

**Universidad San Carlos de Guatemala
Centro Universitario De Occidente
División de Ciencias de la Ingeniería**

Modelacion y simulacion 1
Ing. Pedro Domingo

Byron Fernando Torres Ajxup

201731523

ID 16

Manual Técnico

**Simulador de Tráfico: Socavamiento Km 194 – Ruta Cito-Zarco
Versión: 2.0 – Diciembre 2025**

Quetzaltenango 1 de diciembre del 2025

Índice

Introducción al Simulador	3
Primeros Pasos	3
Cómo iniciar una simulación	4
Explorar vehículos	5
Alertas de accidentes	5
Ver historial de accidentes	5
Controles Detallados	6
Panel de configuración (izquierda)	6
Gráficos en vivo (inferior)	6
Escenarios	7
4.1 Escenario “Ideal”	7
Escenario “Caos”	7
Consejos y Mejores Prácticas	7
Solución de Problemas	7

Arquitectura de Software

Stack Completo

Capa	Tecnología	Propósito
Frontend	React 18 + Vite	SPA rápida, HMR
Estado	React Hooks (useState, useEffect, useRef)	Sin Redux, ligero
Estilos	TailwindCSS 3.4	Utility-first, modo responsive
Gráficos	Recharts	Área y línea, animación desactivada
Iconos	Lucide React	Consistentes y SVG puros
Mapa	Google Maps Embed	Solo vista, sin API key necesaria
Build	Vite + ESBuild	< 2 s en desarrollo
Deploy	Netlify / GitHub Pages	Drag-&-drop de carpeta dist

Estructura de Archivos

```
src/
  └── components/
    ├── HomePage.jsx
    ├── Dashboard.jsx
    ├── RoadVisualizer.jsx
    └── shared/
      ├── Button.jsx
      ├── Tooltip.jsx (por si quieres extraer)
      └── Icons.js (exportar icons de lucide)
  └── simulation/
    ├── constants.js
    ├── engine.js
    └── utils/
      ├── math.js (lerp, clamp, randGauss)
      └── formatters.js (formatTime, formatMoney)
  └── types/
    └── Vehicle.ts (opcional JS Doc)
```

```

    └── Cargo.ts
  hooks/
    └── useSimulationEngine.js (symlink a simulation/engine.js)
  styles/
    └── index.css (Tailwind base)
  tests/
    ├── engine.test.js (vitest)
    └── Dashboard.test.jsx (React Testing Library)

```

Motor de Simulación

```

intervalId = setInterval(() => {
  clockTick += timeScale;
  if (clockTick >= 4) { advanceSimClock(); clockTick = 0; }

  updateTrafficPhase();          // máquina de estados
  spawnVehicles(density);       // Poisson discreto
  vehicles = updatePhysics(vehicles);
  stats = accumulateCosts(vehicles);
  cleanupOffscreenAndExpired();
}, 50);

```

Física Simplificada

- Aceleración: +2 px/tick hasta v_max
- Frenado: -4 px/tick (emergency -8)
- Distancia de seguridad: SAFE_DISTANCE + speed * 0.5
- Overtake: requiere gapBack > 25 px && gapFront > 25 px
- Estrés mecánico: +1 por arranque en pendiente > 50x → falla aleatoria

Probabilidades de Eventos

Evento	Fórmula	Valor base
Accidente base	BASE_CHANCE = 0.00006	0.006 %
Lluvia	+ rain * 0.000012	+0.12 % max
Adelantamiento	+ 0.05 si frontal	5 %
Falla mecánica	stress > 50 && rand < 0.001	0.1 %

Modelo Económico

Ecuaciones

$$C_{\text{total}} = C_{\text{tiempo}} + C_{\text{combustible}} + C_{\text{carga}} + C_{\text{productividad}} + C_{\text{comercio}}$$

$$C_{\text{tiempo}} = 80\ 820 \text{ personas} \times (T/60) \text{ h} \times Q 51/\text{h}$$

$$C_{\text{combustible}} = 18\ 000 \text{ veh} \times 43 \text{ km} \times Q 4.20/\text{km}$$

$$C_{\text{carga}} = 981 \text{ camiones} \times (T/60) \text{ h} \times Q 400/\text{h} \times \text{Factor}$$

$$C_{\text{productividad}} = 40\ 000 \text{ empleados} \times (T/60) \text{ h} \times Q 150/\text{h} \times \text{Factor}$$

$$C_{\text{comercio}} = Q 120\ 000\text{--}150\ 000/\text{día} \text{ (estimado regional)}$$

Factores de Escenario

Escenario	Factor	Tiempo base	Costo/día
Optimista	1.0	72 min	Q 15.8 M
Real	1.5	90 min	Q 21.1 M
Caos	2.0	126 min	Q 32.3 M

Calibración y Validación

Fuentes de Datos Reales

- COVIAL: TPDA 2024 = 18 000 veh/día
- PROVIAL: Tiempo real de despeje = 3–5 min
- INEA: Costo oportunidad conductor = Q 51/h
- Camara de Comercio: Valor carga Almolonga ≈ Q 15 000/camión
- TPDA por tipo: 54 % livianos, 16 % pickups, 10 % buses, 8 % camiones, 12 % motos

Ajustes Finales

- Ajuste Wiedemann 99: LookAhead = 120 m (equivalente a SIGHT_DISTANCE = 300 px)
- Validación visual: cola máxima observada 2.8 km → simulada 2.9 km
- Error porcentual < 5 % en costo acumulado a 24 h

Rendimiento y Optimización

Benchmarks (laptop i5-1135G7, 16 GB)

Métrica	Valor
Vehículos simultáneos	250 sin lag
CPU React	4 % en 1x, 11 % en 2x
Memoria heap	42 MB promedio
FPS	50 estables (20 ms/frame)

Seguridad y Privacidad

- Sin cookies
- Sin tracking
- Sin API key expuesta (Google Maps Embed libre)
- Validación de rangos en todos los inputs (min=0, max=10, step=1)
- Escape de texto en Tooltips (innerText, no innerHTML)

Roadmap Técnico

- [] Migrar a TypeScript (strict)
- [] SWR o React Query para guardar récords en backend
- [] WebGL para >500 vehículos (PixiJS)
- [] Web Workers para física en hilo separado
- [] PWA: offline, instalable, push de alertas
- [] Tests visuales con Cypress (comparación de capturas)

Contacto y Repositorio

- Autor: Byron Fernando Torres Ajxup
- Correo: ajxupbyron@gmail.com
- GitHub: <https://github.com/tu-usuario/simulador-km194>
- Demo en vivo: <https://tu-usuario.github.io/simulador-km194>