciona la estructura básica de la interfaz de usuario para la simulación de tráfico. La interfaz permite al usuario ingresar la fecha y hora de inicio y fin de la simulación, y visualizar los resultados generados por el script JavaScript.

**1.2 Estructura del HTML**

******

******

**1.3 Descripción de los Elementos**

1. **<!DOCTYPE html>:** Declara el tipo de documento y la versión de HTML utilizada, en este caso, HTML5.
2. **<html>:** Elemento raíz del documento HTML.
3. **<head>:** Contiene metadatos sobre el documento, incluyendo el título y los estilos CSS.
4. **<title>:** Define el título de la página que aparece en la pestaña del navegador.
5. **<style>:** Sección para incluir estilos CSS que definen la apariencia de la página. Aquí se especifican estilos para el cuerpo, contenedor, grupos de formularios, botones e imágenes.
6. **<body>:** Contiene el contenido visible de la página.
7. **<div class="container">:** Un contenedor centrado con estilo para la simulación, incluyendo el título, imagen, campos de entrada, botón y área para los resultados.
8. **<img src="Carretera.gif" alt="Carretera">:** Imagen que se muestra en la interfaz, proporcionando una representación visual del tráfico.
9. **<div class="form-group">:** Contiene los elementos de entrada para las fechas y horas de inicio y fin de la simulación.
10. **<button onclick="runSimulation()">Iniciar Simulación</button>:** Botón que inicia la simulación llamando a la función runSimulation() cuando se hace clic.
11. **<div id="result"></div>:** Área donde se muestran los resultados de la simulación.
12. **<script src="simulacion.js"></script>:** Enlace al archivo JavaScript que contiene la lógica de la simulación.

**2. Código JavaScript**

**2.1 Descripción General**

El código JavaScript gestiona la lógica de la simulación de tráfico, calculando el flujo de vehículos y los retrasos en función de los datos de entrada. El script se activa cuando el usuario inicia la simulación desde la interfaz HTML.

**2.2 Estructura del JavaScript**

(Pagina Siguiente).

****

**2.3 Descripción de las Funciones**

1. **runSimulation()**
   * **Propósito**: Ejecuta la simulación de tráfico entre las fechas y horas de inicio y fin proporcionadas por el usuario.
   * **Proceso**:
     + Obtiene y valida las fechas de inicio y fin.
     + Define los parámetros de la simulación, incluidos los datos de tráfico y las interrupciones.
     + Calcula el flujo vehicular y los retrasos en un bucle mientras actualiza la dirección de tráfico.
     + Muestra los resultados en la página.
2. **getTrafficFlow(time, direction, data, isHoliday)**
   * **Propósito**: Determina el número de vehículos en función de la hora, la dirección y si es un día festivo.
   * **Proceso**:
     + Identifica el día de la semana y la hora.
     + Retorna el flujo de vehículos según el tipo de día y la dirección, usando un índice de hora pico.
3. **getDelay(time, direction, data, isHoliday)**
   * **Propósito**: Calcula el retraso en la circulación basado en el día de la semana, la dirección del tráfico y si es un día festivo.
   * **Proceso**:
     + Devuelve el retraso basado en el día y la dirección, o 0 si es un festivo.
4. **getPeakHourIndex(hour, peakHours)**
   * **Propósito**: Calcula el índice de la hora pico en función de la hora actual.
   * **Proceso**:
     + Determina el índice de hora pico basado en comparaciones de la hora actual con las horas pico definidas.
5. **checkIfHoliday(date)**
   * **Propósito**: Verifica si la fecha dada corresponde a un día festivo.
   * **Proceso**:
     + Compara la fecha dada con una lista de días festivos definidos para determinar si es un festivo.

**3. RESUMEN DE RESULTADOS**

1. **Interfaz de Usuario:** El diseño HTML ofrece una interfaz intuitiva para la entrada de datos y la visualización de resultados. Incluye campos para la selección de fechas y horas, un botón para iniciar la simulación, y un área para mostrar los resultados generados.
2. **Lógica de Simulación:** El código JavaScript maneja la lógica compleja de la simulación, incluyendo la gestión del tiempo, el cálculo del flujo vehicular y los retrasos, y la consideración de días festivos. Las funciones están diseñadas para proporcionar resultados precisos y coherentes, considerando tanto las horas pico como las condiciones especiales.
3. **Validación y Resultados:** El sistema verifica las entradas del usuario para asegurar que sean válidas y muestra los resultados de manera clara y comprensible. La simulación alterna la dirección del tráfico y calcula el flujo y los retrasos en función de parámetros predefinidos y condiciones dinámicas.

**4. IMPACTO**

La simulación proporciona una herramienta valiosa para la planificación y gestión del tráfico en la Circunvalación 1, permitiendo a los responsables de la gestión de tránsito tomar decisiones informadas basadas en datos precisos. La interfaz de usuario facilita la interacción y el análisis, mientras que la lógica de simulación garantiza resultados confiables y útiles para la optimización del flujo vehicular.

**5. FUTURAS MEJORAS**

Se podrían considerar futuras mejoras como:

* **Integración de Datos en Tiempo Real:** Incorporar datos en tiempo real para una simulación más dinámica.
* **Optimización de la Interfaz:** Mejorar la interfaz gráfica para una experiencia de usuario más rica y visualmente atractiva.
* **Expansión de la Simulación:** Incluir más parámetros y escenarios para un análisis más completo del tráfico.

**CONCLUSIÓN**

La simulación de tráfico para la vía rápida Circunvalación 1 en Maracaibo ha sido implementada con éxito a través de una interfaz de usuario HTML y una lógica de procesamiento en JavaScript. El sistema permite a los usuarios ingresar las fechas y horas de inicio y fin de la simulación, proporcionando un análisis detallado del flujo vehicular y los retrasos esperados en función de diferentes condiciones.