

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**Facultad de Ciencias Naturales y Matemáticas**

Optimización de Rutas de Venta Directa de una Empresa de Alimentos en

Guayaquil empleando Algoritmos Metaheurísticos

**Proyecto Integrador**

Previo la obtención del Título de:

**Ingeniero en Logística y Transporte**

Presentado por:

Erick Renato Garzón Chávez

Dana Nicole Soledispa Núñez

Guayaquil - Ecuador

Año: 2023

## DEDICATORIA

---

Este proyecto está dedicado a mis padres, quienes me han brindado su apoyo incondicional a lo largo de mi vida estudiantil; a mis hermanas, por confiar en mí y en mis capacidades; y a mis abuelitos, quienes siempre me demostraron amor y estuvieron pendientes de mí.

*Dana Nicole Soledispa Núñez.*

Dedico este trabajo a mis queridos padres, abuelos maternos y paternos, cuyo amor y apoyo han sido mi inspiración. En especial, honro la memoria de mi abuelo, Jaime Garzón, y mi tío, Joselo Chávez, quienes, aunque ya no están físicamente, siguen siendo una fuente eterna de motivación y sabiduría en mi vida.

*Erick Renato Garzón Chávez.*

## AGRADECIMIENTOS

---

En primer lugar, agradezco a Dios por permitirme culminar otra etapa en mi vida, por sus infinitas bendiciones y su guía. También quiero expresar mi profundo agradecimiento a mi familia, quienes a lo largo de mi carrera universitaria se preocuparon por mi bienestar y me brindaron su apoyo incondicional.

Quiero extender mi gratitud a aquellos amigos que la universidad me ayudó a conseguir. En especial, quiero dedicar un especial agradecimiento a Erick. Más que un compañero de tesis se convirtió en un gran amigo que la universidad me regala.

No puedo dejar de expresar mi más profundo agradecimiento a William, mi compañero y fiel apoyo, quien ha estado a mi lado en cada paso de esta última etapa académica. Su amor, aliento y apoyo han sido fortaleza para alcanzar mis metas.

*Dana Nicole Soledispa Núñez.*

Quiero expresar mi profundo agradecimiento a Dios por su guía divina que ha iluminado mi trayectoria. A mis padres, quienes han sido el cimiento sólido de mis logros, les dedico un agradecimiento especial por su inquebrantable apoyo y sacrificios.

A mi familia y amigos cercanos, les agradezco sinceramente por cada aporte valioso a mi éxito. Además, quiero reconocer a mi compañera de tesis, Dana, por su dedicación y amistad inquebrantable. Juntos, hemos conformado un equipo excepcional. Su contribución ha sido invaluable en este viaje académico.


A todos, gracias por ser parte esencial de mi camino.

*Erick Renato Garzón Chávez.*

## DECLARACIÓN EXPRESA

---

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; Dana Nicole Soledispa Núñez y Erick Renato Garzón Chávez damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”



Dana Nicole Soledispa Núñez



Erick Renato Garzón Chávez

## **EVALUADORES**

---

---

**Msc. Carlos Alfredo Ronquillo**

**Franco**

Profesor de Materia

---

**Mat. Fernando Francisco Sandoya**

**Sánchez**

Tutor de proyecto

## RESUMEN

Este proyecto aborda la problemática de determinar rutas óptimas de ventas directas para minimizar la distancia total recorrida y reducir las emisiones de  $CO_2$  en una empresa de la industria alimentaria. Con una cartera de 9.539 clientes y 53 rutas mal estructuradas desde el Centro de Distribución de Guayaquil, se enfrentaban inconvenientes, como visitas incompletas y recursos subutilizados. El objetivo general fue optimizar las rutas mediante un modelo matemático mTSP, complementado con el método de machine learning “K-Medoids” y un modelo de clusterización, junto con el algoritmo metaheurístico GRASP. La sinergia de estos procedimientos más la capacidad de manejo de datos de Wolfram Mathematica proporcionó una estrategia eficiente para la distribución de rutas. Los resultados resaltaron una disminución en la flota de camiones y mejoras en las rutas, lo que contribuyó a la optimización de la logística y a un aumento en la satisfacción del cliente. Además, se logró una notable reducción de costos totales y de la huella de carbono. En conclusión, se logró optimizar las rutas de la empresa y se demostró el impacto positivo de los algoritmos metaheurísticos en la gestión logística.

**Palabras Clave:** mTSP, Clusterización, algoritmo, emisiones de  $CO_2$ , eficiencia operativa.

## ABSTRACT

*This project addresses the challenge of determining optimal direct sales routes to minimize total distance traveled and reduce CO<sub>2</sub> emissions in a company in the food industry. With a portfolio of 9,539 clients and 53 poorly structured routes from the Guayaquil Distribution Center, issues such as incomplete visits and underutilized resources were encountered. The overall objective was to optimize routes using an mTSP mathematical model, complemented by the "K-Medoids" machine learning method and a clustering model, along with the GRASP metaheuristic algorithm. The synergy of these procedures, combined with the data handling capabilities of Wolfram Mathematica, provided an efficient strategy for route distribution. The results highlighted a reduction in the truck fleet and improvements in routes, contributing to logistics optimization and increased customer satisfaction. Furthermore, a significant reduction in total costs and carbon footprint was achieved. All in all, the company's routes were successfully optimized, demonstrating the positive impact of metaheuristic algorithms on logistics management.*

**Keywords:** *mTSP, Clustering, algorithm, CO<sub>2</sub> emissions, operational efficiency.*



## ÍNDICE GENERAL

CAPÍTULO 1 .....	15
1. INTRODUCCIÓN .....	15
1.1 Descripción del Problema .....	16
1.2 Justificación del Problema .....	18
1.3 Objetivos .....	19
1.3.1 <i>Objetivo General</i> .....	19
1.3.2 <i>Objetivos Específicos</i> .....	19
1.4 Marco Teórico .....	20
1.4.1 <i>Antecedentes</i> .....	20
1.4.2 <i>Estado del Arte</i> .....	20
1.4.3 <i>Marco Conceptual</i> .....	23
CAPÍTULO 2 .....	26
2. METODOLOGÍA .....	26
2.1 Técnicas De Investigación .....	26
2.1.1 <i>Levantamiento De Información</i> .....	27
2.1.2 <i>Recopilación De Los Datos</i> .....	29
2.2 Análisis De La Información Levantada: Situación Actual .....	30
2.3 Modelos Matemático.....	32
2.3.1 <i>Modelo mTSP</i> .....	32
2.3.2 <i>Modelo de Clusterización</i> .....	35
2.4 Uso De Software .....	37
2.5 Consideraciones Legales Y Éticas .....	38
2.6 Fases Del Proyecto .....	40
2.7 Cronograma De Trabajo.....	41
CAPÍTULO 3 .....	43
3. RESULTADOS Y ANÁLISIS.....	43

3.1 Clusterización.....	43
3.2 Asignación de Rutas.....	46
3.3 Análisis de Costo de la Propuesta .....	49
3.4 Comparativa entre Situación Actual y Propuesta de Mejora .....	50
3.4.1 Distancia Total Recorrida .....	50
3.4.2 Impacto Medioambiental.....	50
3.4.3 Ahorro en costos.....	51
3.5 Entregables .....	53
CAPÍTULO 4 .....	54
4. CONCLUSIONES Y RECOMENDACIONES.....	54
4.1 Conclusiones .....	54
4.2 Recomendaciones.....	55
BIBLIOGRAFÍA.....	57
APÉNDICES .....	59
Apéndice A.....	59
Apéndice B .....	75
Apéndice C.....	91
Apéndice D.....	110
Apéndice E .....	117

## **ABREVIATURAS**

ESPOL Escuela Superior Politécnica del Litoral

ODS Objetivo de Desarrollo Sostenible

mTSP Multiple – Traveling Salesman Problem

CD Centro de Distribución

## SIMBOLOGÍA

km	Kilómetro
tn	Tonelada
$CO_2$	Dióxido de carbono
$NO_x$	Óxidos de nitrógeno

## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Mapa de actores .....	27
<b>Figura 2.</b> Mapa de empatía.....	28
<b>Figura 3.</b> Mapa térmico de los clientes .....	30
<b>Figura 4.</b> Diagrama de barras de clientes visitados por ruta .....	31
<b>Figura 5.</b> Fases del proyecto.....	40
<b>Figura 6.</b> Cronograma de trabajo .....	42
<b>Figura 7.</b> Clientes de la ruta 1 .....	45
<b>Figura 8.</b> Clientes de la ruta 50 .....	45
<b>Figura 9.</b> Recorrido de la ruta 1 – lunes (2) .....	47
<b>Figura 10.</b> Recorrido de la ruta 50 – lunes (2) .....	48
<b>Figura 11.</b> Comparación de costos anuales .....	52
<b>Figura 12.</b> Movimiento de dinero para los 3 primeros años .....	53

## ÍNDICE DE TABLAS

<b>Tabla 1.</b> Rutina de chofer en rutas de ventas directas .....	17
<b>Tabla 2.</b> Datos en bruto .....	29
<b>Tabla 3.</b> Softwares utilizados: ¿qué son?, ventajas y desventajas.....	37
<b>Tabla 4.</b> Clientes por ruta .....	43
<b>Tabla 5.</b> Secuencia de visita por ruta.....	46
<b>Tabla 6.</b> Costo de la propuesta .....	49
<b>Tabla 7.</b> Comparación de distancia total recorrida.....	50
<b>Tabla 8.</b> Comparación de emisiones de <i>CO2</i> .....	50
<b>Tabla 9.</b> Comparación de emisiones de <i>NOx</i> .....	51
<b>Tabla 10.</b> Comparación de costos.....	51
<b>Tabla 11.</b> Liquidación del personal .....	52

# CAPÍTULO 1

## 1. INTRODUCCIÓN

La industria alimentaria y de distribución exige eficiencia en la gestión y la planificación de rutas de ventas a sus clientes dada la alta competitividad del mercado. La empresa sujeta de estudio tiene presencia a nivel mundial y está consolidada en el mercado ecuatoriano desde hace varios años. Esta empresa maneja una amplia gama de productos, entre los cuales están alimentos para bebés, lácteos, chocolates y café. Dado su fuerte crecimiento en el mercado nacional, la empresa cuenta con flota propia de vehículos de carga para manejar sus ventas directas con tiendas minoristas. No obstante, las rutas actuales no logran alcanzar los objetivos de ventas, originando clientes sin cobertura, por lo que se requiere optimizarlas.

El hecho de que no se han diseñado rutas óptimas de venta ocasiona algunos efectos negativos: insatisfacción en los clientes, menor sostenibilidad ambiental e ineficiencia de la operación. El problema de optimización de rutas es un gran desafío para las empresas distribuidoras, ya que deben considerar diversos factores al momento de buscar la optimización del ruteo. Algunos aspectos claves para tener en cuenta son: demanda, capacidad de los vehículos, distancia, tiempo de viaje y otras reglas de operación.

La planificación del sistema de transporte involucra no solo a la empresa alimentaria, sino también a sus equipos de ventas, transportistas y consumidores finales. Este proyecto de graduación se centra en abordar los desafíos y oportunidades en la optimización de rutas de venta directa a través de un modelo matemático de optimización y la implementación de un algoritmo basado en metaheurísticas para resolver adecuadamente este problema de negocio y contribuir significativamente a la mejora de la eficiencia operativa, la satisfacción del cliente y al impacto medioambiental.

## 1.1 Descripción del Problema

La problemática que pretende solucionar este proyecto es la determinación de rutas óptimas de ventas directas con el objetivo de minimizar la distancia total recorrida, y por consiguiente disminuir las emisiones de CO<sub>2</sub> hacia el medioambiente.

Hoy en día, la empresa cuenta una cartera de 9,539 clientes, internos y foráneos, cuyas visitas son cubiertas por 53 rutas que parten desde el Centro de Distribución de Guayaquil. Estas rutas no están correctamente estructuradas, puesto que algunas cuentan con menos de 50 clientes mientras que otras superan los 250. El sobrecargo de clientes en una ruta causa que no todos los clientes que se planificaron en la misma sean visitados, generando así inconvenientes e inconformidades por parte de los consumidores por lo que al día siguiente solicitan una visita por parte del chofer. Esto a su vez causa que el chofer se desvíe de su ruta programada con la finalidad de atender al cliente que no pudo visitar el día anterior, creando así una cadena de inconvenientes por los clientes que no logran ser atendidos. A su vez, las rutas que cuentan con pocos clientes ocasionan que los recursos involucrados, personal y vehículos asociados, estén siendo subutilizados. Por lo tanto, lo que la empresa necesita es que las nuevas rutas cuenten con 32 clientes dentro de la cartera de clientes diaria, es decir 192 clientes a la semana.

La empresa opera con una flota homogénea de 53 vehículos, con una capacidad de carga de 1.1 toneladas cada uno. Cada ruta tiene asignado su respectivo vehículo, el cual es operado por un conductor designado. Esta persona es la única que se moviliza de punto a punto para realizar las ventas directas a los clientes.

La rutina de un chofer en ruta se la puede observar en la Tabla 1.



**Tabla 1.** Rutina de chofer en rutas de ventas directas

<b>Hora</b>	<b>Actividades</b>
<b>06h30</b>	Llegada al CD  Inventario  Revisión de indicadores con Supervisor
<b>07h30</b>	Traslado al primer punto de venta
<b>08h10</b>	Llegada al primer punto de venta
<b>13h20</b>	Fin primer turno (15 clientes visitados)  Inicio Almuerzo
<b>14h20</b>	Fin almuerzo  Llegada al punto de venta
<b>18h20</b>	Fin segundo turno (11 clientes visitados)  Traslado al CD
<b>19h00</b>	Llegada al CD  Cierre de jornada, entrega de liquidación, depósitos y notas de crédito al Supervisor
<b>19h30</b>	Salida del CD

Fuente: Empresa de Alimentos

En la Tabla 1 se puede observar que el primer turno del chofer tiene una duración de 5 horas y 10 minutos, con lo cual se identifica que, en promedio, el tiempo de visita por cliente es de 22 minutos. Dentro de estos 22 minutos, el chofer debe buscar parqueo, identificar las necesidades del cliente (proceso de negociación), cerrar la venta, cobrar, y finalmente despachar y entregar el pedido.

El alcance de este proyecto se centra en diseñar rutas óptimas que logren cubrir los 9,539 clientes que son visitados desde el centro de distribución de Guayaquil. Los datos que se requieren

para el desarrollo de este proyecto son los siguientes: posiciones geográficas de los clientes, posición geográfica del centro de distribución, tiempo promedio de visita por cliente, número mínimo de clientes por día, cantidad de días laborales y rutas actuales que se encuentran operando. Con esta información, se puede construir un modelo matemático que pueda ser resuelto con técnicas metaheurísticas y que permita encontrar las rutas óptimas que garanticen una operación eficiente.

## **1.2 Justificación del Problema**

La presente investigación tiene como finalidad abordar un problema de relevancia crítica en el contexto de una empresa alimentaria en Guayaquil: la optimización de rutas de ventas directas.

Un aspecto que justifica esta investigación es su contribución al cuidado del medio ambiente. Al optimizar las rutas con respecto a la minimización de distancia total recorrida, cuantificada en kilómetros, se reducirán las emisiones de CO<sub>2</sub> lo que respalda los esfuerzos de sostenibilidad y responsabilidad ambiental de la empresa.

Este proyecto apunta a promover el Objetivo de Desarrollo Sostenible (ODS) número 12, "Producción y Consumo Responsables", dado que el trabajo se basa en la creciente importancia de abordar la sostenibilidad en el ámbito empresarial y la optimización de recursos en la cadena de suministro de alimentos y bebidas. El sector de alimentos y bebidas enfrenta desafíos significativos en términos de eficiencia, desperdicio y sostenibilidad. El consumo insostenible y la producción ineficiente de alimentos tienen un impacto negativo en el medio ambiente, la economía y la sociedad. Por lo tanto, es esencial abordar estos problemas a través de enfoques innovadores.

Asimismo, este trabajo busca mejorar la cobertura de la cartera de clientes de cada chofer. Los clientes finales serán beneficiarios directos, ya que contarán con visitas regulares y planificadas, lo que les permitirá anticipar y gestionar sus pedidos de manera eficiente, mejorando la experiencia del cliente. Los choferes, por su parte, gozarán de rutas más eficientes y podrán

atender a la totalidad de su cartera de clientes de manera más efectiva, lo que puede aumentar su satisfacción laboral. La empresa misma se beneficiará significativamente. La optimización de rutas permitirá alcanzar los objetivos de ventas pactados de manera más eficiente y rentable, al tiempo que reducirá la flota necesaria y el consumo de combustible.

La investigación no solo ayudará a resolver un problema específico en la empresa, sino que también tiene un potencial de aplicabilidad más amplio. El algoritmo desarrollado en este proyecto podría ser utilizado por otras empresas con características similares que enfrentan desafíos relacionados con la optimización de rutas y el cumplimiento de visitas a clientes.

Este proyecto no solo se alinea con los valores de la empresa, sino que también responde a las demandas cambiantes del mercado y la creciente preocupación por la sostenibilidad.

## **1.3 Objetivos**

### ***1.3.1 Objetivo General***

Optimizar las rutas utilizadas para la venta directa en una empresa de la industria alimentaria, considerando como objetivo la minimización de la distancia total recorrida mediante la implementación de algoritmos metaheurísticos para mejorar la eficiencia de la operación.

### ***1.3.2 Objetivos Específicos***

- Analizar los datos proporcionados con herramientas tecnológicas para así garantizar su calidad y coherencia.
- Formular el modelo matemático de optimización identificando todas las restricciones relevantes y reglas de distribución de la empresa para resolver el problema de estudio.
- Implementar los algoritmos metaheurísticos en un lenguaje de programación científico que permita encontrar soluciones de buena calidad y en tiempos aceptables.
- Medir el beneficio que la empresa obtendría con la propuesta de mejora realizando un análisis comparativo con las rutas actuales.

## **1.4 Marco Teórico**

### ***1.4.1 Antecedentes***

La optimización de rutas en el sector de distribución y logística de empresas alimentarias es un desafío crítico que ha cobrado una creciente importancia en los últimos años. De acuerdo con (Ballou, 1991), la función del sistema de transporte se basa en la selección de los medios idóneos para cada ruta, teniendo que definirlos con relación a costes fiabilidad y riesgos. Por su parte la ciudad de Guayaquil, Ecuador, como un centro importante para la producción y distribución de alimentos, enfrenta desafíos significativos en términos de eficiencia logística, costos operativos y satisfacción del cliente. La entrega oportuna de alimentos perecederos y no perecederos es esencial para garantizar la frescura de los productos y la competitividad en el mercado.

Históricamente, la planificación de rutas ha sido un proceso manual, según (Murray, 2018) la planeación de rutas ha sido empleada por compañías por varios años, y a pesar de experiencia de los planeadores, el sistema cambiante y complejo actual afecta la validez de una ruta día a día. Sin embargo, en las últimas décadas, los avances en la tecnología de la información y la investigación en algoritmos de optimización han brindado nuevas oportunidades para abordar estos desafíos.

### ***1.4.2 Estado del Arte***

La optimización de rutas es un campo ampliamente estudiado en la literatura académica y ha sido aplicado con éxito en diversas industrias, por lo que a continuación se presenta 3 fuentes de información que complementan el desarrollo de este trabajo.

Como primera fuente de consulta se presenta el artículo titulado *Solución del problema de múltiples agentes viajeros resuelto mediante técnicas heurísticas*. Éste menciona que las exigencias de los clientes con respecto a productos y servicios se intensifican con el paso del tiempo; los consumidores demandan menores tiempos de visita y/o entrega (Ocampo, Bolaños, &

Echeverri, 2014) Es por esto por lo que es indispensable contar con un sistema de transporte óptimo que garantice el suministro pertinente de insumos, así como la entrega en el tiempo y lugar preciso. La metodología que se maneja en este artículo es la heurística de dos fases: asignar primero, rutear después. Esta se basa en, primero, establecer los clústeres, es decir grupos de clientes. Luego, se rutea dentro de cada clúster y se aplica una mejora al tour encontrado. El método para definir los clústeres fue la heurística de barrido, la cual se encarga de formar grupos mediante el giro de una semirrecta con origen en el depósito y se detiene cuando se sobrepase la restricción dada. El problema que se resolvió en esta investigación fue la visita de 75 ciudades por medio de la heurística del vecino más cercano. Las secuencias obtenidas fueron mejoradas con estrategias de búsqueda local, tales como shift, swap y 2-opt. En términos generales, todas las estrategias mejoraron la solución inicial significativamente; no obstante, la que tuvo un mejor desempeño fue la shift con parámetros 1 y 0.

El aporte de este artículo científico al proyecto se centra en las políticas de rutas y en la metodología empleada. El presente proyecto maneja una política fija, ya que los clientes deben conocer con antelación el día de visita para poder planificar su pedido. Por otro lado, la metodología de heurísticas de dos fases con estrategias de mejoras es aplicable a esta problemática debido a que se debe cumplir con un número de clientes en cada portafolio de chofer, por lo que se debe crear los clústeres y luego rutear dentro de cada uno, sin olvidar aplicar mejoras al tour obtenido.

*“Comparison of K-means and K-medoids in Tourist Attraction Clustering based on visitor Characteristics”* es el segundo artículo que se consultó como fuente de investigación. A lo largo de este artículo se manejan la terminología de K-Means y K-Medoids, los cuales son métodos de particionamiento para formar clústeres. El análisis de división empieza creando un clúster con todos los objetos, luego se separan los objetos con la mayor diferencia del clúster. Así sucesivamente hasta que todos los objetos separados se convierten en un nuevo clúster. Por su

parte, el de aglomeración combina dos objetos con similitudes en un clúster, así hasta formar un clúster en donde todos los objetos se relacionen unos con otros. Cabe destacar que el parámetro que se emplea para determinar si son similares es la distancia euclidiana (Jauhari, et al., 2023).

La data que se trató para la comparación de los métodos K-Means y K-Medoids fue recopilada de 21 puntos turísticos del área de Bangkalan, tomando en cuenta el número de comida de cada turista. Los datos pasaron por un proceso de depuración y normalización para luego poder ser clusterizados; de esta manera, se puede identificar el nivel de influencia que las características (edad, género, estado civil, etc.) de los turistas tienen sobre los puntos más visitados en la zona. Los resultados de aplicar los dos métodos de agrupamiento arrojaron que K-Medoids tuvo un mejor desempeño que K-Means. La característica que resultó más dominante fue el género, con un porcentaje de 90.48% empleando el método K-Medoids.

Este artículo contribuye al presente proyecto integrador ya que se tratan de métodos para agrupar datos, siendo esto parte primordial para el establecimiento de rutas. Ya sea el método K-Means o K-Medoids el que se emplee, se debe evaluar no solo el resultado de los clústeres obtenidos sino también el tiempo computacional puesto que se trata de una cartera extensa de clientes.

Finalmente, la tercera fuente de investigación que se analiza es el artículo titulado “*An Improvement to the 2-Opt Heuristic Algorithm for Approximation of Optimal TSP Tour*”. Este artículo aborda el TSP, uno de los problemas NP-Hard con aplicaciones prácticas más significativas y que se adapta al contexto de la problemática del proyecto de materia integradora. La metodología que se propone es una mejora con el algoritmo heurístico llamado 2-Opt++, el cual resuelve problemas del TSP usando la técnica de búsqueda del 2-Opt para generar mejores resultados.

Esta herramienta brindaría muchos beneficios al proyecto en cuestión debido a que reduciría considerablemente el tiempo de cálculo computacional, teniendo en cuenta que se

trabajarán con más de 9,000 clientes. En el algoritmo 2-Opt++ se implementa un paso de compresión de gráfico opcional. Este enfoque se basa en la lista de candidatos, revertir los caminos seleccionados y luego probar una solución mejorada es un método simple, pero poderoso (Uddin, et al., 2023). Sin embargo, hay un problema en este método, y es que a medida que se buscan todas las soluciones surge la posibilidad de perder una solución mejor, y este punto se aborda en el algoritmo a través de criterios de posible aceptación, es decir, si el margen de ganancia en el paso anterior es mayor que 0, se mantiene la nueva solución.

Asimismo, al analizar y contrastar los resultados de margen de error con otros algoritmos en instancias predefinidas, el algoritmo 2-Opt++ es claramente superior y más efectivo que el resto, en términos de porcentaje de error comparado con el resultado óptimo. Basado en la eficacia de la heurística se plantea emplear métodos similares para resolver la problemática actual.

### ***1.4.3 Marco Conceptual***

Habiendo revisado literatura relacionada a la problemática a tratar, se presentan algunos conceptos esenciales que ayudan a la comprensión del desarrollo de este trabajo.

#### **Multiple – Traveling Salesman Problem**

El Multiple Traveling Salesman Problem, o mTSP es una variante muy relevante del problema clásico del agente viajero (TSP), donde se consideran múltiples agentes viajeros los cuales deben visitar un número de ciudades exactamente una vez y volver a la posición inicial con el mínimo costo de viaje (Cheikhrouhou & Khouf, 2021).

El mTSP es uno de problemas de optimización más importantes, ya que su aplicación en escenarios de la vida real es muy destacada. Para aplicar el mTSP se deben establecer la representación de los agentes viajeros; estos pueden ser aviones, drones o camiones (agente de este proyecto). De manera análoga, las ciudades visitadas por los agentes pueden ser representadas por objetivos en servicios militares, víctimas en emergencias o clientes.

Con respecto a la función objetivo, se pueden establecer diferentes escenarios tales como, minimizar el costo del tour, el tiempo de la misión, la energía consumida, o la distancia total recorrida, el cual es el escenario por tratar.

Por último, las restricciones desempeñan un papel esencial en la formulación ya que ayudan a garantizar la factibilidad y calidad de las soluciones considerando limitaciones operativas, tiempo, etc.

### **Algoritmo GRASP (Greedy Randomized Adaptive Search Procedures)**

Con el propósito de brindar soluciones óptimas a problemas de la industria, no solo se necesitan métodos heurísticos que sean moldeables a los requerimientos sino también de herramientas novedosas tales como procedimientos de reinicio y aleatorización controlada (Feo & Resende, 1995).

Las siglas GRASP corresponden a Greedy Randomized Adaptive Search Procedures, lo que en español se traduce a procedimientos voraces de búsqueda adaptativa aleatoria. Los autores Feo y Resende mencionan que GRASP es una heurística que se basa en un proceso iterativo y consiste en dos fases: construcción y mejora.

En la fase de construcción se arma una solución de manera iterativa basándose en una función glotona. Esta función se encarga de medir el beneficio de agregar cada uno de los elementos a la solución y de esta manera optar por el de mayor beneficio. El aspecto que caracteriza a esta heurística es su componente aleatorio, pues de una lista de candidatos restringidos (LCR) se escoge aleatoriamente un elemento. Dentro de la LCR se encuentran los mejores elementos a formar parte de la solución, por lo que esta lista se va actualizando en cada iteración (Sandoya, 2023).

En la fase de mejora se emplean algoritmos de búsqueda local que ayuden a perfeccionar la solución que se tiene. Un algoritmo de búsqueda local se encarga de reemplazar un elemento de la solución actual por uno mejor dentro de una vecindad definida, por lo tanto, el criterio de parada



está definido por la vecindad ya que, si no hay una mejor solución dentro de ésta, se detiene el algoritmo arrojando así la última solución obtenida (Sandoya, 2023).

La implementación de esta heurística requiere de dos parámetros claves, los cuales son el tamaño de la LCR y el número de iteraciones, es por esto por lo que Feo y Resende (1995) consideran que la sencillez y la facilidad de implementación son de las cualidades más importantes del GRASP.

## **CAPÍTULO 2**

### **2. METODOLOGÍA**

#### **2.1 Técnicas De Investigación**

En el desarrollo de este proyecto, se implementaron diversas técnicas de investigación con el fin de recopilar y analizar datos críticos. En primer lugar, se destaca la realización de entrevistas online a través de la plataforma Microsoft Teams, en donde se tuvo contacto con los colaboradores de la empresa alimentaria permitiendo así obtener información directa acerca del proceso de ventas.

Asimismo, la empresa proporcionó una base de datos de Excel, de la cual se obtendría información crucial sobre la localización geográfica de los clientes. Esta data se analizó para la comprensión de la distribución geográfica, facilitando la identificación de áreas con alto volumen de clientes.

Para garantizar la precisión y calidad de los datos, se llevó a cabo un exhaustivo proceso de depuración a través de Google Sheets, con el objetivo de identificar posibles errores y asegurar la fiabilidad de los resultados obtenidos. En particular, se eliminaron dos clientes ubicados en Galápagos, ya que no era factible considerar una ruta terrestre desde el centro de distribución en Guayaquil hasta esta ubicación insular. Esta decisión se tomó en concordancia con restricciones logísticas y de transporte. Además, se identificó que siete clientes en Durán carecían de coordenadas geográficas, lo que complicaba la integración efectiva en la optimización de rutas.

Estas metodologías combinadas permitieron obtener insights significativos que constituyeron la base para proponer las estrategias efectivas de optimización de rutas de venta directa en el contexto específico de Guayaquil.

### 2.1.1 Levantamiento De Información

En el proceso de levantamiento de información, se destaca la aplicación del proceso de empatía, que permitió obtener una visión integral de los actores clave en el sistema de transporte.

**Figura 1.** Mapa de actores



Una de las consideraciones más significativas fue la creación de un mapa de actores englobando a figuras fundamentales del servicio tal como se observa en la *Figura 1*. Esto permitió identificar y tener una idea gráfica de las partes relevantes del sistema tales como la jefa de transporte, analistas de transporte, conductores y clientes, quienes fueron los que se posicionaron en el centro del mapa dado que son el ente más importante y deben ser satisfechos con el servicio que se provee.

Además, se logró comprender las relaciones latentes en el sistema, dado que se pudo observar gráficamente cuál es la cadena de interacciones del sistema, quién influye sobre quién, y cómo se comunican entre sí. De manera similar, se rescata que el mapa de actores brindó una clara idea de cómo podría planificarse la implementación de las nuevas rutas propuestas y cómo participarían los actores del sistema.

**Figura 2. Mapa de empatía**



De forma similar, se realizaron mapas de empatía para comprender las necesidades de cada parte. La *Figura 2* se centró en la jefa de transporte con el propósito de identificar sus motivaciones, desafíos y objetivos individuales.

Entre los puntos más relevantes se destaca la visión de la jefa de transporte, a quien se reconoció que estaba consciente de que las rutas actuales no eran óptimas, y estaban muy lejos de serlo. Se comentó haber identificado oportunidades de mejora a las rutas de transporte aplicando una mejor clusterización empleando algoritmos que consideren las restricciones del escenario existente. Otro detalle que aportó en gran medida a la realización de este proyecto fue el razonamiento lógico que establece que, al recorrer más kilómetros, no se asegura que se realizan más ventas.

El análisis proporcionó una base sólida para el diseño del proceso de optimización que no solo maximiza la eficiencia en el sistema de transporte, sino que también considera las dinámicas interpersonales y las metas particulares de quienes desempeñan roles claves en la gestión logística.

### 2.1.2 Recopilación De Los Datos

**Tabla 2.** Datos en bruto

Num. Cliente	Rutas	Días	CustomerCity	CusLatitude	CusLongitude
1	EC9201	4	Guayaquil	-2.079554	-79.853561
2	EC9201	4	Guayaquil	-2.078533	-79.852824
3	EC9201	4	Guayaquil	-2.148299	-79.868375
4	EC9201	4	Guayaquil	-2.149788	-79.868097
5	EC9201	4	Guayaquil	-2.078526	-79.852818
6	EC9201	6	Guayaquil	-2.054763	-79.881871
7	EC9201	2	Guayaquil	-2.033584	-79.908674
8	EC9201	6	Guayaquil	-2.053525	-79.879864
9	EC9201	6	Guayaquil	-2.054252	-79.88178
10	EC9201	6	Guayaquil	-2.061016	-79.872625

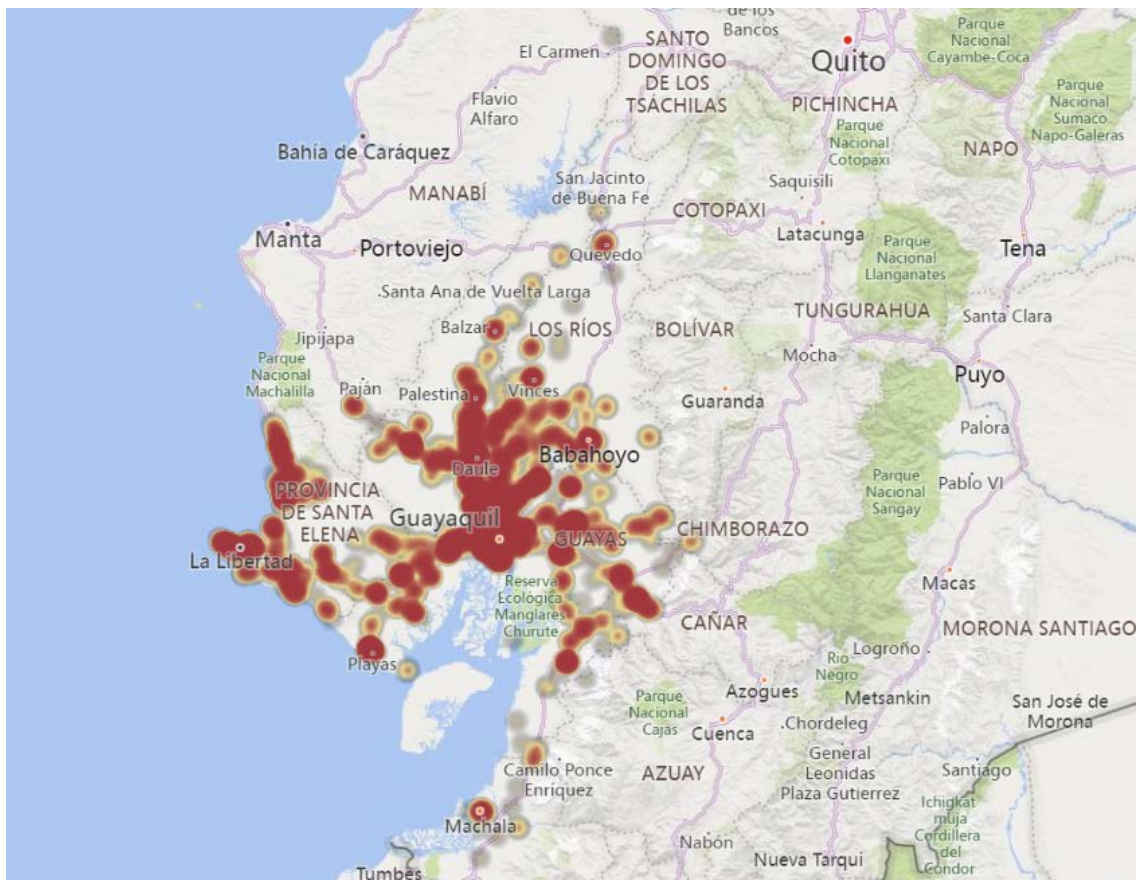
Fuente: Empresa de Alimentos

La *Tabla 2* alberga información esencial para la optimización de rutas de venta directa. Cada fila representa un cliente, en este apartado se colocaron las primeras 10 filas de la tabla. En primer lugar, se encuentra la columna de la identificación por número del cliente. La siguiente columna es la de rutas que las identifica por códigos que comienzan con "EC" junto con un número asociado a la ruta. Seguido a esto, está la columna Días. En esta se encuentran los números asociados al día de la semana, empezando por el domingo con 1 y finalizando el sábado con 7. Mientras que en la columna CustomerCity está la ciudad a la que pertenece el cliente, y por último se encuentran las ubicaciones que se detallan mediante coordenadas de latitud y longitud. Este conjunto de datos ofrece una visión detallada de la distribución espacial de los clientes, proporcionando la base para abordar la eficiencia de las rutas.

## 2.2 Análisis De La Información Levantada: Situación Actual

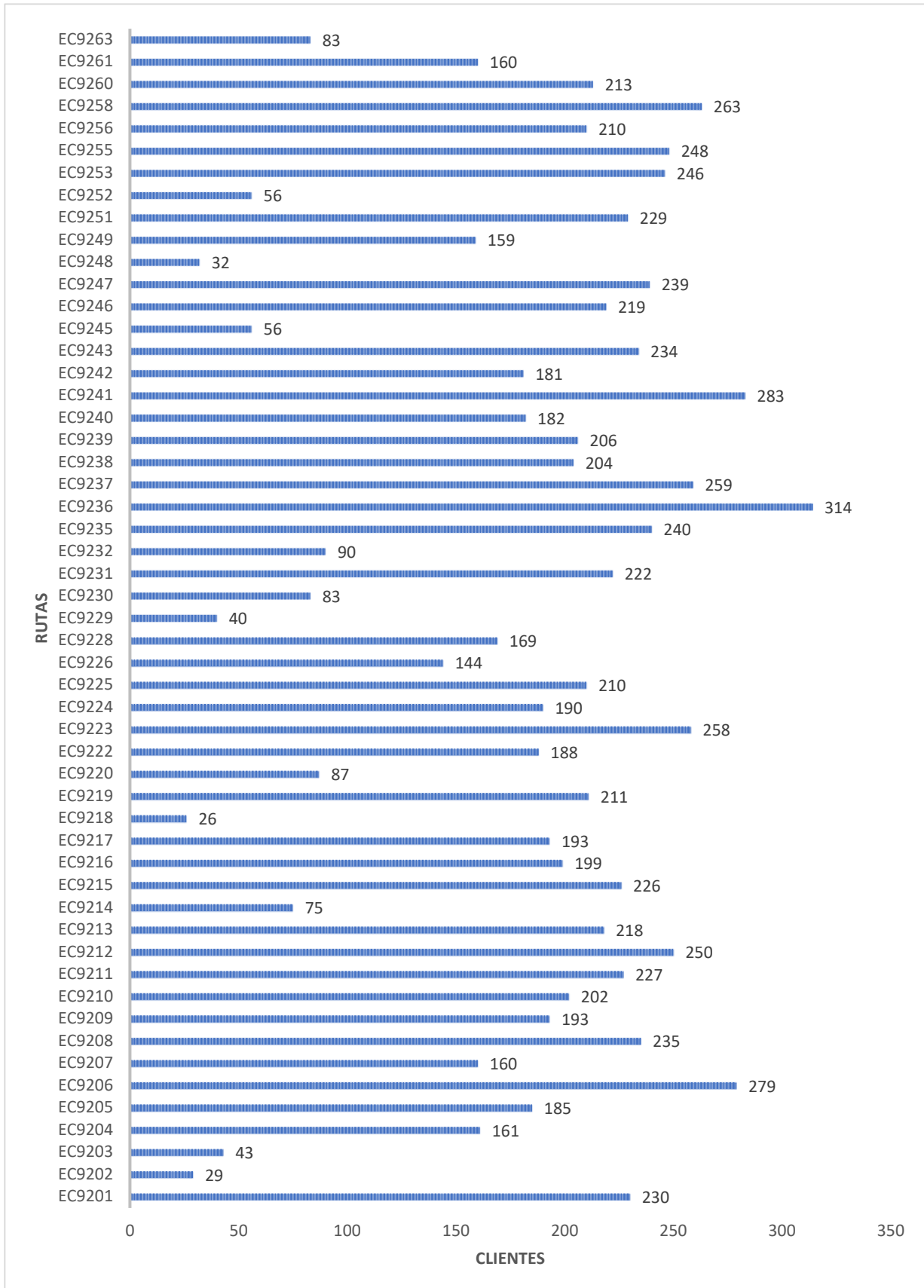
Como parte del análisis de la data recopilada, se realizó un mapa de calor, con el fin de identificar las zonas que más clientes tienen, y un diagrama de barras de los clientes por rutas, para tener una visión más clara de la repartición actual de clientes.

**Figura 3.** Mapa térmico de los clientes



En la *Figura 3* se puede observar las zonas con mayores concentraciones de clientes, siendo estas identificadas con color rojo. Guayaquil, Babahoyo y La Libertad son de las ciudades con un elevado volumen de clientes. Con color amarillo se puede observar las zonas que albergan una cantidad moderada de clientes, mientras que las zonas grises representan una baja concentración de clientes.

**Figura 4.** Diagrama de barras de clientes visitados por ruta



En la *Figura 4* se logra identificar cuantos clientes son visitados por cada ruta durante la semana. El objetivo de la empresa de alimentos es visitar semanalmente 300 clientes. Esta grafica deja en evidencia que no se está cumpliendo con aquello, pues hay rutas que están muy por debajo del objetivo, como la rutas EC9218 y EC9202. Por otro lado, se evidencia una ruta que supera los 300 clientes, la cual es EC9236 con 314 clientes.

## **2.3 Modelos Matemático**

### **2.3.1 Modelo *mTSP***

Los modelos matemáticos que se consideraron formular para este problema de estudio fueron: el Vehicle Routing Problem (VRP), el Travelling Salesman Problem (TSP) y el *mTSP*. Sin embargo, se descartó el VRP por su complejidad computacional. Resolver el VRP conlleva más trabajo computacional debido al número de restricciones por implementar. En cuanto al TSP, se descartó este modelo debido a su limitación de un único agente viajero. Además, el TSP no es muy representativo en aplicaciones reales, pues en el campo de la distribución es común contar con múltiples vehículos. En cambio, el *mTSP* logra hacer partícipe a varios vehículos, siendo así este último el modelo matemático elegido para este problema de ruteo.

Se optó por este modelo pues aborda los siguientes puntos (Granada Echeverri, Cubides Rivera, & Bolaños, 2022):

- La distancia total recorrida de todas las rutas debe ser minimizada.
- Todas las rutas deben comenzar y culminar en el depósito.
- Cada cliente debe ser visitado una sola vez por una ruta.
- Existe un número  $M$  de agentes viajeros.
- Cada agente viajero visita un número máximo de clientes  $P$ .

La formulación del modelo matemático comienza con la definición de los conjuntos. Se definió el conjunto  $V$  de nodos que representan los clientes. Después, se definió al depósito, es



decir al centro de distribución (CEDI), como  $R$  y pertenece también a  $V$ . A su vez, se definió el conjunto  $E$  para denotar que tanto  $i$  como  $j$  pertenecen a  $V$ .

Con respecto a los parámetros, se trabajó con la matriz  $d_{i,j}$  que muestra la distancia entre los clientes, incluyendo el depósito. El valor de  $M$  corresponde al número de conductores o vehículos disponibles, en este caso sería 53.  $P$  hace referencia al número máximo de clientes a visitar por ruta, por lo que el valor que se atribuye es 192.

En cuanto a las variables, se planteó una binaria y otra entera,  $x_{i,j}$  y  $u_i$  respectivamente. La primera representa el paso por el arco  $ij$  y la segunda la secuencia de visita al cliente  $i$ .

### Conjuntos:

$$V = \{1,2,3, \dots, 9539\}$$

$$R \in V \therefore V = \{0,1,2,3, \dots, 9539\}$$

$$E \in \{i \in V, j \in V\}$$

### Parámetros

$d_{i,j}$  = matriz de distancia.

$M$  = número de conductores o vehículos disponibles.

$P$  = número máximo de clientes a visitar por ruta.

### Variables de decisión

$x_{i,j}$  = variable de decisión binaria que representa el paso por arco  $ij$ . Toma el valor de 1 si se visita al cliente  $i$  y luego se visita al cliente  $j$ , 0 caso contrario.

$u_i$  = variable de decisión entera que representa la secuencia de visita al cliente  $i$ .

### Función objetivo

$$\min \sum_{e \in E} d_e * x_e \quad (2.1)$$

## Restricciones

$$\sum_{e \in \delta(i)} x_e = 2 \quad \forall i \in V \quad (2.2)$$

$$\sum_{e \in E} x_e = |V| - M - 1 \quad (2.3)$$

$$\sum_{j \in V, j \neq R} x_{Rj} = M \quad (2.4)$$

$$\sum_{i \in V, i \neq R} x_{iR} = M \quad (2.5)$$

$$u_i - u_j + 1 \leq P(1 - x_{i,j}) \quad \forall i, j \in V \setminus \{R\}, i \neq j \quad (2.6)$$

$$2 \leq u_i \leq P \quad \forall i \in V \setminus \{R\} \quad (2.7)$$

$$u_R = 1 \quad (2.8)$$

$$x_e \in \{0,1\} \quad \forall e \in E \quad (2.9)$$

La ecuación (2.1) representa la función objetivo. Esta busca minimizar la distancia total recorrida.

La ecuación (2.2) hace referencia a que la suma de arcos incidentes en el nodo debe ser 2, es decir, uno para entrada y otro para salida.

Por otro lado, la restricción (2.3) asegura que, en la solución óptima del modelo, la cantidad de arcos activos sea equivalente al número de nodos menos uno (para excluir el depósito), añadiendo un arco adicional por cada ruta que represente el retorno al depósito desde el último cliente visitado.

La restricción (2.4) establece que los conductores deben salir estrictamente del depósito.

Mientras que la restricción (2.5) indica que cada conductor debe regresar al depósito después de visitar al último cliente.

Con el fin de evitar subciclos, se recurre a las restricciones MTZ (Miller-Tucker-Zemlin), las cuales se utilizan para garantizar que cada cliente sea visitado exactamente una vez y asegurar la formación de rutas coherentes. La restricción (2.6) elimina subciclos. Se plantea una desigualdad donde se le adiciona una unidad a la suma de  $u_i$  y  $u_j$  con el fin de que esta se cumpla solo cuando  $x_e = 1$ , lo que indica que el cliente  $j$  sigue al cliente  $i$  en la ruta. Por su parte, la restricción número (2.7) mantiene el número de visitas entre 2 y P (32 clientes).

La ecuación (2.8) indica que las secuencias de las rutas empezarán desde el depósito.

Finalmente, la restricción (2.9) define a la variable binaria  $x_e$ , es decir solo puede tomar valores de 0 y 1.

### **2.3.2 Modelo de Clusterización**

El modelo matemático de clusterización corresponde a un problema de programación entera – binaria. Su formulación comienza definiendo los índices:  $i$  que representa a los clientes y  $j$  que representa a los centroides.

Se trabajó con dos parámetros. El primer parámetro fue la matriz  $d_{i,j}$  que muestra la distancia entre los clientes y los centroides. El segundo parámetro es  $C$  que corresponde al número máximo de clientes por clúster, en este caso sería 192.

En cuanto a la variable de decisión, se planteó  $x_{i,j}$ . Esta variable es binaria y representa la asignación de un cliente a un clúster.

#### **Índices:**

$$i = \{1,2,3, \dots, 9539\}$$

$$j = \text{centroides}$$

#### **Parámetros**

$$d_{i,j} = \text{matriz de distancia.}$$

$$C = \text{número máximo de clientes por clúster.}$$

### Variable de decisión

$x_{i,j}$  = variable de decisión binaria que toma el valor de 1 si el cliente  $i$  se asigna al centroide  $j$ , 0 caso contrario.

### Función objetivo

$$\min \sum_i \sum_j d_{i,j} x_{i,j} \quad (2.10)$$

### Restricciones

$$\sum_j x_{i,j} = 1 \quad \forall i \quad (2.11)$$

$$\sum_j x_{i,j} \leq 192 \quad \forall j \quad (2.12)$$

$$x_{i,j} \in \{0,1\} \quad (2.13)$$

La ecuación (2.10) representa la función objetivo. Esta busca minimizar la distancia total recorrida de los clústeres.

La restricción (2.11) garantiza que cada cliente sea asignado exactamente a un clúster.

La restricción (2.12) impone que la suma de las asignaciones por clúster no supere los 192 clientes.

La restricción (2.13) establece que  $x_{i,j}$  es binaria y, por ende, solo puede tomar valores de 0 y 1.

Para la subclusterización dentro de cada clúster se aplica el mismo modelo de clusterización cambiando el parámetro  $C$ .

Para las rutas con 180 clientes, se considera  $C = 32$ . Mientras que, para la ruta con 131 clientes, se agrega un cliente ficticio para considerar  $C = 22$ . Posteriormente, se elimina el cliente ficticio del subclúster en el que se encuentre.

Cabe destacar que el modelo matemático mTSP propuesto se presenta como una base para abordar el desafío de optimizar las rutas de venta directa. No obstante, se reconoce que, dada la

complejidad inherente del problema, se requiere una estrategia más avanzada para obtener soluciones eficientes en un tiempo razonable. Por lo tanto, se incorporó el método de machine learning K-Medoids para encontrar centroides y con ello emplear un modelo de clusterización para la realizar la nueva distribución de las rutas. Además de un algoritmo metaheurístico GRASP para obtener soluciones con respecto al ruteo. La sinergia entre el modelo mTSP, la clusterización y el enfoque GRASP proporcionó una estrategia integral y eficiente para la optimización de las rutas de venta directa en el contexto específico.

## 2.4 Uso De Software

Para el desarrollo de este trabajo se emplearon cuatro softwares, los cuales son: Wolfram Mathematica, Microsoft Excel, Google Sheets, Power BI. A continuación, se presenta una tabla en donde se puede identificar la función principal del software, sus ventajas y desventajas.

**Tabla 3.** *Softwares utilizados: ¿qué son?, ventajas y desventajas*

Software	¿Qué es?	Ventajas	Desventajas
<b>Wolfram Mathematica</b>	Es un lenguaje de programación y cómputo científico que destaca en “los campos de la computación técnica, incluyendo redes neuronales, aprendizaje automático, procesamiento de imágenes, geometría, ciencia de datos y visualizaciones” (Wolfram Mathematica, 2023).	<ul style="list-style-type: none"> <li>• Ofrece una amplia variedad de funciones integradas que facilitan el análisis y la visualización de datos, así como la implementación de algoritmos de optimización.</li> <li>• Tiene una gran capacidad para realizar cálculos con base de datos grandes en tiempos considerables.</li> </ul>	<ul style="list-style-type: none"> <li>• Puede resultar relativamente costoso contar con el software para usuarios individuales o pequeñas empresas.</li> </ul>
<b>Microsoft Excel</b>	Es “una herramienta muy eficaz para obtener información con significado a partir de grandes	<ul style="list-style-type: none"> <li>• Es una herramienta ampliamente utilizada por lo que muchos usuarios ya están</li> </ul>	<ul style="list-style-type: none"> <li>• Puede tornarse lento al momento de manejar grandes conjuntos de datos.</li> </ul>

	cantidades de datos” (Microsoft, 2023).	familiarizados con su interfaz.	
		<ul style="list-style-type: none"> <li>• Ofrece funciones que permiten realizar cálculos estadísticos y matemáticos, además de contar con tablas dinámicas.</li> </ul>	
<b>Google Sheets</b>	Es un espacio en la nube en donde se crean hojas de cálculo y se trabaja en tiempo real (Google, 2023)	<ul style="list-style-type: none"> <li>• La integración con Google Maps es beneficiosa para identificar posiciones geoespaciales.</li> <li>• Permite la colaboración en equipo desde cualquier dispositivo que cuente con conexión a internet.</li> </ul>	<ul style="list-style-type: none"> <li>• Presenta limitaciones en cuanto a procesos computacionales complejos e intensivos.</li> </ul>
<b>Power BI</b>	Es una plataforma unificada y escalable de inteligencia empresarial (BI) (Microsoft Power BI, 2023).	<ul style="list-style-type: none"> <li>• Ofrece herramientas avanzadas de visualización de datos que permiten crear paneles de control interactivos.</li> <li>• Se integra fácilmente con bases de datos de Microsoft Excel.</li> </ul>	<ul style="list-style-type: none"> <li>• Requiere capacitación sobre las distintas herramientas integradas para poder aprovechar al máximo el programa.</li> </ul>

## 2.5 Consideraciones Legales Y Éticas

En la investigación sobre la optimización de rutas directas para una empresa de la industria alimentaria en Guayaquil, se abordó diversas consideraciones legales y éticas para garantizar la validez y responsabilidad del estudio. Desde el punto de vista legal, se tuvo en cuenta la normativa nacional e internacional relacionada con el transporte de alimentos, asegurando el cumplimiento de leyes específicas como las establecidas por la Agencia de

Regulación y Control Fito y Zoosanitario (AGROCALIDAD) en Ecuador. Se analizó específicamente el artículo 22, el cual aborda los aspectos de higiene con el propósito de mantener la inocuidad y prevenir la contaminación (Jácome, 2009).

De manera similar, la investigación se alineó con regulaciones ambientales que controlan las emisiones de vehículos de transporte y la gestión adecuada de residuos relacionados con la industria alimentaria. El artículo 207 del Código Orgánico del Ambiente (Rivadeneira, 2016) responsabiliza a las empresas sobre el constante monitoreo de sus emisiones, descargas y vertidos con la finalidad de cumplir con el parámetro definido en la normativa ambiental.

Desde una perspectiva ética, fue crucial considerar el impacto social de la optimización de rutas. Esto incluyó evaluar cómo los cambios propuestos afectarán a los trabajadores del transporte y a las comunidades locales en términos de empleo, acceso a servicios y calidad de vida. De manera análoga, analizar una propuesta de solución relacionada a la equidad en la asignación de rutas también fue un criterio ético fundamental. Esta propuesta alternativa contempla la implementación de un sistema rotativo de asignación de rutas basado en criterios predefinidos de equidad. Este enfoque manual podría tener en cuenta factores adicionales, como las preferencias y habilidades individuales de los vendedores, así como consideraciones prácticas, como la proximidad de sus residencias a las áreas de venta. Un comité ético interno podría ser designado para revisar y ajustar periódicamente las asignaciones de rutas, garantizando así que la carga de trabajo sea distribuida equitativamente y se adapte a las necesidades cambiantes del personal.

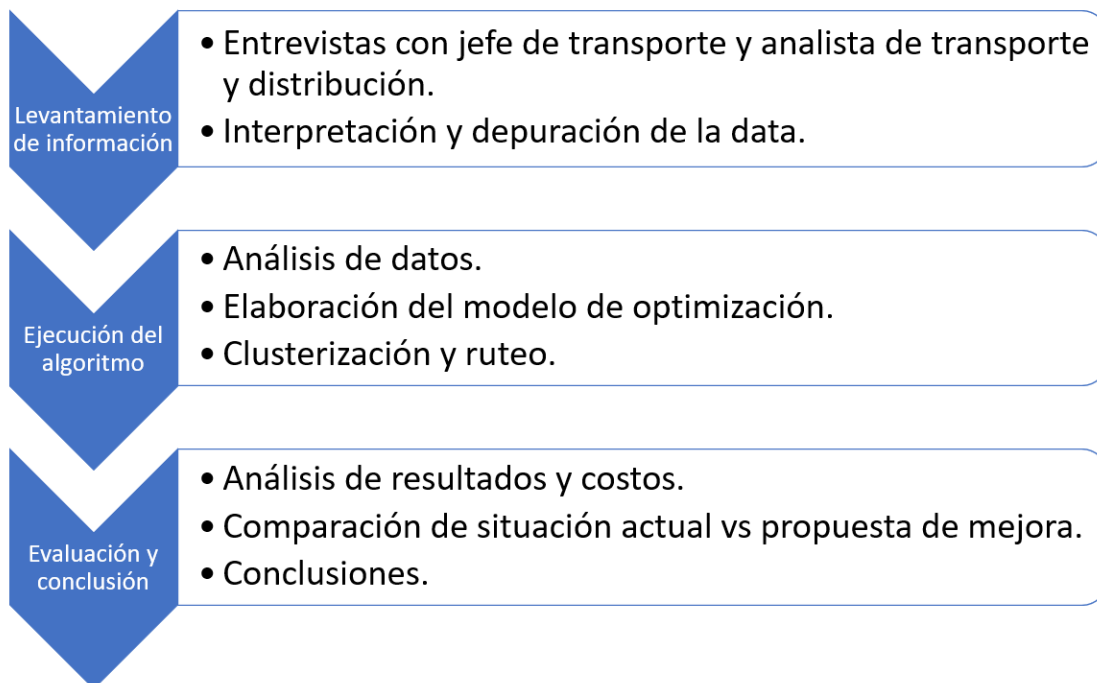
Por último, se abordó el impacto económico de la optimización de rutas. Esto implicó evaluar los costos asociados con la implementación de nuevas estrategias logísticas, así como los beneficios económicos que la empresa podría obtener a través de la eficiencia operativa.

## 2.6 Fases Del Proyecto

En la *Figura 5* se presenta el diagrama de las fases del proyecto. Estas fases fueron definidas con el propósito de contar con una clara organización de los pasos a realizar para no solo culminar el proyecto sino también alcanzar los objetivos planteados.

La primera fase es el levantamiento de información; ésta engloba entrevistas con actores relevantes de la empresa de alimentos y la interpretación y depuración de la data. Como segunda fase está la ejecución del algoritmo. Los pasos para llevar a cabo esta fase contemplan el análisis de datos, elaboración del modelo de optimización y la clusterización y ruteo. Finalmente, la última fase se la denominó evaluación y conclusión. En esta etapa se analizaron los resultados y costos de implementación del algoritmo. También se comparó la situación de situación actual y la propuesta de mejora, para finalizar con las conclusiones.

**Figura 5.** *Fases del proyecto*





## **2.7 Cronograma De Trabajo**

Con la finalidad de establecer un orden y tener un mejor control de las actividades que se necesitaban realizar para completar las fases previamente presentadas se definió un cronograma de trabajo, el mismo que puede visualizarse en la *Figura 6*. En este se detallan las actividades por realizar, las fechas de inicio y fin, y su duración en días.



## CAPÍTULO 3

### 3. RESULTADOS Y ANÁLISIS

#### 3.1 Clusterización

El proceso se inició con la aplicación del método “K-Medoids” para hallar 50 centroides que sirvieron de punto céntrico para armar a su alrededor los clústeres. Los 50 centroides fueron uno de los parámetros de entrada para el modelo matemático de clusterización. Este modelo generó 50 clústeres: 49 clústeres contienen 192 clientes cada uno, y un clúster alberga 131 clientes debido a que se cuenta con 9.539 clientes y estos no pueden ser divididos en grupos iguales de 192 clientes. Los clústeres fueron diseñados con la finalidad de satisfacer la demanda de 32 clientes diarios, equivalente a 192 clientes semanalmente (de lunes a sábado).

Posteriormente, se llevó a cabo una subclusterización en cada clúster con el fin de determinar la asignación diaria de clientes.

Este procedimiento arrojó como resultado 6 subclústeres de 32 clientes cada uno para todos los clústeres (rutas) de 192 clientes. Para el clúster de 131 clientes se obtuvo 6 subclústeres: 5 de 22 clientes y 1 de 21 clientes.

**Tabla 4.** *Clientes por ruta*

Ruta	Día	Secuencia de visita
1	2	{4636, 8102, 269, 98, 229, 4707, 86, 207, 147, 99, 5037, 5203, 5141, 5137, 5235, 7642, 4709, 5076, 5077, 5063, 5035, 5024, 5164, 5162, 5221, 5083, 5121, 5152, 5160, 7660, 184, 91}
	3	{301, 267, 289, 264, 230, 3592, 213, 122, 225, 105, 193, 292, 278, 281, 190, 188, 199, 129, 170, 111, 165, 130, 228, 215, 88, 293, 216, 166, 167, 96, 146, 263}
	4	{4273, 4166, 4704, 4079, 5180, 5036, 5140, 5033, 5079, 5027, 5146, 5190, 5090, 5220, 5055, 5156, 5150, 4265, 4191, 117, 200, 273, 5261, 5183, 5237, 5021, 4697, 4078, 4081, 5132, 5241, 5207}
	5	{5062, 5109, 4668, 5050, 5255, 5194, 5230, 5078, 5214, 5049, 5248, 5106, 5080, 5098, 5104, 5172, 5179, 5122, 5069, 5143, 5208, 5082, 5071, 5138, 5186, 5238, 5028, 5262, 4698, 5215, 5187, 5209}

	6	{5218, 5056, 5096, 5184, 5173, 5127, 5130, 5087, 5073, 5201, 5103, 5213, 5159, 5126, 5217, 5195, 5181, 5234, 5161, 5070, 5247, 5115, 5244, 5107, 5239, 4638, 5131, 5068, 5134, 5025, 5243, 5067}
	7	{5092, 5105, 5174, 5178, 5148, 5097, 5129, 5051, 5136, 5053, 5075, 5155, 5058, 5084, 5047, 5086, 5147, 5211, 5038, 5040, 5041, 5165, 5042, 5043, 5044, 5193, 5099, 5210, 5048, 5039, 5246, 5228}
<b>50</b>	2	{2455, 2444, 2431, 2432, 2438, 2467, 2427, 2460, 2458, 6892, 6831, 6849, 6796, 6826, 2457, 2472, 2456, 2482, 2426, 6797, 6827, 2459}
	3	{4381, 4352, 4391, 4055, 4041, 3920, 3980, 4030, 2434, 4042, 3918, 3970, 4044, 3968, 4051, 4298, 4049, 4382, 4323, 4383, 4408, 4377}
	4	{3613, 2483, 2474, 2462, 2463, 2486, 4384, 4385, 2461, 4379, 4293, 4029, 3899, 4036, 4050, 3614, 6828, 6798, 6799, 4551, 2440, 2435}
	5	{4346, 4308, 4317, 4358, 3553, 6750, 6751, 6747, 6887, 6748, 6749, 6765, 6850, 6800, 6879, 4345, 4347, 4313, 3616, 3554, 3617, 3573}
	6	{4378, 4023, 4018, 4370, 4369, 2451, 4388, 4288, 4386, 4389, 4411, 2428, 4412, 4362, 4380, 2445, 2441, 4311, 4365, 4344, 4294, 4303}
	7	{2442, 7672, 2436, 4277, 2475, 2469, 2473, 4024, 4028, 2477, 2476, 2454, 2465, 2470, 8068, 2487, 2485, 2429, 2430, 2433, 2439}

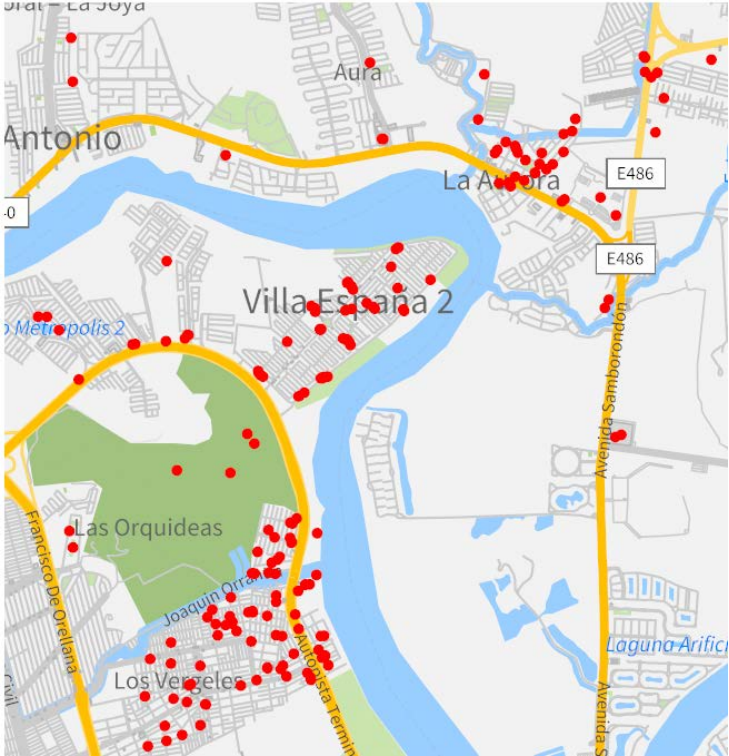
*Nota:* Los valores de la columna “Día” corresponden a: 2 = lunes, 3 = martes, 4 = miércoles, 5 = jueves, 6 = viernes, 7 = sábado.

La Tabla 4 muestra la distribución de los clientes para el primer y último clúster. Para el caso de la ruta 1, se cuenta con 32 clientes por atenderse por día, es decir que la ruta atenderá 192 clientes semanalmente. De manera similar sucede con las rutas 2 a la 49 (ver Apéndice A). Por otro lado, se cuenta con la ruta 50 que cuenta con un total de 131 clientes por atenderse semanalmente. En este caso, un día 21 clientes serán atendidos y el resto de la semana 22 clientes.

A continuación, se presenta la distribución geográfica de los clústeres dentro de la primera ruta (Figura 7) y última (Figura 8) ruta.

Dada la extensa cartera de clientes para este proyecto, en el Apéndice B se hallan los mapas de las rutas 2 a la 49.

**Figura 7.** *Cientes de la ruta 1*



**Figura 8.** *Cientes de la ruta 50*



### 3.2 Asignación de Rutas

Después de la implementación del algoritmo metaheurístico GRASP se obtuvieron las secuencias de visita de todas las rutas.

**Tabla 5.** *Secuencia de visita por ruta*

Ruta	Día	Clúster
1	2	{0, 86, 91, 98, 99, 147, 184, 207, 229, 269, 4636, 4707, 4709, 5024, 5035, 5037, 5063, 5076, 5077, 5083, 5121, 5137, 5141, 5152, 5160, 5162, 5164, 5203, 5221, 5235, 7642, 7660, 8102, 0}
	3	{0, 88, 96, 105, 111, 122, 129, 130, 146, 165, 166, 167, 170, 188, 190, 193, 199, 213, 215, 216, 225, 228, 230, 263, 264, 267, 278, 281, 289, 292, 293, 301, 3592, 0}
	4	{0, 117, 200, 273, 4078, 4079, 4081, 4166, 4191, 4265, 4273, 4697, 4704, 5021, 5027, 5033, 5036, 5055, 5079, 5090, 5132, 5140, 5146, 5150, 5156, 5180, 5183, 5190, 5207, 5220, 5237, 5241, 5261, 0}
	5	{0, 4668, 4698, 5028, 5049, 5050, 5062, 5069, 5071, 5078, 5080, 5082, 5098, 5104, 5106, 5109, 5122, 5138, 5143, 5172, 5179, 5186, 5187, 5194, 5208, 5209, 5214, 5215, 5230, 5238, 5248, 5255, 5262, 0}
	6	{0, 4638, 5025, 5056, 5067, 5068, 5070, 5073, 5087, 5096, 5103, 5107, 5115, 5126, 5127, 5130, 5131, 5134, 5159, 5161, 5173, 5181, 5184, 5195, 5201, 5213, 5217, 5218, 5234, 5239, 5243, 5244, 5247, 0}
	7	{0, 5038, 5039, 5040, 5041, 5042, 5043, 5044, 5047, 5048, 5051, 5053, 5058, 5075, 5084, 5086, 5092, 5097, 5099, 5105, 5129, 5136, 5147, 5148, 5155, 5165, 5174, 5178, 5193, 5210, 5211, 5228, 5246, 0}
	50	2
3		{0, 2434, 3918, 3920, 3968, 3970, 3980, 4030, 4041, 4042, 4044, 4049, 4051, 4055, 4298, 4323, 4352, 4377, 4381, 4382, 4383, 4391, 4408, 0}
4		{0, 2435, 2440, 2461, 2462, 2463, 2474, 2483, 2486, 3613, 3614, 3899, 4029, 4036, 4050, 4293, 4379, 4384, 4385, 4551, 6798, 6799, 6828, 0}
5		{0, 3553, 3554, 3573, 3616, 3617, 4308, 4313, 4317, 4345, 4346, 4347, 4358, 6747, 6748, 6749, 6750, 6751, 6765, 6800, 6850, 6879, 6887, 0}
6		{0, 2428, 2441, 2445, 2451, 4018, 4023, 4288, 4294, 4303, 4311, 4344, 4362, 4365, 4369, 4370, 4378, 4380, 4386, 4388, 4389, 4411, 4412, 0}
7		{0, 2429, 2430, 2433, 2436, 2439, 2442, 2454, 2465, 2469, 2470, 2473, 2475, 2476, 2477, 2485, 2487, 4024, 4028, 4277, 7672, 8068, 0}

*Nota:* Los valores de la columna “Día” corresponden a: 2 = lunes, 3 = martes, 4 = miércoles,

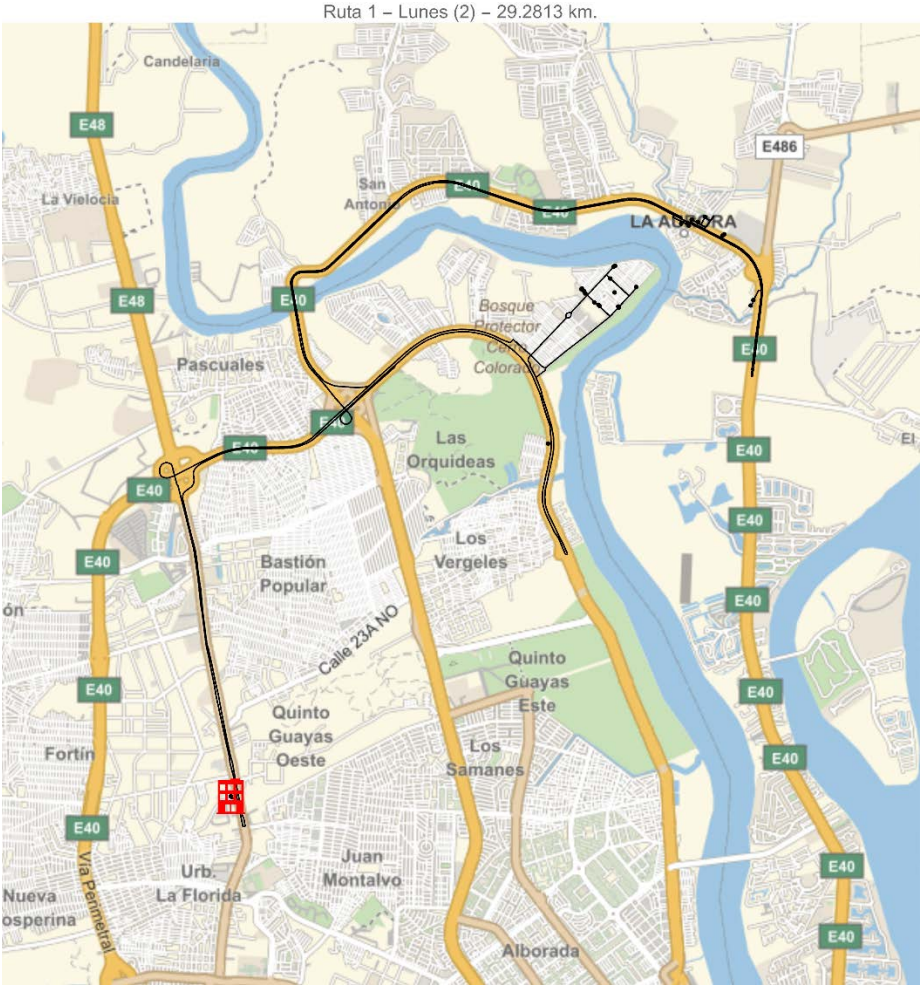
5 = jueves, 6 = viernes, 7 = sábado.

La *Tabla 5* detalla el orden de visita a los clientes por ruta y por día para las rutas 1 y 50. Cada ruta parte desde el centro de distribución (valor 0) y retorna después de completar todas las visitas. De manera similar sucede con las rutas 2 a la 49 (ver Apéndice C).

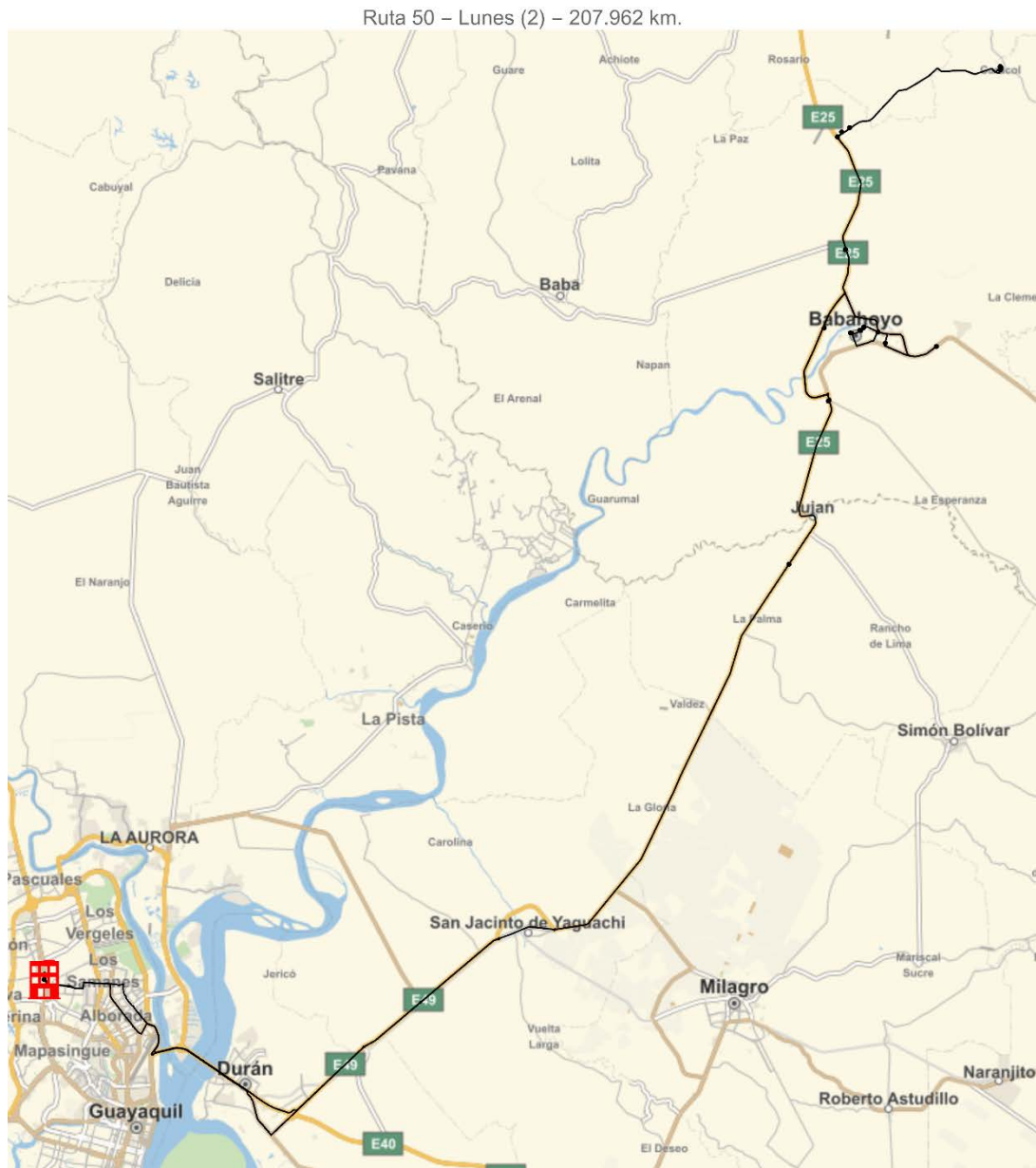
Las distancias totales recorridas por ruta se encuentran anexadas al Apéndice D.

A continuación, las *Figura 9* y *Figura 10* hacen referencia a los recorridos que tanto la ruta 1 como la ruta 50 realizan para el lunes.

**Figura 9.** *Recorrido de la ruta 1 – lunes (2)*



**Figura 10. Recorrido de la ruta 50 – lunes (2)**



Los recorridos para el resto de la semana para estas y las rutas 2 a la 49 se podrán observar en el Apéndice E.



### 3.3 Análisis de Costo de la Propuesta

El software que se requirió para dar solución al problema de optimización de rutas de venta directa fue Wolfram Mathematica. Se calculó el presupuesto de la propuesta con base a los recursos tecnológicos y humanos necesarios.

**Tabla 6.** *Costo de la propuesta*

<b>Rubro</b>	<b>Mensual</b>	<b>Anual</b>
<b>Licencia de Wolfram Mathematica</b>	\$ 138,33	\$ 1.660,00
<b>Consumo de energía eléctrica (proporcional)</b>	\$ 5,00	\$ 60,00
<b>Consumo de internet (proporcional)</b>	\$ 10,78	\$ 129,36
<b>Remuneración del analista</b>	\$ 600,00	\$ 7.200,00
<b>Computadora</b>	\$ 58,25	\$ 699,00
<b>Mantenimiento de computadora</b>	\$ 5,00	\$ 60,00
<b>TOTAL</b>	\$ 817,36	
<b>TOTAL PRIMER AÑO</b>		\$ 9.808,36
<b>TOTAL AÑOS POSTERIORES</b>		\$ 9.109,36

La *Tabla 6* muestra que el valor anual que se deberá cancelar el primer año es \$ 9.808,36. Este valor incluye el pago de servicios básicos, salarios, licencia del software y la adquisición de una computadora. No obstante, el valor anual de los siguientes años será \$ 9.109,36. Éste es menor ya que no se deberá incurrir en la compra de un nuevo computador.

Entre los beneficios que brinda el contar con la licencia de Wolfram Mathematica se tienen (Wolfram Mathematica, 2023):

- Acceso completo a Wolfram Language y Wolfram Knowledgebase.
- Creación y almacenamiento de cuadernos en la nube.
- Interfaz de implementación estática.

### 3.4 Comparativa entre Situación Actual y Propuesta de Mejora

Para los cálculos de esta sección se tuvo en consideración que el combustible que se le administra a la flota es diésel y que cada vehículo recorre 45 km por cada galón de diésel.

#### 3.4.1 Distancia Total Recorrida

En la *Tabla 7* se muestra la reducción de distancia total recorrida gracias a la nueva propuesta de rutas de venta directa. Las nuevas 50 rutas recorren semanalmente 20.346,0 km, generando un ahorro de 16.147,5 km en contraste con el recorrido actual de las 53 rutas.

*Tabla 7. Comparación de distancia total recorrida*

<b>Recorrido (km)</b>	<b>Actual</b>	<b>Propuesta</b>	<b>Reducción</b>
Semanal	36.493,5	20.346,0	16.147,5
Mensual	158.138,5	88.166,0	69.972,5
Anual	1.897.662,0	1.057.992,0	839.670,0

#### 3.4.2 Impacto Medioambiental

En la *Tabla 8* se muestra la reducción de las emisiones de dióxido de carbono,  $CO_2$ . Con la propuesta se disminuye 187 tn aproximadamente cada año, lo cual se alinea con el Objetivo de Desarrollo Sostenible (ODS) 12 "Producción y Consumo Responsables" y con el compromiso medioambiental de la empresa.

*Tabla 8. Comparación de emisiones de  $CO_2$*

<b>Emisiones de <math>CO_2</math> (Tn)</b>	<b>Actual</b>	<b>Propuesta</b>	<b>Reducción</b>
Semanal	8,10	4,52	3,58
Mensual	35,11	19,57	15,54
Anual	421,28	234,87	186,41

*Nota:* Las emisiones de  $CO_2$  fueron obtenidas gracias a la calculadora de emisiones de Sun Earth Tools.

Además, por medio de la calculadora de emisiones del Instituto Vasco de Logística y Movilidad Sostenible se obtuvo las emisiones anuales de los óxidos de nitrógeno, *NOx*.

**Tabla 9.** Comparación de emisiones de *NOx*

<b>Emisiones de <i>NOx</i> (kg)</b>	<b>Actual</b>	<b>Propuesta</b>	<b>Reducción</b>
Anual	7,59	4,23	3,36

### 3.4.3 Ahorro en costos

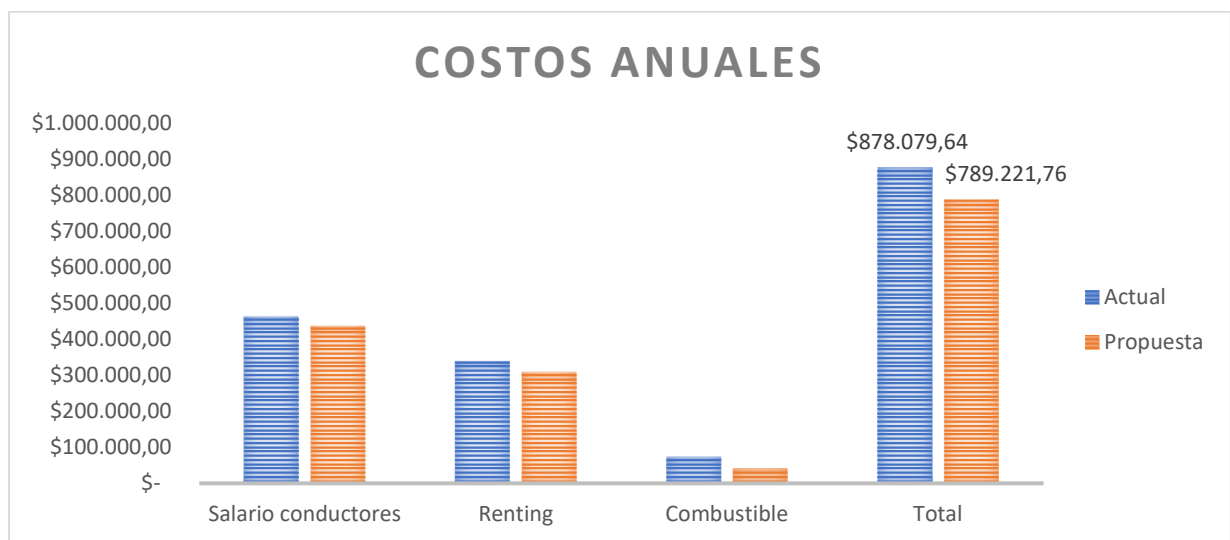
Es importante analizar el ahorro que se obtendría una vez se aplique la solución propuesta. Es por esto por lo que con base a los costos inmersos en la operación se calcularon los costos mensuales y anuales tanto de la situación actual y la propuesta.

**Tabla 10.** Comparación de costos

	<b>Actual</b>	<b>Propuesta</b>
Vehículos propios	19	19
Vehículos rentados	34	31
Total vehículos	53	50
<b>Conductores</b>	53	50
Salarios conductores	\$ 730,00	\$ 730,00
<b>Salario total mensual</b>	\$ 38.690,00	\$ 36.500,00
<b>Salario total anual</b>	\$ 464.280,00	\$ 438.000,00
Renting por vehículo	\$ 833,33	\$ 833,33
<b>Renting total mensual</b>	\$ 28.333,22	\$ 25.833,23
<b>Renting total anual</b>	\$ 339.998,64	\$ 309.998,76
Galón de diesel	\$ 1,75	\$ 1,75
Km por galón	45	45
Galones requeridos semanal	811	453
Combustible semanal	\$ 1.419,25	\$ 792,75
<b>Combustible mensual</b>	\$ 6.150,08	\$ 3.435,25
<b>Combustible anual</b>	\$ 73.801,00	\$ 41.223,00
<b>TOTAL MENSUAL</b>	<b>\$ 73.173,30</b>	<b>\$ 65.768,48</b>
<b>TOTAL ANUAL</b>	<b>\$ 878.079,64</b>	<b>\$ 789.221,76</b>

La Tabla 10 explica en detalle los rubros relacionados con la operación de las rutas de venta directa. En esta tabla se calculan los costos de salarios de conductores, contratación de flota (renting), combustible. Comparando los costos anuales de la situación actual y la propuesta se evidencia una disminución de \$ 88.857,88.

**Figura 11.** Comparación de costos anuales



La Figura 11 brindan una representación gráfica de los costos anuales presentados en la Tabla 10. Este diagrama presenta por separado los 3 rubros que se consideraron para el cálculo de los costos totales.

**Tabla 11.** Liquidación del personal

Reducción personal	3
Años promedio trabajados (supuesto)	4
<b>Liquidación por persona</b>	<b>\$ 1.177,47</b>
<b>Liquidación total</b>	<b>\$ 3.532,41</b>

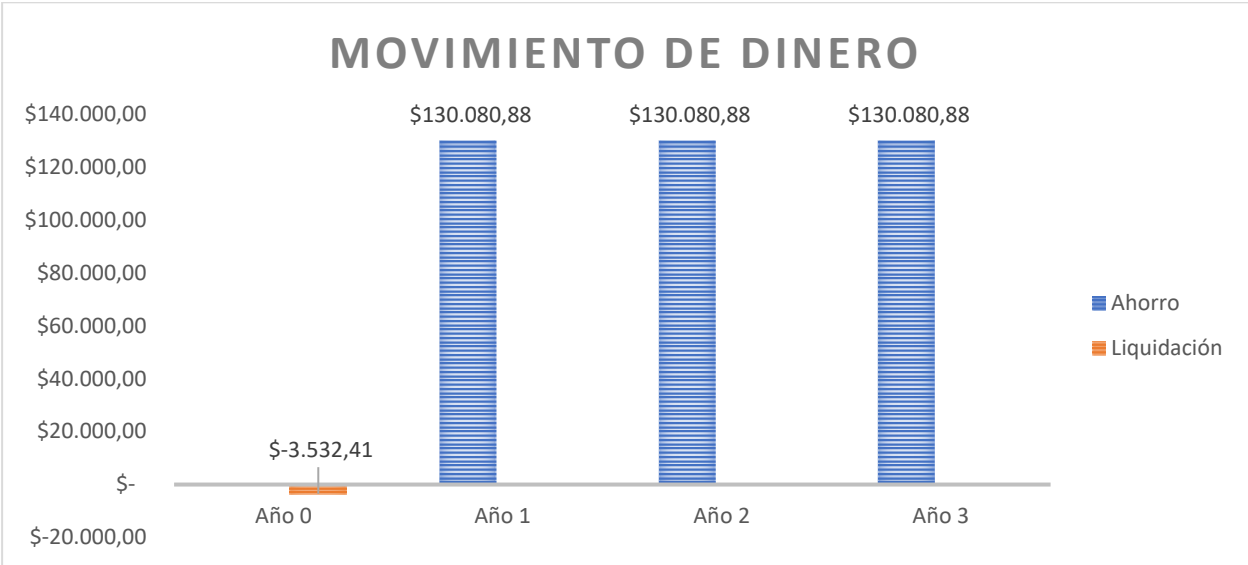
*Nota:* La liquidación por persona fue obtenida gracias a la calculadora de liquidaciones del Ministerio de Trabajo del Ecuador.

La optimización de las 53 rutas generó 50 nuevas rutas, lo cual implica que se necesitarían 50 conductores en lugar de 53. Esta reducción de personal genera un desembolso por conceptos

de liquidación. Para el cálculo de la liquidación total se consideró los años de trabajo en promedio, obteniendo así un total de \$ 3.532,41. La Tabla 11 muestra detalles del cálculo de liquidaciones.

Teniendo en cuenta el cálculo de las liquidaciones se elaboró un diagrama de barras que muestre el movimiento de dinero para los 3 primeros años después de aplicar la propuesta. Esto se puede apreciar en la Figura 12.

**Figura 12.** Movimiento de dinero para los 3 primeros años



### 3.5 Entregables

Los entregables que se le proporcionarán a la empresa son:

- Archivo de Microsoft Excel con las rutas optimizadas por medio de la implementación del algoritmo metaheurístico.
- Mapas georreferenciados que muestren el recorrido de cada ruta.
- Comparativa entre la situación actual de las rutas de venta directa y la propuesta de mejora.

## CAPÍTULO 4

### 4 CONCLUSIONES Y RECOMENDACIONES

#### 4.1 Conclusiones

La optimización de las rutas de venta directa se logró de manera sobresaliente, cumpliendo de manera efectiva con los requerimientos establecidos por la empresa. Para atender las necesidades de los clientes, se implementó un modelo matemático que se centró en minimizar la distancia recorrida. Inicialmente, se llevó a cabo una clusterización eficiente de los 9.539 clientes, dividiéndolos en 50 clústeres y 6 subclústeres. Esta segmentación estratégica permitió la posterior resolución del problema MTPS mediante VMCM con GRASP y mejora 2-opt.

Los resultados fueron satisfactorios ya que se redujo la flota de camiones de 53 a 50 y transformando de manera análoga las 53 rutas mal distribuidas y sobrecargadas en 50 rutas distribuidas eficientemente. Este enfoque no solo optimizó la logística de entrega, sino que también mejoró significativamente la satisfacción del cliente al garantizar un servicio más eficiente y puntual.

Se llevó a cabo una investigación detallada para realizar una estimación precisa de los costos. La optimización no solo se tradujo en una mejora operativa, sino que también se reflejó en una reducción sustancial de los costos totales, disminuyendo de \$ 878.079,64 a \$ 789.221,76. Este ahorro financiero significativo demuestra el impacto positivo de la implementación de algoritmos metaheurísticos en la gestión logística de la empresa.

Adicionalmente, se utilizó Wolfram Mathematica para calcular la distancia recorrida antes y después de la optimización. Los resultados fueron impresionantes, pasando de 1.897.662,0 km recorridos anualmente a 1.057.992,0 km. Esta reducción en el recorrido no solo se tradujo en eficiencia operativa, sino también en un impacto ambiental positivo al disminuir la huella de carbono.

La conciencia ambiental también fue una prioridad al evaluar las emisiones de  $CO_2$  anuales. Las emisiones de  $CO_2$  anuales se redujeron significativamente, pasando de 424,63 a 234,87. Este logro subraya el compromiso de la empresa con la sostenibilidad y demuestra cómo la optimización de rutas puede tener beneficios tanto económicos como ambientales.

En conjunto, los resultados obtenidos respaldan la efectividad de la implementación de algoritmos metaheurísticos en la optimización de rutas de venta directa para empresas de la industria alimentaria en Guayaquil

## **4.2 Recomendaciones**

En el contexto de la optimización de rutas de venta directa en la industria alimentaria mediante la implementación de algoritmos metaheurísticos, se presentan diversas recomendaciones que buscan mejorar la respuesta del sistema y hacerlo más realista.

- Se sugiere evaluar la posibilidad de incorporar ventanas de tiempo en la planificación de rutas, contemplando la utilización de tiempos promedio por cliente. Esta adición permitiría optimizar las entregas al tener en cuenta las restricciones de tiempo, mejorando así la eficiencia del proceso de venta directa. Además, otra consideración valiosa para la personalización de rutas sería la inclusión de restricciones de priorización por cliente. Dado que algunos clientes pueden tener necesidades especiales, asignar prioridades en la planificación podría contribuir significativamente a una atención al cliente más efectiva. Estas restricciones adicionales no solo modelarían de manera más realista la situación, sino que también enriquecerían la adaptabilidad del sistema a las necesidades específicas de cada cliente.
- Además, se destaca considerar la posición geográfica del estacionamiento relacionada al cliente al diseñar las rutas. Se pudo observar que existen zonas donde los clientes están alejados de las vías transitadas, por lo que planificar el estacionamiento estratégico del

vehículo cerca de esos puntos de difícil acceso puede mejorar la eficiencia y reducir tiempos de desplazamiento innecesarios.

- Para futuros estudios, se aconseja la aplicación de otros algoritmos metaheurísticos que aborden el problema MTSP. Comparar los resultados obtenidos con estos algoritmos adicionales permitirá evaluar la robustez y eficacia de la propuesta actual, contribuyendo a la selección del algoritmo más idóneo para la optimización de rutas de venta directa en la industria alimentaria.
- Se sugiere considerar la posibilidad de incorporar información sobre el tráfico actual o estimado durante las 8 horas de trabajo en la asignación de rutas a los subclústeres. Integrar datos de tráfico en tiempo real o estimaciones precisas permitirá ajustar las rutas de manera dinámica, adaptándolas a las condiciones cambiantes del tráfico. Esto no solo optimizará los tiempos de desplazamiento, sino que también ayudará a prevenir posibles retrasos y mejorar la puntualidad en las visitas a los clientes.
- A pesar de haber calculado una notable disminución de emisiones de CO<sub>2</sub>, pasando de 424,63 a 234,87 toneladas anuales mediante la implementación de optimización de rutas, se identifica una oportunidad adicional para mejorar aún más este valor. Se sugiere considerar la incorporación de una flota de vehículos eléctricos en la logística de entrega. La adopción de vehículos eléctricos no solo reduciría significativamente las emisiones de gases contaminantes, sino que también contribuiría al compromiso ambiental de la empresa. Este enfoque sostenible no solo mejoraría la eficiencia operativa, sino que también posiciona a la empresa como un actor responsable y consciente de su impacto ambiental, generando beneficios tanto económicos como medioambientales a largo plazo.



## BIBLIOGRAFÍA

- Ballou, R. H. (1991). *Logística Empresarial: control y planificación*. (D. d. Santos, Ed.)
- Cheikhrouhou, O., & Khouf, I. (25 de Febrero de 2021). *A Comprehensive Survey on the Multiple Travelling*. Obtenido de *Computer Science Review*: <https://doi.org/10.1016/j.cosrev/2021.100369>
- Cheikhrouhou, O., & Khoufi, I. (25 de Febrero de 2021). *A Comprehensive Survey on the Multiple Travelling Salesman Problem: Applications, Approaches and Taxonomy*. *Computer Science Review*. doi:[doi.org/10.1016/j.cosrev.2021.100369](https://doi.org/10.1016/j.cosrev.2021.100369)
- Feo, T. A., & Resende, M. G. (Marzo de 1995). *Greedy Randomized Adaptative Search Procedures*. *Journal of Global Optimization*, 109-133. doi:<https://doi.org/10.1007/BF01096763>
- Google. (2023). *Google Workspace*. Obtenido de <https://www.google.com/intl/es/sheets/about/>
- Granada Echeverri, M., Cubides Rivera, L. C., & Bolaños, R. I. (2022). *Modelos matemáticos para la programación óptima de rutas: formulación e implementación*. 186. Pereira, Colombia: Universidad Tecnológica de Pereira. Obtenido de <https://repositorio.utp.edu.co/server/api/core/bitstreams/8db5775f-5d70-41e9-b887-5fba3151e1e0/content>
- Jácome, F. (17 de Diciembre de 2009). *AGROCALIDAD*. Obtenido de Buenas Prácticas AGRÍCOLAS: <https://www.agrocalidad.gob.ec/wp-content/uploads/2023/09/Gui%CC%81a-de-BPA-General-jul.pdf>
- Jauhari, A., Rahabillah, M. R., Anamisa, D. R., Haq, A. F., Mufarroha, F. A., & Purnama, A. A. (2023). *Comparison of K-means and K-medoids in Tourist Attraction Clustering based on Visitor Characteristics*. *IEEE Xplore*. doi:[10.1109/ICSINTESA56431.2022.10041666](https://doi.org/10.1109/ICSINTESA56431.2022.10041666)
- Microsoft. (2023). *Microsoft*. Obtenido de <https://support.microsoft.com/es-es/office/tareas-b%3A1sicas-en-excel-dc775dd1-fa52-430f-9c3c-d998d1735fca>

- Microsoft Power BI. (2023). *Microsoft Power BI*. Obtenido de <https://powerbi.microsoft.com/es-es/what-is-power-bi/>
- Murray, M. (09 de Abril de 2018). *Live About*. Obtenido de <https://www.liveabout.com/route-planning-2221322>
- Ocampo, E. M., Bolaños, R. I., & Echeverri, M. G. (02 de Junio de 2014). Solución del problema de múltiples agentes viajeros mediante técnicas heurísticas. *Scientia Et Technica*, 174-182. doi:<https://www.redalyc.org/articulo.oa?id=84931680004>
- Rivadeneira, G. (21 de diciembre de 2016). *Organización de las Naciones Unidas para la Alimentación y la Agricultura*. Obtenido de <https://faolex.fao.org/docs/pdf/ecu167116.pdf>
- Sandoya, F. (2023). *Metaheurísticas: Herramientas Inteligente para Resolver Problemas Complejos* (Segunda Edición ed.). Guayaquil, Ecuador.
- Uddin, F., Riaz, N., Manan, A., Mahmood, I., Song, O.-Y., Malik, A. J., & Abbasi, A. A. (2023, Junio 16). An Improvement to the 2-Opt Heuristic Algorithm for Approximation of Optimal TSP Tour. *MDPI*. doi:<https://doi.org/10.3390/app13127339>
- Wolfram Mathematica. (2023). *Wolfram Mathematica*. Obtenido de <https://www.wolfram.com/mathematica/>

# APÉNDICES

## APÉNDICE A

Clientes por clúster

Ruta	Día	Clúster
1	2	{86,91,98,99,147,184,207,229,269,4636,4707,4709,5024,5035,5037,5063,5076,5077,5083,5121,5137,5141,5152,5160,5162,5164,5203,5221,5235,7642,7660,8102}
	3	{88,96,105,111,122,129,130,146,165,166,167,170,188,190,193,199,213,215,216,225,228,230,263,264,267,278,281,289,292,293,301,3592}
	4	{117,200,273,4078,4079,4081,4166,4191,4265,4273,4697,4704,5021,5027,5033,5036,5055,5079,5090,5132,5140,5146,5150,5156,5180,5183,5190,5207,5220,5237,5241,5261}
	5	{4668,4698,5028,5049,5050,5062,5069,5071,5078,5080,5082,5098,5104,5106,5109,5122,5138,5143,5172,5179,5186,5187,5194,5208,5209,5214,5215,5230,5238,5248,5255,5262}
	6	{4638,5025,5056,5067,5068,5070,5073,5087,5096,5103,5107,5115,5126,5127,5130,5131,5134,5159,5161,5173,5181,5184,5195,5201,5213,5217,5218,5234,5239,5243,5244,5247}
	7	{5038,5039,5040,5041,5042,5043,5044,5047,5048,5051,5053,5058,5075,5084,5086,5092,5097,5099,5105,5129,5136,5147,5148,5155,5165,5174,5178,5193,5210,5211,5228,5246}
	2	2
3		{1516,1525,1527,1529,1550,1559,1567,1597,1607,1611,1621,1624,1625,1626,1632,1644,1646,1647,1648,1666,1670,1675,1693,1696,1702,1706,1711,1713,1717,4639,4703,8701}
4		{1549,1596,1610,1628,1643,1689,1700,1709,3241,3260,3264,3279,3280,3281,3284,3294,3302,3314,3317,3324,3326,4071,4653,4654,4658,4663,4664,4665,4681,4683,4689,4694}
5		{1541,1542,1543,1568,1571,1573,1574,1577,1579,1580,1582,1590,1598,1605,1613,1616,1620,1649,1653,1668,1682,1697,1705,1707,1725,1782,1836,1914,4644,4655,4685,4687}
6		{1552,1553,1554,1555,1558,1576,1581,1584,1603,1606,1619,1642,1660,1669,1676,1681,1683,1684,1690,1691,1712,1716,3239,4631,4641,4650,4657,4660,4680,4693,4695,9538}
7		{3236,3240,3243,3253,3254,3262,3265,3275,3277,3288,3291,3295,3296,3297,3298,3299,3300,3301,3303,3304,3310,3311,3312,3322,3327,3328,3333,3339,3340,4642,4647,4691}
3		2
	3	{30,32,34,60,61,77,118,120,121,124,125,127,128,151,156,157,158,159,177,181,185,187,192,196,197,198,201,203,204,2413,3110,3120}
	4	{40,63,69,76,81,97,100,101,106,109,110,112,119,123,148,169,175,205,209,210,211,222,3455,3458,3494,6714,6740,6743,6771,6840,6858,6863}
	5	{50,52,57,68,71,78,82,107,150,178,219,227,260,279,282,3430,3431,3435,3436,3444,3460,3472,3479,3482,3483,3485,3543,3556,3567,3568,3600,3607}
	6	{48,51,70,72,74,80,83,94,102,135,142,149,160,176,265,268,271,290,3410,4702,5032,5046,5066,5119,5120,5125,5188,5197,5212,5249,5254,9476}
	7	{29,33,59,67,85,87,90,126,133,136,139,143,155,172,174,179,180,183,206,212,220,221,3442,3445,3446,3450,3451,3462,3470,3538,3570,4282}

		{3049,3081,4072,4077,4089,4091,4097,4102,4105,4106,4117,4118,4120,4140,4161,416
	2	3,4167,4175,4177,4179,4181,4182,4198,4214,4216,4228,4238,4243,4244,4248,4260,426
		7}
		{4074,4075,4087,4100,4103,4114,4115,4124,4126,4128,4137,4142,4144,4145,4150,415
	3	4,4159,4184,4193,4197,4202,4205,4213,4235,4236,4245,4252,4268,4272,4275,4640,465
		1}
		{3114,4067,4084,4090,4092,4119,4131,4136,4146,4192,4209,4220,4221,4222,4226,422
	4	9,4234,4271,4669,4977,5057,5059,5081,5091,5102,5108,5111,5133,5142,5257,5258,526
4		3}
		{3073,4066,4068,4076,4082,4083,4086,4093,4096,4099,4101,4110,4111,4116,4143,414
	5	7,4164,4172,4174,4178,4180,4186,4190,4204,4207,4224,4230,4231,4233,4241,4256,427
		4}
		{4069,4085,4121,4125,4135,4138,4141,4148,4149,4171,4188,4196,4206,4208,4210,421
	6	5,4217,4218,4237,4249,4253,4255,4259,4262,4270,4632,4682,4944,4978,6322,6358,642
		8}
		{4073,4080,4094,4095,4107,4108,4109,4112,4113,4122,4129,4130,4132,4133,4134,413
	7	9,4151,4152,4162,4168,4183,4185,4187,4195,4232,4246,4676,4945,5060,5110,5135,514
		4}
		{2917,2918,2935,2947,2956,2962,2990,2991,2992,2994,3014,3031,3076,3088,3099,310
	2	2,4088,4098,4155,4156,4158,4169,4170,4189,4200,4211,4219,4223,4240,4242,4247,425
		1}
	3	{41,113,115,116,144,145,152,173,191,224,226,3104,3432,3434,3437,3443,3557,3572,35
		84,3585,3879,3979,3986,4046,4047,6723,6735,6737,6759,6811,6878,6885}
	4	{16,266,275,298,2943,2944,2996,2997,3048,3056,3057,3058,3115,3533,4123,4127,4157
		,4173,4199,4225,4227,4261,6261,6729,6755,6757,6852,6856,6864,6886,7151,7647}
5		{2913,2914,2920,2937,2940,3002,3003,3004,3005,3006,3007,3010,3053,3071,3093,349
	5	7,4967,5166,5168,5227,6251,6257,6260,6264,6270,6275,6282,6284,6313,6324,6342,765
		4}
		{2916,2938,2999,3000,3001,3091,6247,6250,6256,6266,6268,6273,6277,6283,6291,629
	6	2,6293,6296,6297,6299,6304,6305,6309,6310,6337,6341,6363,6373,6385,6393,6403,641
		2}
		{3433,3448,3453,3461,3467,3534,3552,3586,3877,3888,3905,3915,3923,3924,3925,392
	7	9,3930,3944,3946,3948,3953,3995,4007,4009,4011,4045,6754,6785,6809,6859,6876,716
		7}
		{1768,1770,1774,1781,1788,1789,1793,1794,1795,1797,1801,1802,1803,1804,1805,180
	2	6,1807,1808,1826,1869,1872,1882,1889,1897,1899,1901,1902,1903,1904,1910,1944,487
		3}
		{1737,1741,1742,1743,1748,1749,1750,1767,1771,1785,1854,1855,1858,1860,1861,186
	3	6,1879,1896,1900,1912,1931,1942,3196,3203,3248,3285,3316,4633,4637,4662,4667,470
		6}
		{1759,1823,1863,1873,1894,1934,1936,3138,3150,3184,3185,3195,3197,3198,3199,320
	4	0,3201,3202,3237,3247,3249,3250,3269,3272,3274,3287,3293,3306,3307,3342,4666,467
		2}
6		{1724,1744,1745,1746,1747,1764,1810,1811,1816,1817,1818,1856,1857,1884,1895,190
	5	7,1930,1933,3132,3134,3220,3271,3273,3292,4212,5029,5088,5191,5206,5225,5229,523
		2}
		{1008,1751,1773,1815,1819,1820,1822,1827,1828,1867,1878,1906,1913,1935,4165,464
	6	6,4677,4705,4856,4868,4886,5026,5085,5100,5114,5128,5145,5177,5189,5199,5204,525
		0}
		{3141,3143,3180,3186,3187,3189,3190,3193,3194,3217,3223,3224,3226,3228,3231,323
	7	3,3257,3261,3278,3282,3283,3286,3290,3305,3308,3309,3325,3334,3338,4659,4673,468
		4}
7	2	{3,250,255,1701,2248,2255,2266,2274,2282,2295,2310,2314,2315,2349,2377,2378,2380
		,2384,2396,2409,5416,5420,6491,6496,6497,6503,6505,6506,6511,6514,6702,8088}

		{2213,2214,2225,2240,2241,2262,2268,2269,2297,2299,2304,2317,2318,2336,2337,2342,2346,2357,2386,2387,2390,2399,3397,5493,6490,6501,6507,6510,6513,6697,6698,8090}
	3	{2221,2226,2232,2234,2258,2286,2287,2298,2303,2311,2334,2335,2343,2359,2362,2367,2382,2407,3350,3362,5435,5505,6465,6469,6471,6472,6476,6479,6483,6515,6711,9090}
	4	{5491,5492,5503,6434,6435,6436,6445,6447,6449,6450,6454,6455,6456,6458,6459,6460,6461,6463,6493,6495,6499,6502,6504,6508,6509,6512,6516,6518,6691,6700,6703,8066}
	5	{5410,5412,5417,5422,5426,5428,5437,5440,5449,5454,5456,5472,5473,5475,5477,5478,5479,5484,5486,5487,5488,5498,5499,5500,5502,5556,5569,5577,6486,7671,8087,8111}
	6	{2679,5489,5494,5496,6437,6438,6439,6440,6441,6442,6443,6444,6446,6451,6452,6485,6487,6488,6489,6492,6494,6498,6500,6517,6532,6534,6692,6699,6701,6704,6705,6710}
	7	{1945,1946,1976,1985,2006,2014,2019,2022,2054,2060,2078,2080,2091,2101,2102,2109,2158,2171,2172,2181,5839,5840,5841,5842,5874,5910,5920,5921,5923,6029,6037,7150}
	2	{1961,1962,1974,1975,1981,1983,1989,2010,2011,2030,2034,2035,2036,2040,2043,2063,2065,2068,2076,2083,2089,2090,2094,2097,2104,2105,2106,2108,2131,2148,2176,2184}
	3	{1947,1948,1949,1999,2020,2021,2061,2110,2146,2151,2155,2167,3398,5843,5844,5845,5846,5895,5896,5903,5909,5911,5925,5926,5931,5936,5958,5964,5981,5987,6013,6030}
8	4	{1950,1953,1954,1977,1992,1997,1998,2001,2002,2012,2013,2057,2058,2059,2064,2066,2067,2069,2073,2077,2079,2081,2099,2129,2130,2134,2139,2153,2160,2161,2162,2166}
	5	{1951,1952,1955,1956,1988,1996,2008,2016,2023,2031,2046,2049,2050,2053,2055,2056,2062,2075,2087,2088,2098,2103,2116,2128,2136,2142,2152,2163,2192,2194,3371,5930}
	6	{2052,2093,2107,4764,4770,4771,4780,4781,4782,4783,4784,4786,4788,4789,4796,4802,4808,4809,4814,4829,4830,4831,4835,4836,4857,4863,4867,4907,4908,4910,4915,4917}
	7	{235,239,240,243,248,257,258,2281,2307,2320,2321,2323,2325,2326,2328,2329,2348,2365,2370,2373,2374,2391,2395,2403,2406,5962,7540,7855,7914,7930,7943,7949}
	2	{237,2196,2198,2201,2205,2231,2238,2246,2247,2270,2283,2291,2381,2408,3353,3354,3381,3392,4888,5852,5862,5863,5899,5900,5905,5938,5940,5943,6006,7853,7875,7970}
	3	{245,2195,2217,2249,2364,2366,2372,3361,5860,5861,5939,5946,5956,5957,5960,5961,6020,6021,6022,6023,6024,6025,6026,6035,7974,8056,9469,9479,9485,9490,9521,9522}
	4	{246,247,3111,3117,7887,7890,7891,7892,7893,7894,7895,7896,7897,7905,7913,7928,7929,7942,7956,7957,7975,7977,8052,8053,9462,9463,9484,9486,9498,9500,9516,9517}
9	5	{2215,2216,2218,2219,2222,2223,2224,2227,2235,2236,2239,2243,2245,2257,2273,2296,2308,2309,2312,2316,2344,2369,2376,2393,2404,2411,3349,3373,3374,3402,3403,4608}
	6	{3109,3112,3125,3131,5929,5944,6063,7173,7837,7850,7851,7852,7858,7862,7877,7889,7900,7904,7906,7907,7908,7909,7910,7911,7912,7915,7922,7925,7968,7969,9510,9524}
	7	{2206,2207,2208,2209,2210,2211,2212,2233,2263,2264,2265,2271,2288,2302,2333,2350,2351,2354,2375,2383,2392,2400,2412,3376,3379,3419,3422,3424,3425,3427,6615,6625}
10	2	{253,254,849,2202,2203,2228,2230,2237,2242,2256,2261,2289,2290,2292,2300,2301,2313,2341,2345,2353,2361,2379,3345,3347,3348,3382,3383,3386,3407,3413,3420,4975}
	3	

		{1517,1518,1561,1562,1563,1564,1565,1566,1570,1587,1589,1595,1599,1617,1629,1633,1639,1640,1656,1662,1664,1672,1673,1686,1698,4629,4648,4649,4674,4696,4699,5175}
		{1546,1547,1548,1569,1572,1575,1591,1618,1634,1663,1674,3346,3358,3367,3375,3390,3412,5153,5857,5858,5859,5869,5870,5873,5891,5902,5904,5906,5932,6008,6016,6040}
		{2197,2199,2200,2204,2244,2251,2253,2254,2267,2280,2284,2294,2305,2355,2360,2371,2385,2394,2398,2405,3352,3355,3365,3368,3378,3384,3385,3391,3400,3405,3416,7182}
		{3359,3387,3394,3401,3409,3417,3421,5000,5847,5865,5871,5877,5885,5886,5887,5888,5889,5890,5892,5893,5894,5897,5915,5934,5941,5954,5965,5966,5985,6011,6034,6039}
		{2169,2177,2178,2179,2182,2714,2731,2739,2768,2798,2812,2837,2840,2850,2901,2907,2908,2912,4719,4752,4787,4807,4828,5878,5879,5972,5973,5991,7025,7149,7160,7166}
		{1969,1972,1973,2032,2038,2039,2044,2086,2117,2119,2127,2727,2734,2736,2743,2745,2753,2777,2810,2824,2864,2871,2872,2896,2902,2903,2904,2905,5838,7024,7027,7155}
		{1963,1994,2017,2018,2113,2124,2190,2729,2767,2797,2800,2801,2829,2866,2869,2870,2884,2888,5837,5880,5881,5882,5967,5968,5969,5971,5974,5975,5976,5989,5990,6002}
<b>11</b>		{2715,2716,2719,2720,2724,2725,2728,2732,2733,2737,2738,2742,2744,2746,2752,2759,2763,2772,2779,2781,2782,2787,2795,2796,2831,2833,2841,2845,2846,2855,2856,2897}
		{2721,2723,2740,2747,2751,2755,2765,2771,2793,2799,2802,2803,2805,2806,2808,2809,2811,2814,2818,2819,2838,2839,2851,2852,2853,2857,2865,2877,2880,2886,2895,2899}
		{2000,2092,2120,2121,2122,2123,2125,2156,2180,2183,2185,2191,2820,2832,2867,2881,2887,3019,3025,3129,3344,3369,4894,5883,5901,5916,5928,5996,5999,6000,6001,7134}
		{2921,2922,2924,2942,2949,2952,2955,2974,3015,3016,3036,3065,3077,3094,3098,6248,6254,6265,6298,6314,6332,6345,6347,6348,6370,6384,6405,6420,6720,6731,6732,7128}
		{3008,6252,6263,6267,6311,6319,6330,6352,6369,6371,6375,6376,6380,6382,6394,6395,6424,6427,6716,6724,6728,6733,6738,6739,6756,6788,6789,6842,6843,6877,7143,7156}
		{4153,4160,4257,4258,4263,4264,4269,4652,4679,5167,6249,6294,6317,6318,6321,6327,6336,6349,6351,6359,6374,6386,6387,6389,6398,6401,6404,6408,6418,7152,7171,7179}
<b>12</b>		{4250,4254,6271,6272,6279,6281,6285,6286,6290,6295,6300,6308,6316,6320,6323,6329,6331,6335,6339,6344,6346,6362,6377,6379,6388,6390,6397,6399,6400,6415,6416,6417}
		{4936,5170,5259,6253,6255,6258,6274,6276,6301,6302,6307,6312,6340,6350,6364,6365,6367,6378,6392,6407,6409,6414,6421,6422,7133,7162,7644,7649,7655,7656,7657,7667}
		{5094,5169,5171,6259,6262,6269,6278,6280,6288,6289,6303,6306,6315,6325,6326,6328,6333,6334,6343,6353,6354,6357,6361,6366,6368,6372,6381,6391,6396,6402,6406,7653}
		{277,288,1593,1729,1758,1766,1784,1812,1813,1814,1821,1824,1850,1892,1905,2478,3121,3575,3579,4765,4792,4840,4848,6287,6974,6981,7032,7035,7650,8014,8064,8077}
<b>13</b>		{1730,4716,4730,4744,4753,4755,4756,4757,4758,4760,4761,4766,4768,4794,4800,4841,4847,4859,4861,4862,4865,4870,4876,4877,4878,4879,4889,4891,4900,4901,4914,4918}

		{2932,2939,2941,2946,2951,2958,2959,2964,2967,2969,2984,2986,2989,2995,3021,3022,3026,3035,3038,3043,3045,3051,3060,3069,3072,3097,3101,3103,4591,5019,7127,7137}
	4	{2915,2919,2925,2926,2927,2928,2929,2930,2931,2933,2934,2948,2953,2957,2963,2998,3013,3046,3061,3067,3075,3083,3100,4203,4266,4610,4678,7130,7146,7169,7170,7648}
	5	{2936,2960,2961,2965,2970,2971,2972,2975,2976,2978,2993,3011,3017,3018,3020,3023,3028,3029,3033,3037,3039,3041,3044,3066,3074,3078,3082,3090,3096,3105,3124,4675}
	6	{2945,2950,2954,2966,2968,2973,2977,2979,2980,2981,2982,2983,2985,2987,2988,3009,3012,3024,3032,3040,3042,3054,3063,3064,3079,3080,3084,3085,3095,4104,4194,4239}
	7	{323,325,339,353,354,358,369,374,375,377,379,383,387,388,393,409,410,413,436,437,459,2418,2420,2421,2422,2425,3106,3896,3960,3971,7645,7669}
	2	{337,338,344,345,376,378,407,408,418,427,434,463,3880,3881,3883,3885,3889,3900,3906,3909,3914,3919,3932,3938,3947,3962,3966,3974,3991,4022,4064,4286}
	3	{3878,3890,3892,3895,3901,3911,3916,3922,3927,3928,3937,3940,3951,3963,3969,3973,3978,3985,3990,3996,3999,4008,4016,4021,4025,4033,4058,4060,4061,4062,4063,4065}
	4	{3050,3052,3087,5017,6018,6338,6356,6360,6411,6419,6423,6426,6717,6734,6745,6767,6768,6773,6787,6801,6819,6821,6822,6829,6835,6836,6837,6853,6866,6868,6870,6889}
	5	{3086,6355,6383,6410,6413,6712,6713,6719,6721,6722,6725,6726,6727,6736,6741,6742,6761,6770,6790,6791,6792,6793,6820,6830,6834,6855,6867,6872,6873,6881,6882,6888}
	6	{3127,3499,3500,3882,3893,3898,3933,3934,3952,3956,3972,3988,3997,4010,4013,4017,4031,4043,4048,4054,6715,6718,6752,6753,6772,6786,6807,6810,6844,6851,7142,7178}
	7	{304,305,306,315,322,328,329,330,334,341,347,348,363,364,386,411,421,441,442,443,444,445,451,452,453,462,2464,3517,4297,4309,4326,7661}
	2	{316,317,318,319,320,352,359,360,361,362,366,373,381,382,392,414,3459,3464,3520,3522,3527,3536,4281,4285,4302,4331,4335,4336,4337,4338,4363,4390}
	3	{321,332,343,346,356,415,450,456,457,461,2443,2466,2468,2471,2481,3501,3535,3539,3558,3894,3897,3917,3945,3955,3957,4003,4005,4032,4053,4056,7174,7652}
	4	{3480,3608,3876,3884,3891,3902,3908,3913,3921,3931,3935,3949,3961,3975,3981,3983,3989,4000,4001,4004,4026,4038,4039,4052,4059,4319,4366,4367,4398,4403,4405,4415}
14	5	{2452,3886,3887,3904,3907,3910,3912,3926,3936,3939,3941,3942,3950,3959,3964,3965,3976,3977,3987,3992,3993,3994,4002,4012,4014,4015,4019,4020,4035,4037,4040,4057}
	6	{2447,3515,3903,3943,3954,3958,3967,3982,3984,4006,4284,4287,4289,4292,4296,4312,4320,4322,4327,4328,4329,4330,4332,4333,4334,4364,4371,4372,4374,4376,4394,4414}
	7	{303,307,309,310,312,313,324,326,327,331,335,336,350,351,370,371,401,403,404,416,420,430,431,432,439,446,447,448,455,458,2417,2453}
	2	{308,357,365,372,380,384,385,389,390,391,394,395,402,412,417,422,423,424,425,426,428,435,440,449,460,2423,2437,2449,7646,7658,7662,7664}
	3	{311,314,333,340,342,349,355,367,368,396,397,398,399,400,405,406,419,429,433,438,454,2446,7938,7939,8020,8027,8028,8029,8030,8031,8032,8040}
	4	{6094,6095,6096,6097,6099,6104,6131,6132,6133,6134,6135,6136,6137,6138,6139,6143,6161,6175,6176,6177,6183,6185,6188,6189,6191,6192,6203,6204,6210,6226,6227,6236}
	5	

		{6100,6101,6102,6103,6144,6145,6146,6179,6180,6193,6194,6195,6208,6209,6219,6234,6235,6238,6243,6244,6246,7940,7993,7994,7995,7998,7999,8000,8001,8002,8035,8039}
		{6163,6164,6165,6166,6167,6168,6169,6170,6171,6178,6211,6212,6213,6214,6215,6216,6228,6239,6240,7978,7979,7996,7997,8003,8023,8024,8025,8033,8034,8041,8042,8051}
17	2	{470,497,512,513,518,519,526,528,531,535,544,546,547,552,555,556,565,566,588,592,595,596,600,601,613,614,620,621,625,637,8096,8109}
	3	{493,502,539,557,558,560,561,563,573,574,575,576,577,583,587,590,591,593,609,610,623,624,632,648,974,979,982,983,987,1013,8104,8116}
	4	{474,475,480,486,491,492,510,527,536,537,540,542,553,554,559,564,567,578,585,586,597,598,599,615,616,617,618,619,629,633,635,647}
	5	{464,465,472,478,496,498,499,503,504,505,506,514,523,525,532,548,549,550,551,562,584,589,611,612,622,628,631,636,638,640,641,642}
	6	{937,939,940,941,945,947,955,959,967,972,975,993,1016,1023,1025,1049,1069,1070,1078,2527,2551,2552,2610,2617,2628,5011,5334,5345,5359,5363,5382,8106}
	7	{986,989,997,1001,1006,1010,1011,1012,1022,1027,1028,1029,1043,1044,1045,1046,1047,1050,1073,1075,1076,1077,1082,1087,7147,8100,8101,8103,8110,9043,9044,9046}
	2	{928,929,936,944,958,991,992,998,1014,1030,1048,1057,1085,1161,1167,1170,1174,1176,1185,1186,1187,1196,1204,1207,1245,1270,1298,1299,1322,2585,2709,2710}
3	{930,931,933,935,943,952,964,965,969,981,984,985,988,994,995,1019,1038,1040,1041,1042,1061,1062,1063,1072,2600,2605,4420,4428,4509,4535,4567,8074}	
4	{1177,1180,1188,1190,1193,1195,1198,1199,1200,1202,1203,1205,1206,1208,1210,1213,1216,1217,1218,1219,1226,1230,1232,1233,1234,1235,1236,1239,1269,1311,1312,1320}	
18	5	{1162,1163,1164,1165,1166,1168,1169,1172,1178,1209,1211,1212,1214,1215,1220,1221,1222,1223,1224,1238,1276,1277,1278,1280,1294,1295,1296,1301,1303,1315,1511,5306}
	6	{1179,1181,1182,1183,1184,1189,1191,1192,1194,1242,1251,1257,1266,1273,2489,2490,2491,2492,2493,2494,2495,2496,2497,2498,2499,2500,2547,2556,2582,2584,2701,8063}
	7	{1197,1201,1237,1241,1243,1244,1247,1248,1252,1255,1256,1259,1260,1263,1264,1265,1267,1313,2586,2589,2599,2604,2607,2609,2706,4426,4435,4439,4440,4453,4456,4570}
	2	{1103,1104,1106,1107,1109,1111,1112,1113,1114,1115,1116,1117,1118,1119,1120,1121,1125,1127,1129,1135,1136,1137,1139,1143,1144,1148,1149,1150,1157,1159,1175,4443}
19	3	{466,467,468,469,471,481,538,932,934,951,953,970,1002,1021,1051,1068,1080,1086,1089,1090,1091,1092,1094,1095,1096,1097,1099,1100,1102,1105,4445,4450}
	4	{477,482,484,485,487,488,494,495,511,515,516,517,522,524,529,533,534,571,604,605,606,607,608,630,634,639,646,1093,4992,4998,5001,7665}
	5	{572,626,1318,4424,4430,4431,4432,4433,4434,4436,4437,4438,4441,4442,4444,4446,4447,4448,4449,4451,4452,4454,4455,4531,4559,4560,4562,4571,6641,6683,6689,8105}
	6	{1122,1123,1124,1126,1128,1130,1131,1132,1133,1134,1138,1140,1141,1142,1145,1146,1147,1151,1152,1153,1154,1155,1156,1158,1160,1171,1173,1271,1274,1275,1297,1309}
	7	{473,476,479,483,489,500,501,508,520,521,541,545,568,569,570,582,602,603,627,643,644,645,942,948,954,999,1000,1017,1026,1081,4701,5018}
20	2	{507,509,543,579,580,581,4972,4989,7651,8824,8838,8839,8841,8843,8850,8854,8863,8864,8866,8869,8875,8876,8894,8899,8946,8954,8976,9023,9060,9064,9117,9257}
	3	{490,530,594,4997,5008,5020,8842,8847,8877,8878,8885,8892,8905,8909,8914,8949,8958,8960,8961,8962,8978,8994,8998,9027,9032,9078,9118,9178,9495,9501,9533,9536}



		{4690,8825,8853,8859,8860,8861,8865,8870,8871,8904,8910,8935,8942,8943,8945,896 4 5,9013,9014,9017,9022,9025,9061,9079,9175,9180,9235,9237,9264,9290,9291,9491,953 1}
		{4940,4942,4943,4948,4979,4980,4981,4982,4983,4984,4985,4986,4987,4988,4990,499 5 1,4993,5002,5003,5004,5006,5007,5012,5013,5015,5016,5022,5023,8927,9005,9058,915 9}
		{3778,6595,6645,6648,6650,6651,6653,6655,6656,6658,6661,6663,6664,6666,6668,668 6 0,7934,8913,9116,9130,9131,9183,9187,9188,9189,9190,9191,9202,9223,9255,9281,928 2}
		{3871,7666,7832,7976,9105,9108,9115,9124,9133,9136,9138,9139,9144,9146,9151,917 7 0,9179,9186,9194,9199,9214,9231,9238,9240,9266,9267,9279,9283,9285,9295,9296,949 6}
		{1225,1227,1228,1229,1231,1268,1283,1286,1290,1291,1292,1293,1302,1305,1306,131 2 0,1319,1321,5299,5313,5315,5319,5332,5361,5364,5397,5401,5402,5515,5558,5563,643 0}
		{1240,1246,1249,1250,1253,1254,1258,1261,1262,1316,2501,2502,2503,2504,2505,250 3 6,2507,2516,2560,2569,2583,2591,2592,2705,4427,5267,5346,8075,8089,8108,8112,811 5}
		{2488,2531,2533,2535,2537,2541,2542,2544,2545,2561,2564,2566,2568,2571,2572,257 4 3,2574,2576,2577,2581,2687,2688,2693,2712,5272,6429,6684,6686,6687,8061,8092,809 7}
21		{2508,2509,2510,2511,2512,2513,2514,2515,2517,2518,2519,2520,2521,2522,2523,252 5 4,2525,2526,2528,2529,2532,2538,2539,2543,2546,2548,2549,2550,2553,2555,2557,811 4}
		{2530,2534,2536,2540,2554,2558,2559,2587,2588,2590,2593,2594,2595,2596,2597,259 6 8,2601,2602,2603,2606,2608,2707,2708,4425,4429,6670,6671,6673,6678,6706,8069,808 4}
		{2680,2681,2682,2683,2684,2685,2686,2689,2690,2691,2692,2694,2695,2696,2697,269 7 8,2699,2700,2702,2703,2704,2711,6521,6643,6646,6667,6677,6679,6681,6685,6694,670 7}
		{5703,5714,5758,5759,5801,5802,5805,5815,5822,5824,5825,5826,7139,7158,7216,721 2 7,7222,7223,7229,7235,7239,7241,7246,7251,7307,7309,7315,8639,8647,8682,8740,879 3}
		{1384,1423,1426,1453,1466,1487,8643,8646,8656,8657,8658,8673,8678,8681,8683,868 3 7,8688,8689,8692,8695,8708,8709,8725,8736,8750,8751,8778,8788,8795,8810,8811,881 2}
		{3363,3366,3380,3388,3428,3429,3817,5848,5849,5850,5851,5875,5898,5912,5918,593 4 3,5949,5988,6009,6014,6028,7359,8050,8672,8706,8710,8711,8729,8731,8754,8796,880 9}
22		{699,5581,5582,5584,5590,5613,5614,5615,5616,5617,5618,5620,5682,5687,5688,5700, 5 5748,5749,5756,5757,5761,5765,5806,5810,5833,8637,8653,8690,8699,8742,8761,8779 }
		{5736,8630,8631,8640,8642,8645,8676,8679,8684,8696,8697,8713,8721,8722,8730,873 6 9,8747,8755,8756,8759,8760,8767,8768,8770,8774,8784,8790,8797,8799,8800,8816,881 8}
		{4941,4966,7240,7342,7367,8611,8615,8628,8629,8632,8634,8636,8638,8641,8655,865 7 9,8662,8666,8675,8717,8738,8741,8746,8749,8752,8753,8765,8791,8792,8798,8801,881 5}
		{652,654,661,693,696,715,726,735,737,742,751,780,819,827,837,847,860,867,869,871,8 2 75,876,877,878,879,880,888,890,893,918,1396,6955}
		{685,710,761,781,787,788,794,800,809,826,831,864,866,882,883,887,889,894,3618,362 3 0,3624,3626,3627,3667,3670,3708,3751,3796,3836,3860,5010,7051}
23		{673,706,708,720,724,746,747,762,807,822,844,845,848,900,902,904,905,906,911,914,9 4 20,921,922,923,6957,6959,7046,7094,8827,9148,9526,9528}

		{896,3625,3671,3682,3685,3688,3691,3694,3712,3719,3733,3749,3760,3793,3798,3800,3802,3804,3807,3808,3813,3815,3826,3830,3847,3848,3858,3859,5224,8972,9030,9534}
	5	}
	6	{662,663,717,718,745,765,782,792,874,892,898,916,919,924,926,1326,1388,1428,1444,1447,1448,1455,1467,1479,1490,1498,1501,4938,4946,4955,6961,9508}
	7	{3619,3623,3629,3630,3649,3661,3674,3681,3686,3706,3709,3721,3730,3805,3814,3820,3827,3828,3832,4935,7093,7096,8071,8073,9154,9182,9210,9227,9457,9477,9480,9511}
24	2	{649,741,805,834,835,1506,7226,8612,8613,8614,8616,8617,8619,8621,8622,8624,8651,8694,8698,8700,8703,8716,8720,8724,8732,8733,8734,8735,8737,8748,8771,8776}
	3	{658,660,664,666,672,674,686,711,714,740,775,784,797,801,802,812,852,855,903,927,1387,1474,4952,4974,7299,8623,8769,8775,8783,8789,8805,8806}
	4	{651,653,655,668,684,698,700,707,709,725,727,734,755,767,776,779,783,791,808,815,824,825,828,832,842,850,854,861,912,915,2848,3728}
	5	{659,677,680,682,688,695,701,702,704,728,729,730,736,743,752,768,772,773,777,789,798,816,820,829,836,841,853,857,863,881,884,7390}
	6	{656,665,669,705,744,748,763,769,774,796,804,814,830,858,3621,7220,7221,7231,7238,7245,7250,7252,7272,7292,7352,7358,7401,8618,8620,8702,8744,8766}
	7	{667,670,676,679,690,692,697,703,712,722,723,733,750,753,756,757,766,771,786,790,795,817,818,885,899,917,7189,7191,7206,7291,7357,7396}
25	2	{1334,1360,1365,1366,1368,1378,1380,1382,1385,1411,1449,1471,1472,1480,1482,1483,1484,1485,1486,1491,1494,1504,1505,1510,1514,1515,2450,6960,7088,9164,9487,9506}
	3	{716,891,1329,1335,1340,1350,1374,1381,1386,1391,1392,1404,1405,1406,1425,1432,1433,1435,1436,1452,1457,1460,1478,1488,1492,1502,1503,1507,1508,4969,7047,7099}
	4	{873,897,913,1323,1324,1325,1327,1331,1336,1339,1346,1355,1362,1363,1367,1407,1408,1414,1418,1419,1446,1454,1458,1461,1465,1469,1476,1499,1509,4950,4976,7159}
	5	{1330,1333,1349,1354,1357,1358,1364,1373,1377,1389,1403,1416,1424,1429,1434,1437,1438,1473,1495,2854,4939,4994,7860,7886,7898,7917,7920,7921,7955,8018,9466,9518}
	6	{1328,1342,1345,1359,1372,1375,1379,1398,1401,1402,1415,1417,1420,1431,1442,1443,1445,5005,7841,7844,7846,7849,7859,7867,7868,7870,7881,7935,7937,7972,8013,8822}
	7	{1464,3122,3123,3126,4626,4937,7668,7670,7838,7839,7842,7845,7863,7865,7866,7878,7879,7883,7899,7916,7923,7924,7926,7931,7933,7944,7946,7950,7973,8006,8019,8021}
26	2	{294,1630,1635,1650,1654,1714,1738,1777,1779,1790,1791,1792,1796,1832,1852,1871,1874,1888,1890,1920,3136,3144,3145,3154,3155,3179,3191,3204,3211,3213,3230,3235}
	3	{938,946,950,957,960,963,977,990,1007,1009,1018,1020,1031,1032,1037,1039,1052,1053,1054,1055,1056,1064,1065,1071,1079,1088,6708,8065,8083,8091,8093,8098}
	4	{973,976,1005,1035,1036,1058,1059,1066,1067,1074,1083,4422,4472,4475,4480,4485,4495,4513,4529,4547,4550,4553,4554,4561,4584,4587,4588,7643,7663,8081,8082,8107}
	5	{272,286,949,961,962,966,978,996,1003,1004,1033,1034,1535,1536,1537,1538,1540,1655,1704,3259,4423,4463,4494,4518,4519,4536,4537,4552,4555,4564,4572,8095}
	6	{1614,1732,1734,1735,1752,1760,1765,1778,1837,1839,1843,1844,1845,1846,1847,1848,1851,1853,1859,1881,1915,1916,1928,1929,1932,1939,3142,3178,3183,3210,3212,3332}
	7	{956,968,971,980,1015,1024,1060,1084,4421,4473,4474,4479,4483,4484,4492,4500,4508,4512,4522,4534,4556,4565,4566,4568,4575,4576,4577,4580,4581,7145,8076,8079}
27	2	{1708,3151,3152,3153,3156,3157,3158,3161,3163,3165,3166,3168,3169,3170,3173,3205,3214,3215,3216,3219,3221,3245,3251,3252,3276,3289,3313,3321,3329,3331,3335,3336}

	3	{262,1731,1733,1736,1739,1740,1753,1757,1769,1776,1783,1838,1840,1842,1876,1877,1922,1924,1937,1940,1941,1943,3174,3177,3209,3229,3255,3323,4645,4671,4686,4962}
	4	{53,89,217,1519,1521,1522,1523,1533,1534,1585,1586,1592,1604,1615,1622,1623,1631,1638,1641,1658,1659,1671,1677,1694,1695,1710,2174,3135,3139,3140,3164,3225}
	5	{274,296,1272,1279,1281,1282,1284,1285,1287,1288,1300,1304,1307,1308,1314,1317,1539,1608,1636,1678,1715,3330,4700,5321,5390,5517,5518,5521,5557,5568,5575,8094}
	6	{1524,1551,1556,1557,1560,1602,1665,1667,1679,1680,1688,1692,1727,1728,1756,1780,1809,1841,1862,1865,1911,1921,1925,3133,3172,3175,3176,3181,3207,3337,4661,4670}
	7	{3137,3146,3147,3148,3149,3159,3160,3162,3167,3171,3188,3206,3208,3218,3222,3227,3232,3234,3238,3244,3256,3258,3263,3266,3267,3268,3270,3315,3318,3319,3341,4688}
	2	{1289,1544,1578,1637,1645,1685,1720,1721,1723,1754,1762,1775,1833,1834,1835,1908,1918,1919,1923,2095,2173,3343,3389,3411,3414,4635,4656,4804,4885,4892,4893,4895}
	3	{1718,1719,1722,1726,1763,1786,1787,1798,1799,1800,1825,1830,1864,1868,1870,1875,1880,1885,1886,1887,1893,1898,1917,1926,1927,1938,4791,4849,4852,4890,4898,4930}
	4	{1755,1761,1829,1831,1891,1909,3182,4721,4726,4731,4732,4733,4735,4737,4739,4741,4742,4743,4747,4822,4825,4827,4860,4866,4875,4896,4903,4911,4920,4923,4929,4933}
<b>28</b>	5	{1883,4717,4720,4722,4724,4738,4740,4748,4749,4795,4816,4823,4832,4833,4846,4850,4851,4854,4855,4858,4864,4869,4871,4872,4880,4881,4883,4884,4887,4919,4925,8246}
	6	{1970,1971,1990,1995,2009,2029,2071,2072,2074,2111,2112,2115,2132,2145,2147,2154,2164,2168,2170,2189,3370,3396,3399,3404,3406,3426,4715,4734,4826,4905,5950,7857}
	7	{1849,3192,4729,4745,4750,4751,4754,4759,4763,4767,4785,4790,4793,4797,4798,4799,4810,4815,4818,4819,4821,4834,4843,4844,4845,4909,4912,4913,4922,4928,4932,7164}
	2	{231,1098,1101,1108,1110,2575,2677,2678,4569,5268,5340,5343,5344,5347,5351,5360,5367,5371,5376,5377,5378,5384,5389,5400,5404,5476,5485,5495,5497,5510,8067,8072}
	3	{2562,2563,2565,2567,2570,2578,2580,2612,2616,2621,2624,2627,2629,2633,2634,2635,2637,2639,2640,2648,2653,2673,4457,5277,5305,5320,5331,5333,5335,5338,5342,5358}
	4	{2614,2619,2620,2625,2630,2631,2632,2638,2642,2643,2647,2650,2651,2666,2668,2670,2674,5283,5288,5289,5292,5294,5303,5307,5314,5322,5323,5326,5329,5330,5372,5381}
<b>29</b>	5	{2611,2649,2652,2659,2665,2672,2676,2713,5285,5295,5328,5337,5350,5353,5355,5357,5365,5375,5395,5396,5399,5403,5405,5458,5462,5464,5467,5468,5480,5490,5524,5567}
	6	{2579,2613,2641,2644,2645,2646,4502,5284,5287,5290,5291,5297,5298,5300,5308,5324,5325,5327,5336,5339,5348,5349,5352,5362,5394,5514,5516,5520,5559,5560,5562,5571}
	7	{2623,2636,2654,2655,2656,2657,2658,2660,2661,2662,2663,2664,2667,2669,2671,2675,5280,5293,5341,5354,5356,5366,5368,5374,5379,5383,5391,5442,5445,5481,5508,5555}
<b>30</b>	2	{283,3351,3357,3364,3372,3377,3408,3418,5876,5935,5953,5959,5978,6003,6004,6005,7313,8626,8633,8635,8649,8663,8686,8704,8705,8712,8714,8745,8780,8781,8819,9481}

		{1332,1347,1348,1351,1353,1413,1439,1489,4934,7834,7847,7861,7874,7880,7888,793 3 6,7971,8017,8059,8667,8669,8685,8691,8727,8764,8785,8787,8803,8808,8813,8817,882 0}
		{1337,1338,1341,1344,1352,1356,1369,1397,1399,1421,1422,1427,1440,1441,1450,145 4 1,1456,1462,1463,1468,1477,7165,7873,7952,7958,8010,8011,8012,8022,8043,8054,805 8}
		{8625,8627,8644,8648,8650,8652,8654,8660,8664,8665,8668,8674,8677,8680,8693,871 5 8,8723,8728,8743,8757,8758,8762,8763,8773,8777,8782,8786,8794,8802,8804,8807,881 4}
		{3360,3393,5866,5907,5913,5917,5924,5927,5937,5942,5945,5947,5948,5951,5952,596 6 3,5979,5980,5982,5983,5984,6007,6010,6012,6015,6019,6027,6031,6032,6036,7869,915 0}
		{4957,7833,7835,7836,7843,7848,7856,7864,7871,7872,7876,7885,7918,7932,7941,794 7 5,7948,8004,8005,8007,8008,8009,8015,8016,8044,8049,8055,8661,8670,8671,8719,877 2}
	2	{1,2,5,6,7,8,9,10,11,12,14,22,42,46,54,189,195,208,214,284,300,2419,2717,3456,3463,3 465,3475,3478,3481,3486,5151,5231}
	3	{13,15,17,18,19,20,21,23,24,25,26,35,36,37,38,39,43,44,45,49,137,186,2479,2788,2863, 3466,3484,3489,3490,3545,3550,3998}
	4	{182,3116,3118,3130,3471,3506,3508,3542,3549,3599,4459,4468,4481,4486,4491,4496, 4497,4505,4514,4524,4538,4539,4540,4557,4563,4585,4586,6730,6746,6795,6825,6848 }
<b>31</b>	5	{2424,2480,3107,3128,3454,3469,3473,3476,3488,3504,3511,3512,3513,3528,3541,355 1,3581,3603,3604,3609,4027,4300,4354,4355,4356,4357,4359,4396,4397,4404,6823,682 4}
	6	{4464,4465,4466,4467,4469,4470,4476,4477,4482,4487,4488,4489,4498,4499,4503,450 4,4510,4511,4521,4523,4525,4526,4527,4528,4541,4542,4543,4546,4549,4558,4573,457 8}
	7	{55,138,164,2484,4458,4460,4461,4462,4471,4478,4490,4493,4501,4506,4507,4515,451 6,4517,4520,4530,4532,4533,4544,4545,4548,4574,4579,4582,4583,5914,8086,8099}
	2	{4070,4176,4710,4947,5031,5045,5061,5072,5074,5089,5095,5101,5112,5113,5116,511 8,5123,5163,5176,5192,5200,5216,5223,5226,5236,5240,5242,5245,5251,5252,5253,526 0}
	3	{3439,3440,3447,3449,3452,3474,3491,3507,3516,3519,3532,3537,3544,3562,3591,360 6,4276,4278,4291,4299,4304,4305,4306,4314,4318,4324,4350,4351,4361,4368,4375,440 6}
	4	{161,223,3441,3559,3576,3583,5034,5052,5054,5064,5065,5093,5117,5124,5139,5149,5 157,5158,5182,5185,5196,5198,5202,5205,5219,5222,5233,5256,6760,6808,6875,6880}
<b>32</b>	5	{140,171,3457,3487,3496,3510,3514,3529,3530,3531,3560,3561,3578,3582,3587,3589,3 598,3605,3611,3612,4034,4279,4280,4283,4290,4339,4340,4341,4342,4353,4387,4395}
	6	{3438,3495,3498,3505,3509,3546,3548,3563,3564,3565,3577,3588,3593,3594,3595,359 6,3601,3602,3610,4310,4316,4325,4343,4349,4373,4392,4393,4413,4417,5030,5154,679 4}
	7	{2448,3477,3502,3503,3518,3521,3525,3540,3547,3569,3574,3580,3590,3597,3615,429 5,4301,4307,4315,4321,4348,4360,4399,4400,4401,4402,4407,4409,4410,4416,4418,441 9}
	2	{4,104,232,241,251,252,270,280,285,287,291,295,297,299,302,1594,1609,1661,2327,23 52,2401,2402,2414,2415,2416,3356,4630,4692,4711,7641,7882,8062}
	3	{5273,5276,5281,5302,5310,5311,5409,5411,5413,5415,5418,5419,5421,5429,5430,543 2,5433,5434,5436,5438,5439,5441,5470,5525,5526,5527,5528,5529,5530,5564,5565,556 6}
<b>33</b>	4	{2622,2626,5274,5275,5278,5279,5286,5296,5301,5304,5309,5312,5316,5369,5407,541 4,5423,5424,5425,5427,5443,5444,5446,5447,5453,5455,5459,5461,5463,5466,5576,811 3}

		{3492,3493,3523,3526,3555,3566,5271,5408,5431,5448,5451,5452,5457,5469,5471,5474,5482,5483,5504,5506,5507,5543,5544,5551,5552,5553,5554,6431,6432,6462,6474,6480}
5		{2615,2618,3468,3524,5266,5269,5282,5317,5318,5370,5373,5380,5385,5386,5387,5388,5392,5393,5398,5406,5509,5519,5522,5523,5531,5532,5538,5570,5572,5573,5574,6484}
6		{261,276,3571,4620,4708,5264,5265,5270,5450,5460,5465,5501,5511,5512,5513,5533,5534,5535,5536,5537,5539,5540,5541,5542,5545,5546,5547,5548,5549,5550,6433,6466}
7		{244,2277,2319,2324,2332,2340,2347,2363,2397,2843,6578,6579,6581,6582,6583,6584,6585,6586,6590,6591,6592,6596,6597,6598,6599,6600,6601,6605,6606,6612,6693,6709}
2		{234,249,2188,2220,2229,2250,2252,2259,2260,2272,2275,2276,2278,2279,2285,2293,2306,2322,2330,2331,2338,2339,2356,2358,2368,2388,2389,2410,6464,6467,6473,6475}
3		{6448,6453,6457,6468,6470,6477,6478,6481,6482,6520,6523,6524,6525,6526,6527,6530,6535,6537,6603,6604,6607,6608,6609,6610,6611,6613,6617,6622,6626,6696,8078,8085}
4		{6519,6528,6529,6531,6533,6540,6542,6543,6544,6547,6548,6551,6552,6553,6554,6555,6556,6559,6561,6563,6564,6566,6568,6569,6570,6571,6572,6573,6574,6575,6576,6577}
5		{6522,6536,6538,6539,6541,6545,6546,6549,6550,6557,6558,6560,6562,6565,6567,6616,6618,6620,6624,6631,6633,6634,6636,6637,6640,6649,6652,6654,6659,6665,6688,6690}
6		{6580,6587,6588,6589,6593,6594,6602,6614,6619,6621,6623,6627,6628,6629,6630,6632,6635,6638,6639,6642,6644,6647,6657,6660,6662,6669,6672,6674,6675,6676,6682,6695}
7		{657,770,7163,7201,7207,7208,7219,7228,7234,7237,7242,7253,7254,7255,7261,7262,7263,7264,7265,7266,7276,7294,7296,7304,7320,7345,7353,7361,7370,7373,7382,7397}
2		{689,868,7188,7190,7195,7196,7198,7199,7200,7203,7204,7212,7213,7214,7247,7288,7289,7295,7298,7314,7318,7326,7329,7333,7336,7337,7340,7344,7348,7354,7360,7392}
3		{2890,7244,7257,7259,7260,7278,7284,7285,7303,7316,7321,7322,7325,7332,7338,7341,7343,7346,7347,7365,7378,7387,7389,8283,9299,9394,9402,9404,9415,9417,9424,9456}
4		{4951,4956,5583,5586,5588,5666,5667,5683,5686,5691,5702,5722,5729,5737,5744,5754,5791,5793,7243,7279,7286,7300,7312,7324,7330,7335,7339,7371,7377,7380,7381,7393}
5		{4963,8134,9300,9301,9303,9304,9328,9331,9332,9340,9342,9343,9344,9347,9357,9358,9362,9368,9387,9393,9396,9405,9412,9425,9430,9431,9432,9435,9436,9437,9439,9454}
6		{7187,7192,7193,7194,7205,7209,7210,7211,7236,7258,7268,7277,7283,7287,7290,7297,7305,7306,7308,7310,7311,7319,7327,7334,7351,7355,7356,7364,7366,7376,7383,7398}
7		{4953,5591,5602,5606,5626,5627,5628,6922,7066,7077,7078,7102,7104,7105,7112,7116,7281,7372,8119,8120,8137,8144,8166,8242,8258,8313,8321,8323,8326,8331,8350,8357}
2		{5587,5612,6917,6928,6929,6938,6943,6947,6948,6967,6968,6995,7005,7056,7057,7058,7059,7060,7061,7072,7073,7107,7121,7122,7123,7125,8173,8175,8176,8264,8345,9475}
3		{5600,5601,5629,5630,5631,5642,5646,5663,5671,5674,5675,5678,5684,5694,5707,5716,5717,5731,6896,6899,6918,6942,6985,7020,7068,7079,7081,7083,7084,7087,7117,8243}
4		{851,865,5603,5624,8128,8129,8130,8133,8136,8138,8139,8140,8143,8155,8158,8160,8162,8163,8231,8250,8251,8252,8254,8256,8276,8290,8304,8315,8324,8329,8344,8358}
5		

		{764,5623,5625,7674,7685,7688,7689,7693,7696,7706,7715,7716,7722,7724,7728,7798, 6 7801,7814,7824,7827,8145,8157,8234,8239,8241,8265,8267,8270,8271,8322,8512,8595 }
		{7678,7680,7681,7682,7687,7690,7691,7698,7707,7712,7713,7714,7718,7719,7723,774 7 5,7746,7747,7748,7757,7785,7786,7787,7805,7806,7817,7829,8189,8229,8257,8259,908 3}
	2	{650,671,691,713,731,732,754,760,799,806,823,833,840,843,846,856,870,3622,3628,36 80,3702,3705,3707,3718,3762,3789,3790,3811,3818,3834,3856,9467}
	3	{675,678,681,683,687,694,719,721,738,739,749,758,759,778,785,803,810,811,813,839,8 59,862,872,886,907,908,910,3665,3695,4961,7161,8026}
	4	{3643,3696,3701,3711,3713,3714,3732,3738,3742,3743,3744,3746,3753,3756,3768,379 2,3797,3799,3816,3822,3825,3855,7293,9317,9322,9326,9327,9355,9366,9370,9505,953 0}
37	5	{3645,3764,3835,9308,9312,9315,9316,9318,9321,9323,9324,9325,9329,9330,9334,933 5,9336,9337,9338,9348,9351,9360,9363,9367,9371,9373,9389,9399,9410,9440,9447,945 2}
	6	{3647,9297,9302,9305,9306,9307,9309,9310,9311,9313,9333,9341,9345,9349,9350,935 2,9356,9359,9361,9365,9382,9395,9397,9401,9409,9411,9416,9434,9443,9448,9449,945 1}
	7	{3690,3698,3700,3710,3715,3716,3717,3720,3722,3734,3737,3740,3741,3745,3748,375 4,3763,3771,3785,3786,3821,3823,3829,4954,9353,9354,9427,9428,9429,9433,9445,944 6}
	2	{3108,4965,5592,5609,5611,5659,5660,5664,5669,5697,5699,5706,5708,5723,5733,578 7,5800,5804,5808,5809,5821,7089,8146,8165,8180,8192,8224,8225,8226,8280,8356,871 5}
	3	{3119,5621,7110,7111,8141,8147,8148,8149,8170,8171,8181,8185,8191,8197,8200,820 6,8207,8220,8221,8227,8233,8236,8245,8261,8282,8309,8335,8337,8338,8342,8355,836 2}
	4	{4958,5772,6909,6910,6912,6914,6916,6934,6988,6998,7002,7006,7050,7063,7064,706 5,7080,7113,7126,8202,8208,8209,8210,8211,8212,8219,8228,8293,8318,8319,8346,834 7}
38	5	{5578,5589,5739,5816,6908,6923,7000,7015,7049,7124,8121,8150,8151,8152,8153,815 6,8159,8161,8164,8167,8168,8174,8184,8193,8195,8235,8281,8296,8333,8343,8352,836 1}
	6	{4973,5579,6905,6907,6924,6941,6963,6966,6970,7001,7003,7004,7007,7010,7016,702 1,7036,7040,7041,7048,7067,7070,7082,7106,7114,7115,7120,7157,8118,8154,8214,821 5}
	7	{5608,5610,5665,5668,5681,5693,5698,5704,5710,5786,5792,5834,7218,7224,7225,722 7,7230,7232,7233,7248,7249,7269,7270,7271,7273,7274,7275,7362,7384,7394,7399,773 1}
	2	{793,901,925,1343,1393,1400,1409,1412,1430,1470,1496,1497,1512,4949,4964,4999,69 62,7100,7101,7103,7141,7884,7919,7927,7947,7953,9085,9103,9104,9149,9155,9165}
	3	{3113,3637,3648,3650,3652,3654,3656,3663,3675,3676,3683,3703,3704,3755,3766,376 9,3775,3777,3780,3781,3783,3787,3806,3838,3841,3850,3868,3870,3873,3875,8934,951 5}
	4	{3631,3632,3642,3664,3747,3761,3776,3779,3782,3784,3809,3840,3853,3866,8359,883 2,8844,8922,8936,8939,8940,9009,9029,9034,9062,9065,9075,9076,9461,9465,9509,951 2}
39	5	{8828,8830,8834,8835,8836,8837,8851,8852,8855,8856,8889,8897,8916,8917,8918,891 9,8921,8950,8953,8959,8969,8999,9056,9063,9068,9069,9070,9072,9073,9077,9468,949 9}
	6	{3633,3634,3635,3636,3638,3640,3641,3657,3669,3677,3699,3750,3758,3774,3801,380 3,3810,3837,3842,3845,3849,3851,3852,3854,3863,3864,9319,9320,9339,9414,9497,950 4}

		{8923,8924,8925,8979,8980,8985,8987,8988,8993,9033,9036,9074,9084,9096,9101,9114,9134,9156,9204,9205,9216,9217,9218,9229,9236,9271,9273,9274,9275,9287,9492,9539}
	2	{821,838,895,909,1361,1371,1376,1383,1395,1410,1475,1481,1493,1513,4996,6950,6951,6952,6953,6954,6956,7085,7086,7090,7091,7092,7095,7109,9483,9514,9535,9537}
	3	{3644,3651,3653,3672,3673,3678,3679,3684,3689,3725,3729,3739,3752,3765,3843,3846,3857,3862,3865,3869,8957,8974,8975,9097,9113,9123,9177,9197,9460,9472,9513,9525}
	4	{3655,3660,3666,3668,3687,3692,3693,3697,3723,3724,3726,3727,3731,3735,3736,3757,3767,3772,3773,3788,3791,3794,3795,3812,3819,3824,3831,3833,3867,8973,9523,9527}
<b>40</b>	5	{4970,7132,8080,9087,9089,9092,9093,9111,9112,9120,9121,9122,9125,9126,9127,9128,9147,9158,9162,9163,9176,9185,9192,9209,9219,9245,9253,9292,9459,9474,9529,9532}
	6	{6949,6958,7097,7098,9086,9088,9102,9166,9167,9168,9169,9193,9203,9211,9212,9213,9220,9242,9243,9244,9247,9248,9249,9268,9269,9270,9286,9294,9458,9470,9473,9478}
	7	{3759,9091,9098,9100,9106,9109,9110,9119,9135,9140,9145,9160,9172,9173,9181,9195,9201,9222,9228,9246,9252,9254,9258,9259,9276,9277,9278,9284,9293,9482,9503,9507}
	2	{233,2175,2906,3395,3423,3874,5855,5919,5922,6038,6902,6903,6932,6977,7138,7901,8388,8394,8395,8396,8402,8404,8423,8427,8428,8431,8436,8437,8438,8439,8449,8588}
	3	{1957,1958,1959,1960,1978,1982,1984,2003,2005,2015,2026,2027,2028,2045,2070,2082,2096,2118,2138,2141,2144,2193,5854,5856,5864,5867,5868,5872,5908,5955,5986,6033}
	4	{1964,1965,1966,1968,1991,2007,2033,2041,2048,2051,2114,2140,2149,2157,3415,4712,4714,4718,4723,4725,4736,4769,4803,4812,4817,4824,4882,4897,4904,4921,4926,5997}
<b>41</b>	5	{1967,1979,1980,1986,1987,1993,2004,2024,2025,2037,2042,2047,2084,2085,2100,2126,2133,2135,2137,2143,2150,2159,2165,2186,2187,4968,5853,6913,7129,7153,7902,7903}
	6	{4713,4727,4728,4746,4762,4772,4773,4774,4775,4776,4777,4778,4779,4801,4805,4806,4811,4813,4820,4837,4838,4839,4842,4853,4874,4899,4902,4906,4916,4924,4927,4931}
	7	{6925,6969,8363,8377,8382,8383,8385,8386,8387,8390,8393,8398,8407,8408,8409,8412,8413,8424,8425,8426,8429,8432,8433,8434,8435,8440,8585,8586,8589,8593,8596,8597}
	2	{1370,1390,1394,1459,1500,3861,4960,7954,8060,9099,9107,9141,9153,9157,9161,9171,9174,9184,9196,9198,9200,9221,9224,9239,9241,9250,9251,9289,9471,9489,9502,9519}
	3	{3839,3872,8845,8848,8868,8872,8893,8903,8930,8931,8933,8941,8944,8947,8956,8970,8990,8991,8995,8996,8997,9003,9004,9006,9007,9010,9012,9021,9031,9045,9054,9059}
	4	{3844,8821,8829,8846,8874,8879,8880,8881,8882,8883,8886,8887,8888,8890,8891,8901,8915,8920,8928,8932,8955,8964,8986,9008,9018,9026,9050,9051,9055,9057,9066,9520}
<b>42</b>	5	{8862,8967,8983,9015,9071,9094,9095,9129,9132,9137,9142,9143,9152,9206,9207,9208,9215,9225,9230,9232,9233,9234,9256,9260,9261,9262,9263,9265,9272,9280,9288,9494}
	6	{8823,8826,8849,8857,8858,8867,8896,8898,8902,8906,8907,8908,8912,8938,8948,8951,8952,8966,8968,8984,8992,9002,9016,9019,9035,9037,9039,9040,9041,9042,9049,9080}

		{8831,8833,8840,8873,8884,8895,8900,8911,8926,8929,8937,8963,8971,8977,8981,8982,8989,9000,9001,9011,9020,9024,9028,9038,9047,9048,9052,9053,9067,9081,9082,9226}
	7	{2722,2750,2766,2773,2783,2792,2794,2815,2835,2842,2849,2860,2861,2874,2878,2910,5561,6915,6973,6980,6990,6992,6999,7013,7017,7026,7033,7135,7136,7148,7180,7181}
	2	{2726,2735,2749,2756,2761,2762,2776,2778,2780,2791,2804,2816,2826,2858,2889,2891,2892,2894,2898,2909,2911,3055,5884,6776,6812,6813,6814,6815,6862,6869,6883,7022}
	3	{2741,2754,2758,2760,2764,2769,2770,2774,2775,2784,2785,2786,2789,2790,2807,2813,2817,2821,2825,2827,2828,2836,2844,2859,2868,2875,2879,2883,7029,7031,7034,7154}
43	4	{2923,3027,3030,3034,3047,3062,3089,6425,6764,6774,6778,6780,6781,6783,6784,6804,6805,6806,6816,6833,6838,6839,6847,6854,6857,6860,6861,6871,6874,6890,7175,7176}
	5	{5701,6901,6939,6940,6944,6971,6975,6982,7014,7018,7023,7028,7030,7039,7042,7044,7052,8183,8190,8222,8223,8272,8364,8365,8375,8389,8400,8411,8419,8421,8447,8549}
	6	{7215,7256,7267,7350,7374,7385,7386,7391,7803,7804,7828,7831,7854,8366,8371,8374,8376,8379,8380,8381,8384,8391,8397,8403,8414,8420,8448,8496,8514,8519,8551,8563}
	7	{7700,7701,7702,7703,7704,7705,7710,7711,7721,7759,7761,7795,7807,7820,7821,7823,8186,8188,8196,8260,8446,8468,8469,8470,8472,8497,8515,8516,8517,8518,8580,8584}
	2	{3639,3646,3658,3659,3662,3770,7732,7734,7751,7752,7774,7777,7794,7796,7802,7814,8442,8443,8444,8445,8471,8473,8474,8475,8476,8477,8564,8605,8609,9346,9364,9384}
	3	{5598,5634,5640,5647,5648,5670,5705,5732,5734,5752,5767,5817,5818,7363,7379,7388,7676,7735,7736,7775,7776,8452,9298,9369,9374,9378,9379,9380,9381,9386,9403,9418}
44	4	{5009,5593,5594,5597,5599,5622,5643,5649,5650,5651,5652,5654,5655,5657,5658,5712,5730,5764,5771,5811,5812,5813,5814,5827,5828,7840,8232,8253,8255,8266,8284,8334}
	5	{2900,7673,7684,7686,7717,7725,7726,7729,7737,7738,7739,7740,7741,7742,7743,7744,7749,7750,7753,7754,7756,7781,7792,7793,7797,7799,7815,8070,8238,8240,8286,8287}
	6	{8450,8451,8462,8463,8464,8465,8466,8467,8478,8479,8480,8481,8482,8483,8484,8485,8486,8487,8488,8489,8490,8491,8492,8493,8494,8498,8499,8522,8599,8600,8601,8604}
	7	{4592,4598,4602,4603,4605,4607,4616,4621,4628,6129,6155,6156,6186,7951,7959,7960,7963,7964,7980,7981,7983,7985,7986,7988,7990,7991,7992,8038,8046,8047,8048,8057}
	2	{4623,6068,6114,6115,6116,6117,6118,6119,6120,6121,6122,6125,6126,6127,6128,6174,6190,6218,6231,6237,7537,7961,7962,7965,7966,7967,7982,7984,7987,7989,8036,8037}
	3	{4604,6061,6065,6078,6079,6080,6081,6141,6158,6159,6160,6172,6197,6207,6233,7763,7764,7765,7770,7772,7773,7783,7788,7789,7790,7808,7816,7822,7825,8553,8579,8610}
45	4	{6084,6130,7550,7578,7579,7594,7612,7627,7628,7629,7630,7634,7635,7636,7638,8513,8520,8523,8524,8525,8526,8527,8528,8529,8531,8532,8533,8534,8554,8559,8568,8594}
	5	{6077,6142,6149,6150,6151,6152,6153,6154,6157,6187,6196,6205,6217,6220,6221,6223,6230,6241,7516,7518,7519,7520,7522,7524,7536,7552,7558,7559,7602,7614,7640,8045}
	6	



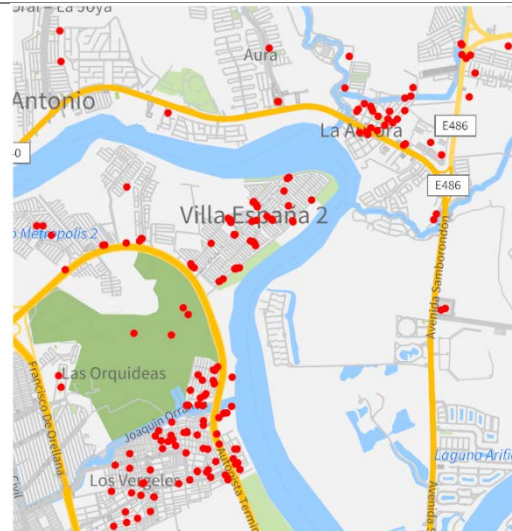
		{4593,4618,6067,6069,6072,6073,6074,6075,6076,6123,6124,6140,6206,6232,7762,776 7 6,7767,7768,7769,7771,7782,7809,7810,8530,8535,8536,8537,8538,8539,8540,8556,860 8}
		{259,2718,2730,2748,2757,2822,2823,2830,2834,2847,2862,2873,2876,2882,2885,4627, 2 5970,5977,5992,5993,5994,5995,6895,6904,6991,6997,7008,7019,7140,7144,8179,8194 }
		{2893,3059,3068,3070,3092,4959,5998,6017,6744,6758,6762,6763,6766,6769,6775,677 3 7,6779,6782,6802,6803,6817,6818,6832,6841,6845,6846,6865,6884,6891,7168,7177,872 6}
		{4589,4596,4609,4611,4615,4617,4622,6894,6897,6965,6993,7009,7692,7720,7730,773 4 3,7779,7813,8367,8368,8369,8370,8378,8392,8410,8415,8416,8417,8418,8422,8441,860 7}
46		{4594,4595,4599,4600,4601,4613,7675,7677,7679,7683,7694,7695,7697,7699,7708,770 5 9,7727,7755,7758,7760,7778,7780,7784,7791,7800,7811,7819,7826,7830,8172,8399,948 8}
		{4971,5619,6906,6911,6920,6926,6927,6933,6964,6976,6978,6979,7043,7071,7119,816 6 9,8177,8178,8230,8244,8263,8273,8294,8300,8327,8328,8336,8339,8340,8341,8349,835 1}
		{5014,6062,6900,6919,6921,6930,6931,6936,6937,6972,6983,6984,6987,6989,7011,703 7 7,7038,7045,7055,7118,7131,8268,8274,8285,8372,8373,8401,8405,8406,8430,8495,870 7}
		{4590,4619,6047,7402,8453,8455,8456,8457,8460,8461,8500,8501,8502,8506,8508,854 2 4,8547,8569,8570,8571,8572,8573,8574,8575,8576,8577,8578,8591,8598,8602,8606,946 4}
		{4597,4614,4624,4625,7418,7464,7465,7466,7468,7472,7474,7475,7478,7480,7482,748 3 8,7512,7534,7547,7548,7565,7592,7593,7596,7598,7599,7605,7606,7609,7616,7617,762 4}
		{4612,7404,7406,7407,7409,7411,7413,7414,7415,7416,7452,7454,7455,7456,7457,745 4 8,7469,7479,7481,7484,7487,7494,7530,7539,7556,7566,7570,7571,7575,7582,7589,824 8}
47		{7460,7461,7462,7467,7470,7471,7476,7477,7489,7490,7491,7500,7501,7502,7503,750 5 4,7505,7509,7510,7511,7525,7526,7538,7541,7549,7551,7560,7567,7597,7600,7604,761 1}
		{7405,7408,7410,7412,7417,7419,7451,7453,7459,7463,7473,7483,7486,7492,7493,749 6 5,7496,7497,7498,7499,7506,7527,7528,7529,7542,7557,7561,7572,7585,7590,7591,760 3}
		{8454,8458,8459,8503,8504,8505,8507,8509,8510,8511,8548,8550,8555,8565,8566,856 7 7,8581,8582,8583,8587,8603,9314,9392,9406,9413,9421,9422,9426,9438,9442,9444,945 0}
		{5580,5604,5605,5607,5633,5636,5637,5644,5645,5656,5695,5696,5709,5721,5738,574 2 0,5742,5747,5751,5755,5768,5770,5785,5788,5789,5796,5797,5798,5799,5829,5836,824 9}
		{5585,6893,6898,6935,6945,6946,6986,6994,6996,7012,7053,7054,7062,7069,7074,707 3 5,7076,7108,8117,8125,8131,8198,8204,8216,8217,8262,8303,8305,8316,8317,8353,836 0}
		{5595,5596,5653,5741,5745,5769,7172,8122,8123,8132,8142,8237,8269,8275,8278,827 4 9,8289,8291,8292,8297,8299,8301,8306,8307,8308,8310,8311,8312,8325,8332,8348,835 4}
48		{5632,5635,5638,5641,5672,5685,5690,5715,5718,5719,5720,5725,5726,5746,5753,577 5 5,5776,5777,5778,5780,5781,5782,5783,5784,5794,5795,5819,5820,5830,5831,5832,730 2}
		{5661,5662,5673,5676,5677,5728,5760,5762,5790,5803,5807,5823,5835,8124,8126,812 6 7,8135,8187,8199,8201,8203,8205,8213,8218,8247,8277,8288,8298,8302,8314,8320,949 3}

		{5639,5679,5680,5689,5692,5711,5713,5727,5735,5743,5763,5766,5779,7183,7184,7185,7186,7197,7202,7280,7282,7301,7317,7323,7328,7331,7349,7368,7369,7375,7395,7400}
<b>49</b>	7	{4606,6041,6043,6048,6049,6050,6051,6052,6053,6054,6057,6059,6060,6064,6066,6070,6082,6083,6098,6108,6109,6110,6111,6148,6162,6181,6224,6225,6245,7812,8295,8330}
	2	{5724,5750,5773,5774,6042,6044,6045,6105,6106,6107,6147,6198,8182,9372,9375,9376,9377,9383,9385,9388,9390,9391,9398,9400,9407,9408,9419,9420,9423,9441,9453,9455}
	3	{6046,6055,6056,6058,6071,6112,6113,6184,6200,6201,6202,6222,7421,7439,7485,7531,7584,7610,8521,8541,8542,8543,8545,8546,8552,8557,8558,8560,8561,8562,8590,8592}
	4	{7403,7423,7425,7426,7432,7436,7437,7438,7440,7513,7514,7515,7521,7532,7533,7535,7543,7546,7563,7564,7573,7580,7581,7586,7587,7607,7608,7618,7619,7620,7621,7622}
	5	{6085,6086,6087,6088,6089,6090,6091,6092,6093,6173,6182,6199,6223,6242,7553,7554,7555,7562,7576,7577,7583,7588,7595,7615,7623,7625,7626,7631,7632,7633,7637,7639}
	6	{7420,7422,7424,7427,7428,7429,7430,7431,7433,7434,7435,7441,7442,7443,7444,7445,7446,7447,7448,7449,7450,7507,7508,7517,7523,7544,7545,7568,7569,7574,7601,7613}
	7	{2426,2427,2431,2432,2438,2444,2455,2456,2457,2458,2459,2460,2467,2472,2482,6796,6797,6826,6827,6831,6849,6892}
	2	{2434,3918,3920,3968,3970,3980,4030,4041,4042,4044,4049,4051,4055,4298,4323,4352,4377,4381,4382,4383,4391,4408}
	3	{2435,2440,2461,2462,2463,2474,2483,2486,3613,3614,3899,4029,4036,4050,4293,4379,4384,4385,4551,6798,6799,6828}
<b>50</b>	4	{3553,3554,3573,3616,3617,4308,4313,4317,4345,4346,4347,4358,6747,6748,6749,6750,6751,6765,6800,6850,6879,6887}
	5	{2428,2441,2445,2451,4018,4023,4288,4294,4303,4311,4344,4362,4365,4369,4370,4378,4380,4386,4388,4389,4411,4412}
	6	{2429,2430,2433,2436,2439,2442,2454,2465,2469,2470,2473,2475,2476,2477,2485,2487,4024,4028,4277,7672,8068}
	7	

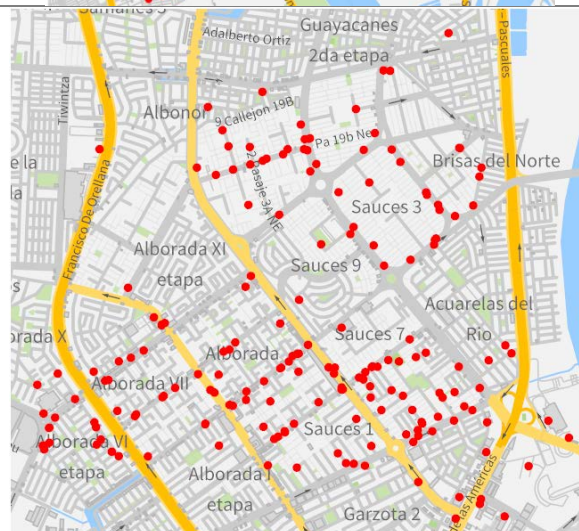
## APÉNDICE B

### Gráficas de clientes por ruta

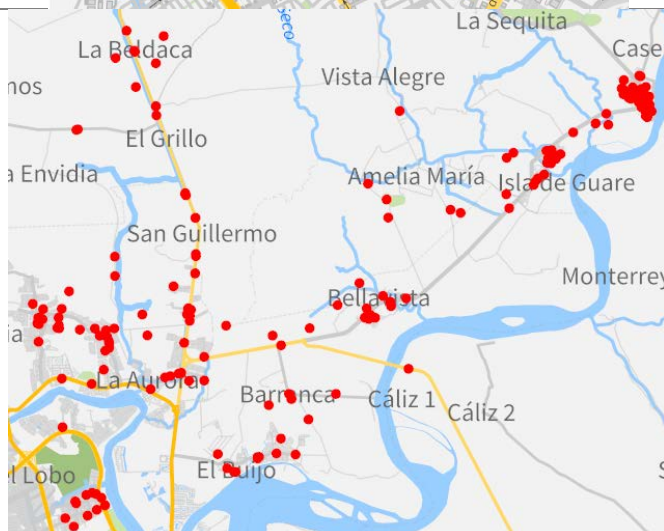
#### Ruta 1



#### Ruta 2



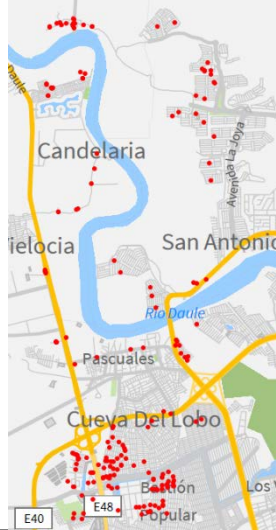
#### Ruta 3



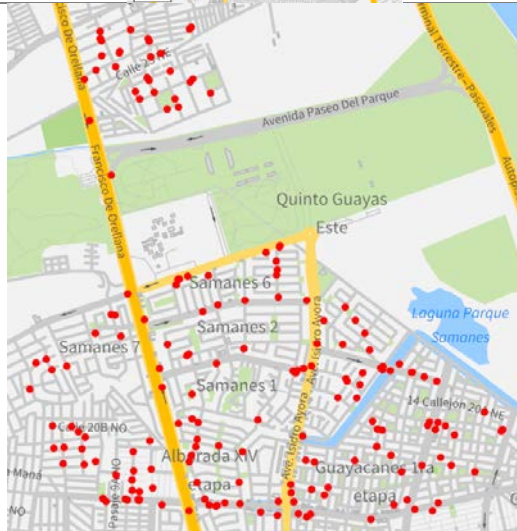
**Ruta 4**



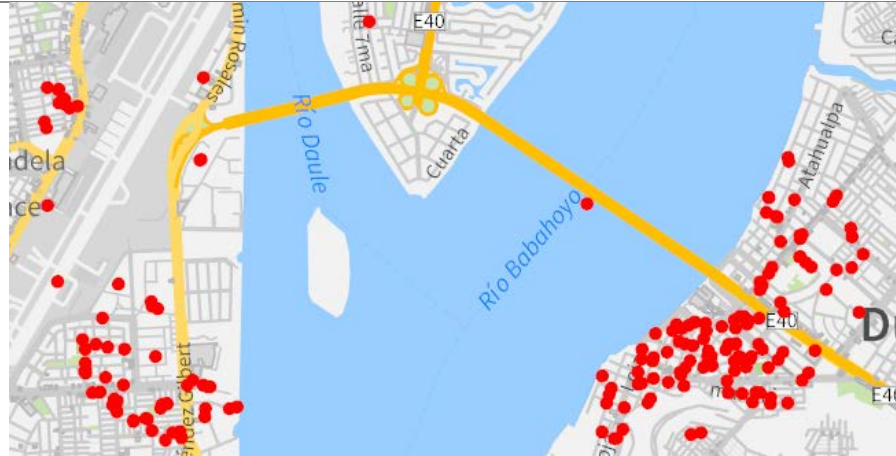
**Ruta 5**



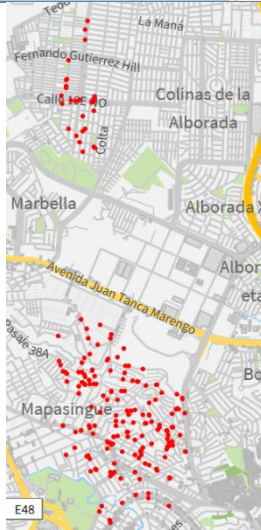
**Ruta 6**



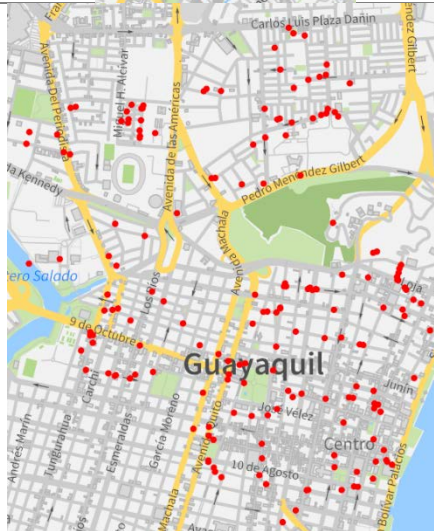
Ruta 7



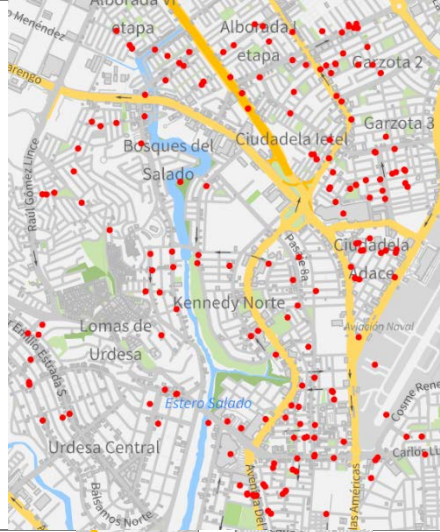
Ruta 8



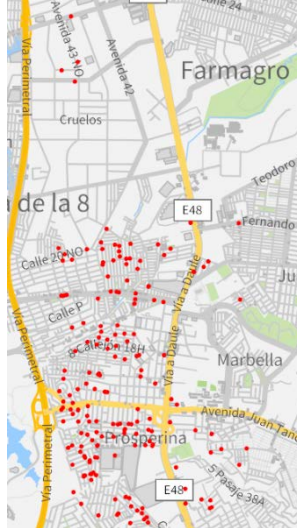
Ruta 9



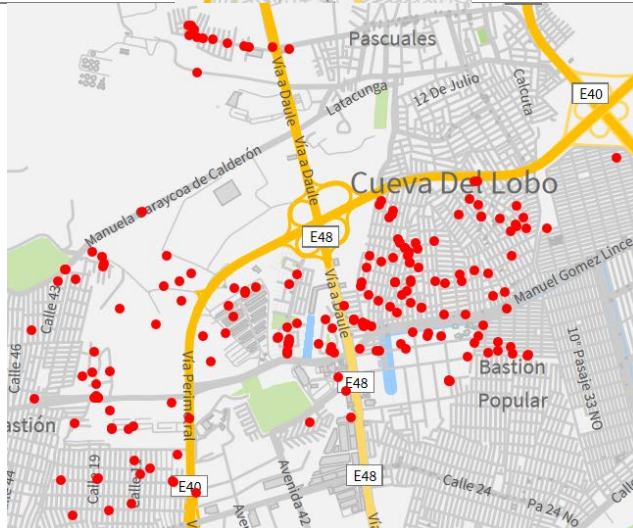
**Ruta 10**



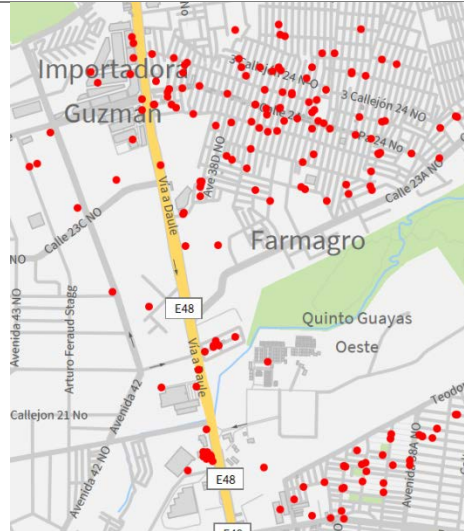
**Ruta 11**



**Ruta 12**



**Ruta 13**



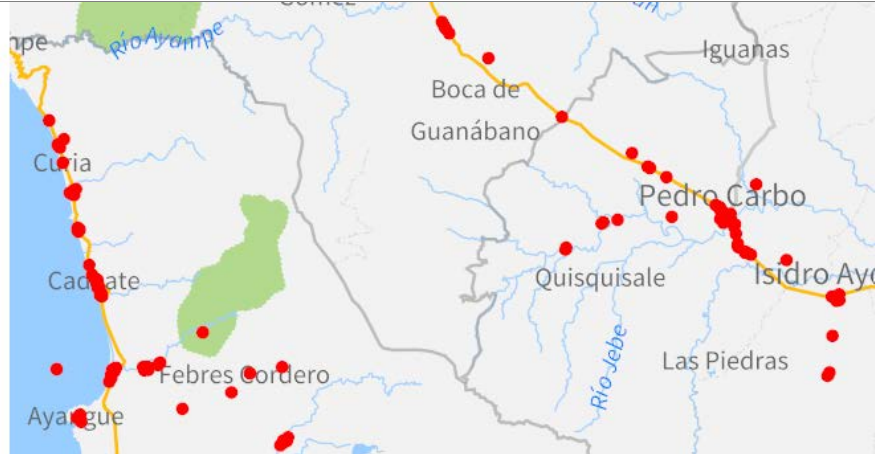
**Ruta 14**



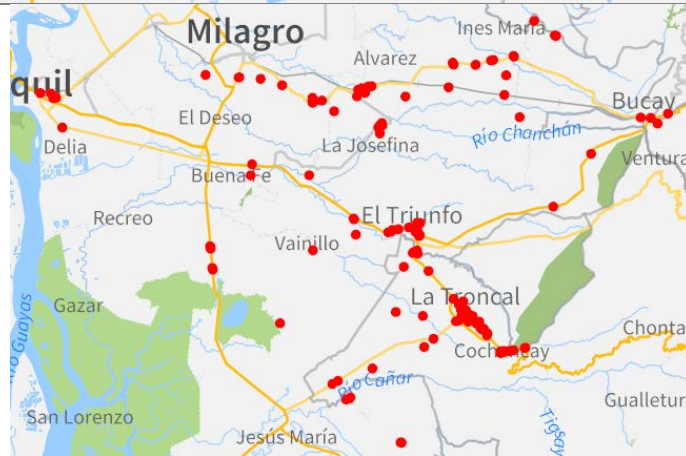
**Ruta 15**



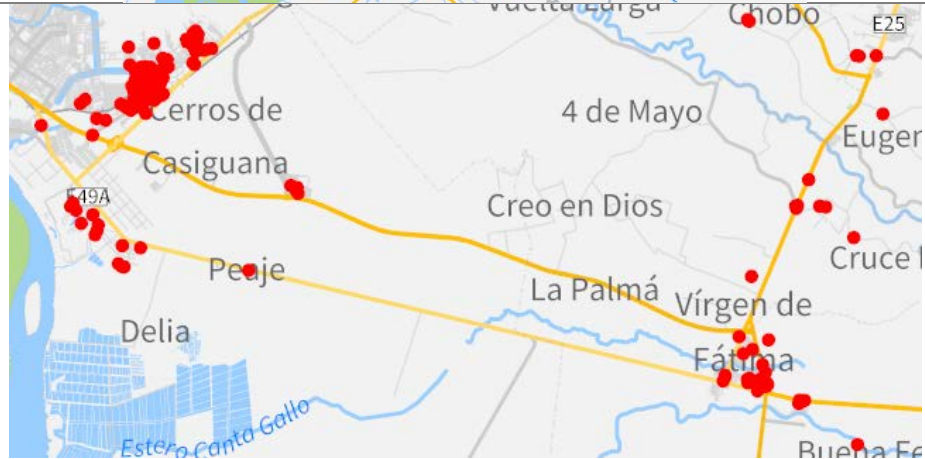
**Ruta 16**



**Ruta 17**



**Ruta 18**

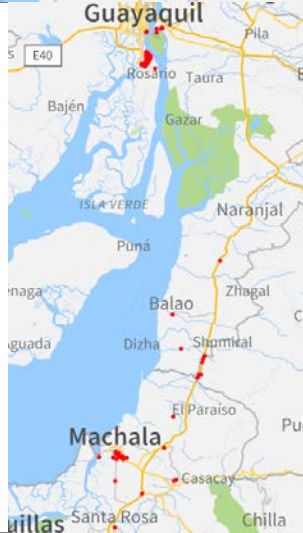




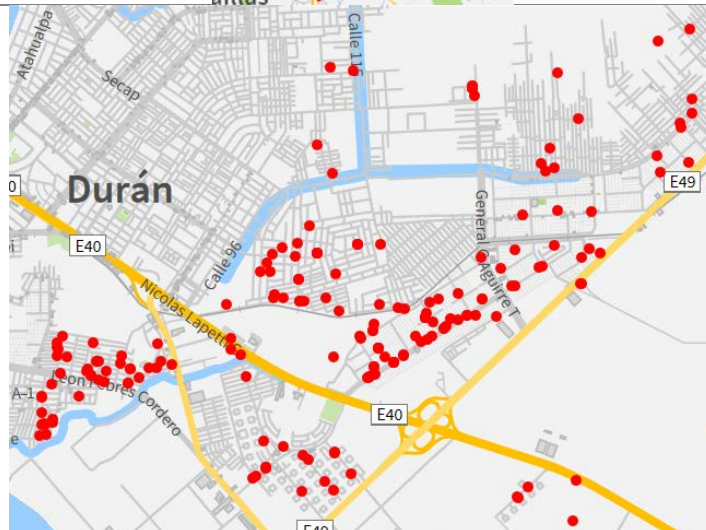
**Ruta 19**



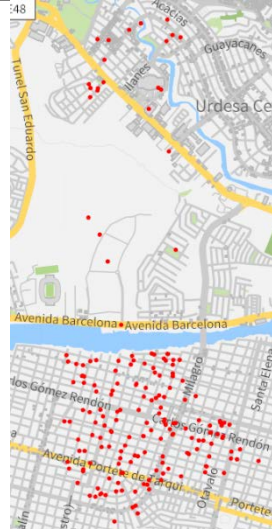
**Ruta 20**



**Ruta 21**



Ruta 22



Ruta 23

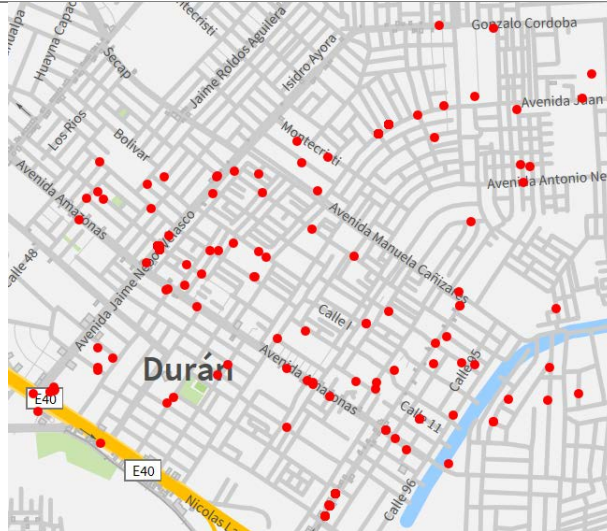


Ruta 24

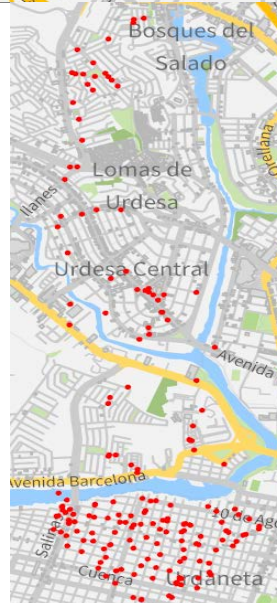




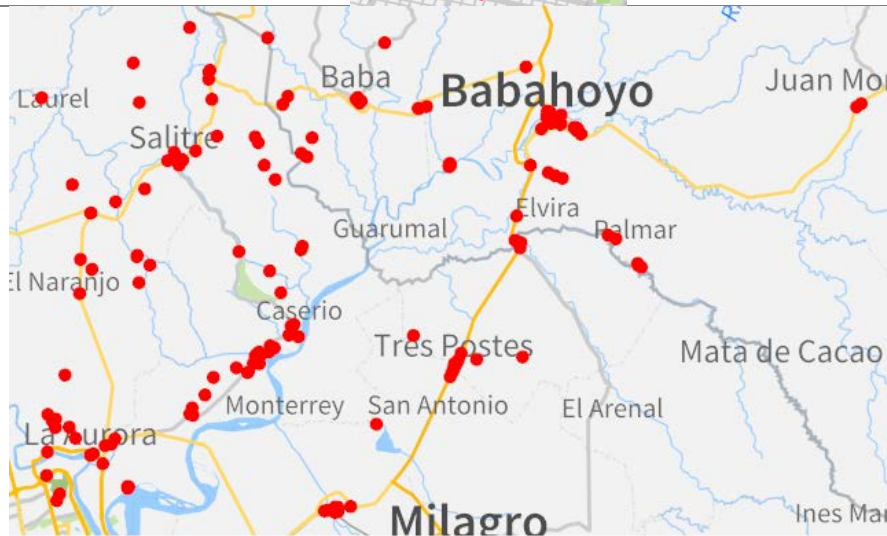
**Ruta 29**



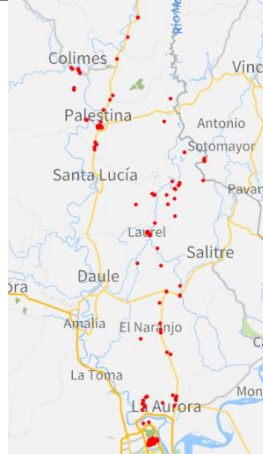
**Ruta 30**



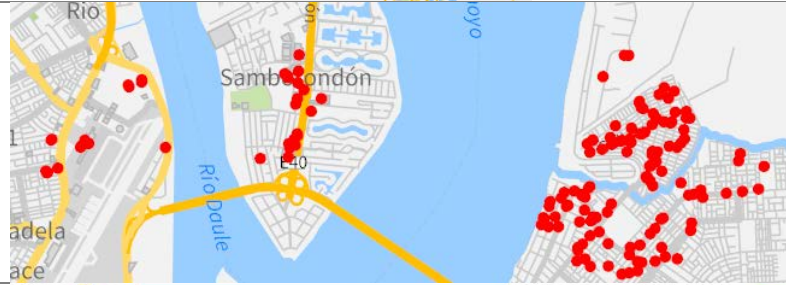
**Ruta 31**



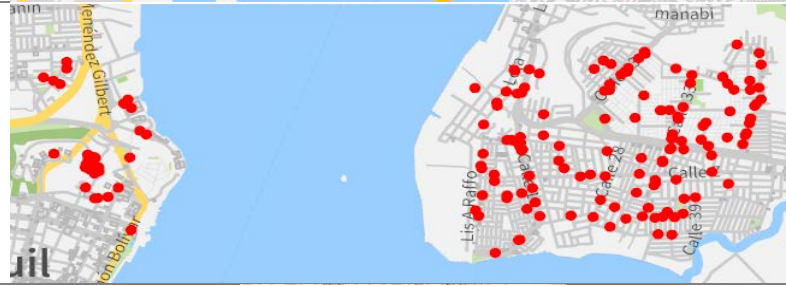
**Ruta 32**



**Ruta 33**



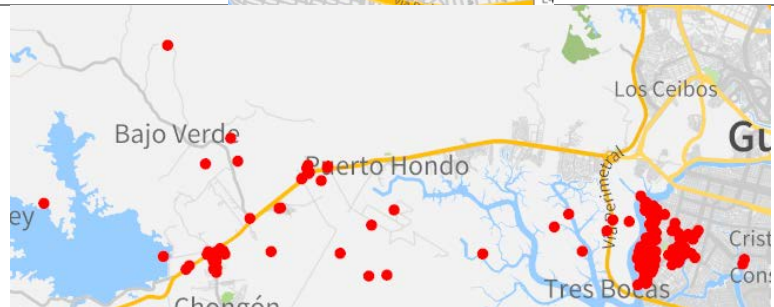
**Ruta 34**



**Ruta 35**



**Ruta 36**

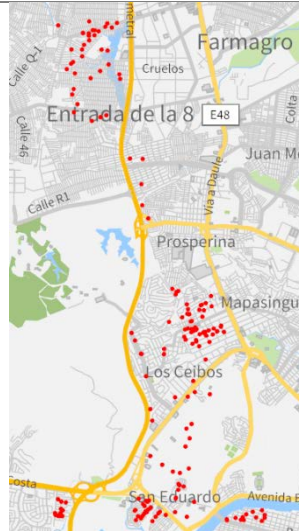






---

**Ruta 43**



---

**Ruta 44**



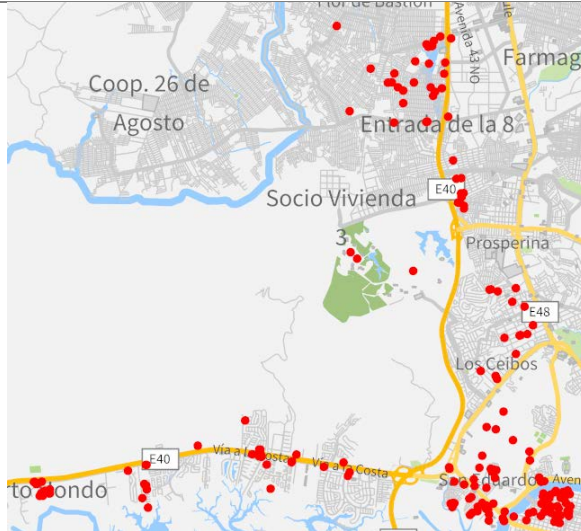
---

**Ruta 45**

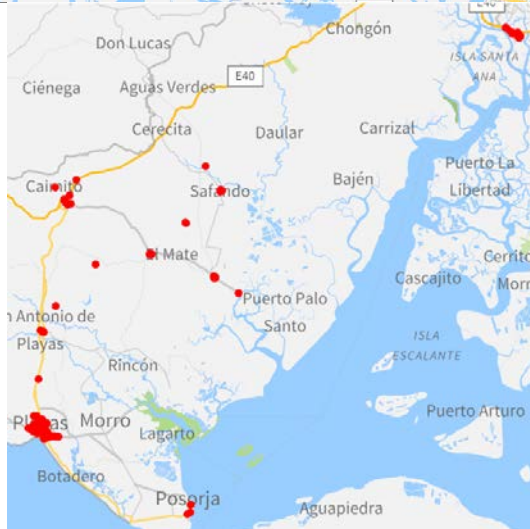




**Ruta 46**



**Ruta 47**

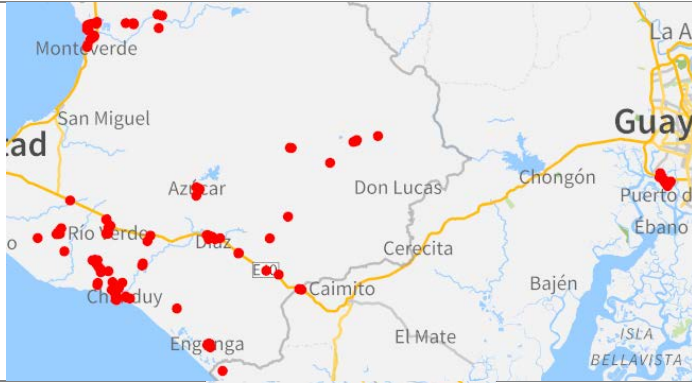


**Ruta 48**



---

**Ruta 49**



---

**Ruta 50**



## APÉNDICE C

### Secuencia de visita por ruta

Ruta	Día	Secuencia de visita
<b>1</b>	2	{0,4636,4709,5076,5077,5063,5035,5024,5164,5162,5221,5083,5121,5152,5160,7660,184,91,8102,269,98,229,4707,86,207,147,99,5037,5203,5141,5137,5235,7642,0}
	3	{0,301,267,289,264,230,188,199,129,170,111,165,130,228,215,88,293,216,166,167,96,146,263,3592,213,122,225,105,193,292,278,281,190,0}
	4	{0,4273,4166,4704,4079,4265,4191,117,200,273,5261,5183,5237,5021,4697,4078,4081,5132,5180,5036,5140,5033,5079,5027,5146,5190,5090,5220,5055,5156,5150,5241,5207,0}
	5	{0,5062,5109,4668,5122,5069,5143,5208,5082,5071,5138,5186,5238,5028,5262,4698,5215,5187,5209,5050,5255,5194,5230,5078,5214,5049,5248,5106,5080,5098,5104,5172,5179,0}
	6	{0,5218,5056,5195,5181,5234,5161,5070,5247,5115,5244,5107,5239,4638,5131,5068,5134,5025,5243,5067,5096,5184,5173,5127,5130,5087,5073,5201,5103,5213,5159,5126,5217,0}
	7	{0,5092,5105,5086,5147,5211,5038,5040,5041,5165,5042,5043,5044,5193,5099,5210,5048,5039,5246,5228,5174,5178,5148,5097,5129,5051,5136,5053,5075,5155,5058,5084,5047,0}
	<b>2</b>	2
3		{0,8701,1621,1625,1717,1646,1607,1567,1624,1711,1559,1693,1670,1648,1644,1550,1529,1527,1696,1525,1675,1626,1611,4703,1516,1647,1706,1597,1632,1713,1666,1702,4639,0}
4		{0,3294,3302,3284,4681,3241,3314,4665,3260,4653,3324,4663,3264,4658,4683,4664,1628,1643,1709,1700,1596,4694,1549,1610,1689,3279,3281,4071,3317,4689,3326,3280,4654,0}
5		{0,4687,4644,1577,1668,1605,1707,1836,1782,1725,1579,1541,1542,1568,1573,1574,1571,1682,1580,1653,1616,1705,1598,1582,1613,1590,1543,1697,4655,1649,1620,4685,1914,0}
6		{0,3239,4631,4650,1554,1581,1683,1553,1691,4641,4680,4660,1681,1716,1552,1690,1555,1606,1669,1676,1660,1576,1603,1684,1584,1619,1712,1642,1558,4657,4693,4695,9538,0}
7		{0,4647,3322,3333,3240,4642,3339,3262,3300,3328,3297,3275,3298,3296,3310,3299,3312,3327,3311,3288,3304,3236,3340,3277,3295,4691,3301,3265,3253,3254,3303,3291,3243,0}
<b>3</b>		2
	3	{0,201,197,60,61,77,177,185,187,118,159,196,2413,158,198,30,3120,3110,121,203,34,181,128,127,204,151,120,156,157,192,32,125,124,0}
	4	{0,69,76,81,205,209,110,101,211,222,40,106,112,100,119,148,123,63,109,6858,6743,6863,6771,6840,6714,6740,210,3458,175,97,169,3494,3455,0}
		}

	5	{0,68,52,50,82,282,107,3607,3543,3435,3485,3567,3568,3430,3431,3436,3444,279,3479,3472,3556,3460,3483,3600,3482,227,219,71,178,57,78,150,260,0}
	6	{0,5249,5188,5066,5125,5212,5032,5046,5119,5197,5254,5120,4702,74,51,72,135,142,70,48,149,176,102,271,94,290,265,9476,268,83,80,160,3410,0}
	7	{0,3442,3570,3451,3445,3446,3462,3538,4282,3450,3470,136,221,139,183,180,87,33,85,126,155,212,206,133,143,29,179,172,67,90,174,220,59,0}
<b>4</b>	2	{0,4120,4161,4182,4091,4118,4248,4267,4140,4077,4260,4163,4238,4177,4072,4198,4097,4214,4167,3049,4244,4216,3081,4181,4105,4179,4175,4106,4102,4243,4228,4089,4117,0}
	3	{0,4252,4651,4193,4087,4115,4150,4114,4103,4197,4205,4144,4145,4202,4124,4275,4235,4074,4272,4245,4184,4100,4126,4159,4075,4236,4268,4154,4640,4142,4128,4137,4213,0}
	4	{0,4119,4090,4234,4222,4220,4146,4092,4229,4084,4226,4192,4221,4067,4131,4209,5059,5142,5133,5111,3114,5263,5257,5108,5091,5057,5102,5258,5081,4669,4977,4271,4136,0}
	5	{0,3073,4116,4143,4233,4256,4086,4068,4101,4164,4178,4076,4093,4230,4172,4180,4066,4147,4190,4099,4083,4224,4110,4186,4204,4082,4231,4207,4241,4174,4111,4096,4274,0}
	6	{0,6428,6322,6358,4237,4978,4171,4253,4188,4069,4632,4215,4125,4255,4217,4270,4085,4944,4121,4206,4196,4135,4259,4262,4682,4149,4138,4208,4141,4148,4218,4210,4249,0}
	7	{0,4109,4195,4185,4113,4080,4122,4134,4139,4130,4945,4133,4162,4129,4676,4132,4183,4073,4187,4107,4108,4112,5060,5135,5110,5144,4246,4095,4168,4232,4152,4094,4151,0}
<b>5</b>	2	{0,3099,2990,2917,2947,3031,4247,4169,3088,4156,4251,4242,4200,4088,4155,4219,4240,4223,4170,4189,4158,4211,4098,2956,3102,3014,2991,2992,2918,2935,2994,2962,3076,0}
	3	{0,6878,6723,6737,6759,6735,115,152,144,145,224,173,116,113,41,226,191,3104,3434,3443,3432,3572,3585,3584,3437,3557,4047,4046,3986,3879,3979,6811,6885,0}
	4	{0,4157,4173,4225,4227,4199,16,3533,6261,4123,4127,2996,3057,3056,2943,3048,2997,3058,2944,4261,298,266,275,6856,6852,6864,6886,6757,6729,6755,7151,3115,7647,0}
	5	{0,3071,3093,2920,2937,2940,6342,2913,5166,3003,5168,5227,6264,6260,6313,6257,6284,6270,6324,7654,3007,3006,6251,3053,3010,3004,6275,3002,3497,4967,2914,3005,6282,0}
	6	{0,2916,6363,6412,3091,6385,6273,6292,6296,6293,6266,6393,6299,2938,3000,6247,6256,6309,6291,6305,6277,6337,6297,6304,6268,6341,6403,6373,2999,6310,6283,6250,3001,0}
	7	{0,7167,6754,6785,6859,6809,6876,3915,4009,4045,3905,3461,3534,3433,3586,3467,3448,3453,3552,3929,3953,4011,3930,3925,3923,3877,3995,3944,3888,4007,3948,3924,3946,0}
<b>6</b>	2	{0,1804,1803,1788,1807,1944,1899,1872,1793,1808,1897,1889,1797,1795,1805,1801,1806,1826,1910,1902,1882,1904,1768,1781,1770,1802,1901,1903,1789,1869,1794,1774,4873,0}
	3	{0,1767,1855,1750,1854,1749,1861,1748,1858,1912,1896,1900,1866,3196,3203,4637,3285,3248,4706,4633,3316,1931,4667,1785,1860,1771,1942,1737,1879,1741,1742,1743,4662,0}

	4	{0,3272,3237,4666,3249,3197,3198,3195,3150,3199,3287,3200,3184,3250,3293,3342,3274,3307,3247,3185,3201,3138,1823,1759,1873,1894,1936,4672,3202,3269,3306,1863,1934,0}
	5	{0,1724,5088,1747,1857,1856,3132,1764,1907,3134,3292,3220,3271,3273,1884,5232,5229,5206,5225,4212,5029,5191,1895,1817,1816,1810,1811,1818,1744,1746,1933,1930,1745,0}
	6	{0,4868,4886,4856,1773,1820,1878,4705,1822,1008,1935,1906,4646,1751,1913,1819,1815,1828,1867,1827,4677,5145,5199,5114,5177,5189,5250,5204,5026,5128,5085,4165,5100,0}
	7	{0,3282,3231,4673,3309,3190,4684,3283,3278,3180,3290,3186,3143,3217,3257,3286,3233,4659,3334,3187,3338,3223,3189,3224,3305,3261,3141,3193,3226,3325,3308,3194,3228,0}
<b>7</b>	2	{0,2314,2396,2266,2274,2378,2255,2248,255,2282,2380,250,2295,2315,2384,2377,2310,2409,2349,5416,5420,6505,6702,6497,6496,6514,6511,6506,6503,6491,8088,3,1701,0}
	3	{0,2346,2318,2214,2268,2269,2213,2297,2299,2304,2240,2399,2342,2262,2357,2387,3397,2241,2390,2337,2336,2317,2386,2225,8090,6698,6490,6697,6507,6501,5493,6510,6513,0}
	4	{0,2311,2343,2226,2334,2298,3350,3362,6711,6479,6483,6471,6469,6515,6476,6465,6472,5505,5435,2303,2362,2258,2359,2287,2382,2335,2286,2367,2407,9090,2234,2232,2221,0}
	5	{0,6434,6493,6454,6516,6518,6435,6463,6495,6460,6436,6508,6450,6509,6459,5503,6456,6502,6455,8066,6458,6504,6461,6703,6691,5491,6499,6447,6449,6512,6700,6445,5492,0}
	6	{0,5428,5422,5426,5417,5437,5500,5502,5577,8111,5440,5477,5479,5475,5473,5488,5484,5449,5454,5456,5472,5478,5556,5498,5487,5486,8087,5499,7671,6486,5412,5569,5410,0}
	7	{0,5494,6488,6710,6494,6485,6438,5496,6701,6705,6487,6489,6443,6498,6500,6492,6699,6532,6534,6451,6452,6444,6446,6441,6439,6692,5489,2679,6437,6704,6440,6517,6442,0}
<b>8</b>	2	{0,2102,1985,2078,2172,2080,1976,2006,2054,2014,2060,1946,2019,2109,2101,5874,1945,6029,7150,5841,5923,5842,5910,6037,5921,5839,5840,5920,2171,2158,2091,2022,2181,0}
	3	{0,2097,2131,2094,1975,1974,2043,1981,2035,2105,1983,2104,2036,2108,2184,2063,2090,2068,2106,2083,1989,2040,2034,2065,2076,2089,2030,2011,2010,2148,2176,1962,1961,0}
	4	{0,2151,2110,1949,2146,2155,2167,2021,5846,1948,2020,3398,2061,5931,5903,5844,5845,5981,5964,5911,1999,1947,6030,5896,5895,5958,5936,5909,6013,5926,5925,5843,5987,0}
	5	{0,2064,2012,2134,2013,1998,1953,2161,2067,2066,2162,1954,2130,2073,2001,1997,2099,2160,2057,2058,1950,2069,2166,2139,2059,1992,2129,2077,2079,2153,1977,2002,2081,0}
	6	{0,2050,2049,1955,2098,1956,2128,1951,1996,2016,2088,5930,2046,2008,3371,2023,2075,2192,2062,2194,2053,2152,1952,2055,2056,2116,2103,1988,2087,2142,2031,2136,2163,0}
	7	{0,4764,4917,4915,4867,4802,4796,4836,4784,4783,4808,4863,4782,4910,4831,4781,4788,4814,4770,4809,4829,4835,4907,4908,4771,4786,4830,4780,4857,4789,2107,2093,2052,0}

<b>9</b>	2	{0,2391,2365,2395,2307,7540,2323,2320,2281,2406,240,243,2329,2403,2321,2328,248,239,7943,7855,257,258,7914,7949,7930,2326,2325,235,5962,2348,2373,2370,2374,0}
	3	{0,5863,5862,5943,5852,5905,237,5899,6006,4888,7875,7853,7970,3392,3354,3381,3353,5900,5938,5940,2238,2247,2381,2283,2246,2408,2205,2196,2270,2291,2231,2198,2201,0}
	4	{0,2195,2217,2249,6020,2366,6021,5956,6026,6025,8056,9521,9469,9479,3361,5939,5861,7974,9485,9490,9522,245,6023,5860,6024,6022,2372,5946,5961,5960,6035,5957,2364,0}
	5	{0,8053,8052,247,3117,9500,9484,7929,7928,7894,7897,7895,7893,7892,7891,7887,7905,7890,7957,7975,7942,7977,9498,7956,9463,9486,9516,7913,246,9517,9462,7896,3111,0}
	6	{0,2227,2245,3373,4608,2219,2308,2215,2369,2216,3374,2411,2218,3402,2296,2393,2312,2404,2316,2257,3403,3349,2309,2236,2376,2273,2224,2223,2222,2344,2235,2243,2239,0}
	7	{0,7852,7969,7968,7851,7837,7906,7907,7908,7909,7910,7911,7904,7912,7858,7850,7862,7915,3109,3112,3125,3131,9510,7922,7889,7173,6063,9524,7877,5929,5944,7900,7925,0}
	<b>10</b>	2
3		{0,2202,3420,2300,2301,3382,2290,2203,3383,2353,2237,2361,2228,2230,2261,253,2256,3348,3386,3345,2379,254,3413,2341,2289,4975,2345,3407,2313,3347,2292,849,2242,0}
4		{0,5175,1570,1673,4648,4629,1565,1639,1662,1563,1564,4649,4696,1640,1589,1518,1633,1587,1517,1595,1664,1672,1566,1629,1686,1698,1599,4699,4674,1617,1656,1562,1561,0}
5		{0,1548,1547,1546,1591,3367,5857,6008,5870,1618,1569,1674,1575,3412,1634,1663,1572,6016,3358,5891,5902,5873,5869,3390,5904,5906,3375,6040,3346,5858,5859,5932,5153,0}
6		{0,3416,3368,3365,3384,3391,3355,2355,2394,2204,2244,2294,2251,2360,2284,2200,2305,2267,2254,2280,2199,3352,3405,3400,3378,7182,2385,2405,2398,2197,2253,2371,3385,0}
7		{0,5877,5892,5893,5894,5915,6039,5890,5985,3387,5954,5965,3359,3394,6034,5966,5889,5887,5888,5886,6011,5885,3421,5941,3401,3417,5865,5847,3409,5897,5000,5871,5934,0}
<b>11</b>		2
	3	{0,1973,2824,2872,2743,2727,2871,2896,2777,2736,2745,2810,2734,2864,2753,7024,7027,2119,2039,2086,7155,2117,2038,2127,5838,2032,1969,2905,2904,2903,2902,1972,2044,0}
	4	{0,2190,1963,2017,5880,2018,5837,5882,2767,2729,2829,2797,2801,2869,2866,2800,2884,2870,5990,5974,5975,2888,5968,5989,5969,5976,5967,5971,6002,5881,2113,2124,1994,0}
	5	{0,2746,2742,2744,2724,2841,2725,2719,2781,2716,2715,2897,2737,2733,2772,2752,2856,2779,2782,2720,2787,2738,2795,2846,2763,2845,2728,2833,2831,2855,2732,2796,2759,0}

	6	{0,2808,2877,2853,2802,2805,2806,2851,2809,2857,2895,2751,2886,2838,2755,2811,2839,2852,2803,2865,2814,2899,2721,2740,2765,2793,2747,2723,2818,2771,2819,2880,2799,0}
	7	{0,3344,4894,2191,2185,2122,2123,2121,2120,2820,2183,2180,5928,2092,2125,2156,2000,5901,2832,2881,3129,5999,5916,5883,6001,6000,5996,2867,2887,3369,7134,3025,3019,0}
<b>12</b>	2	{0,7128,3098,2924,6720,3036,2922,2955,2942,2974,3015,2952,3065,2921,6254,6265,6345,6348,6298,6384,6347,6370,3094,6332,6248,6420,6405,3077,6314,2949,3016,6732,6731,0}
	3	{0,6311,6263,7156,6733,6724,6716,6728,6842,6877,6843,6756,6739,6788,6738,6789,3008,6330,6267,6375,6380,6319,6376,6352,6394,6382,6252,6427,6395,6371,6424,7143,6369,0}
	4	{0,6349,7171,6387,6321,6359,6294,5167,6327,6318,6374,6404,4652,6336,6317,7179,4153,4160,4679,4257,4263,4258,4264,6351,6249,6401,4269,6398,7152,6408,6386,6389,6418,0}
	5	{0,6397,6286,6323,6272,6295,6344,6346,6388,6281,6331,6279,6400,6362,6308,6377,6271,6285,6316,6415,6399,6417,6339,6416,6379,6290,6320,4250,6300,4254,6335,6390,6329,0}
	6	{0,7162,6364,6367,6301,6302,7656,7655,6258,7644,6255,7667,6274,6414,6407,7657,6409,7649,6350,6276,6340,4936,6421,6365,6378,5259,6253,6307,7133,6422,6392,6312,5170,0}
	7	{0,6303,6328,6259,6289,6326,6391,6402,6368,6306,6353,6288,6334,6325,6361,6354,6315,6269,6372,6381,7653,5171,5094,5169,6280,6366,6343,6262,6333,6357,6396,6406,6278,0}
<b>13</b>	2	{0,3575,7032,3121,288,277,7035,6974,8077,8064,6981,1905,1850,1821,1814,1813,1812,1766,1784,1892,1758,1824,2478,8014,7650,6287,1729,3579,4848,4792,1593,4765,4840,0}
	3	{0,4753,4878,4889,4766,4794,4876,4758,4891,4900,4756,4757,4859,4914,1730,4879,4841,4870,4800,4862,4744,4768,4865,4847,4877,4761,4861,4760,4716,4730,4918,4755,4901,0}
	4	{0,2932,2946,3021,3103,3097,3043,2995,2959,2989,3026,3101,2986,3035,3022,2984,3045,4591,7127,7137,3072,2969,2958,2939,3038,3060,2951,2941,2964,5019,3051,2967,3069,0}
	5	{0,4266,2998,3075,2963,2925,2948,3100,4678,7169,7170,7130,7648,2957,3046,2933,2927,2919,2934,4610,2926,2953,2930,3013,2931,3083,2928,2929,7146,2915,3061,4203,3067,0}
	6	{0,3066,2975,3023,3018,3044,3039,3011,3028,2972,2965,3078,2978,2976,3037,2993,4675,3096,3124,3017,3041,2960,2961,2936,3074,3105,2971,3020,3033,3029,2970,3082,3090,0}
	7	{0,2950,3024,3009,3085,3095,2968,3079,3040,3080,2987,2985,2980,2977,2954,2988,4104,3084,4239,4194,3012,2981,2973,3064,3063,2966,3032,2982,2983,2945,2979,3042,3054,0}
<b>14</b>	2	{0,3896,3960,353,374,375,459,325,339,413,377,387,2422,410,323,354,379,2425,2420,7645,7669,2418,437,436,3971,388,409,358,2421,383,393,369,3106,0}
	3	{0,3966,3914,3991,3974,418,337,376,378,3900,3885,3883,3938,3919,3906,3880,4022,3909,4286,3932,3881,3889,4064,463,408,434,427,338,345,407,344,3962,3947,0}

	4	{0,3969,4021,3895,4016,4063,3963,3911,3985,4008,3937,3990,3927,3940,4061,3878,3890,4060,4065,3916,4033,3892,4025,3922,4062,3978,3973,3996,3928,3999,3951,4058,3901,0}
	5	{0,3087,6423,6338,6356,3050,6419,3052,6360,5017,6426,6411,6889,6868,6745,6767,6829,6821,6717,6822,6018,6773,6787,6853,6837,6819,6866,6768,6835,6870,6836,6801,6734,0}
	6	{0,3086,6355,6383,6410,6413,6830,6727,6867,6888,6881,6882,6736,6790,6820,6855,6761,6834,6770,6722,6873,6726,6741,6793,6719,6791,6721,6792,6712,6872,6742,6725,6713,0}
	7	{0,6718,7178,6844,6851,6786,6715,7142,6810,6753,6752,3127,6772,6807,3952,3893,3933,3956,3898,3500,3499,4013,4031,3934,3972,3882,4054,3988,4010,3997,4048,4017,4043,0}
<b>15</b>	2	{0,347,305,386,334,363,443,451,304,462,444,452,7661,348,442,341,411,322,445,441,315,328,329,453,421,330,364,4326,3517,4297,4309,2464,306,0}
	3	{0,320,362,319,316,317,361,360,318,382,392,366,4302,3459,4331,4337,4335,4390,3522,3464,4336,373,352,359,381,414,3536,3520,4281,3527,4285,4363,4338,0}
	4	{0,7174,4005,2443,2481,4032,3917,4003,3897,4056,3945,3894,3955,4053,3957,2466,2471,3501,3539,7652,3558,3535,450,457,461,2468,456,346,332,343,356,321,415,0}
	5	{0,3975,3913,3935,3989,4052,4001,4059,4038,4026,3949,3921,4000,3480,3884,3891,3983,3908,3902,4004,3981,4039,3931,3876,4367,4366,3608,4403,4405,4398,4415,4319,3961,0}
	6	{0,3886,3939,4014,3976,4015,3926,3987,3887,4040,4035,3992,3964,3950,4020,3936,3941,3910,3977,4002,3994,4019,3993,4037,3942,3912,4057,3959,3904,3907,4012,3965,2452,0}
	7	{0,4287,4414,4284,4322,4364,4334,4372,4333,4312,4332,3515,4320,4292,4289,4376,4394,4328,4327,4296,3954,3903,2447,4006,3984,3982,4374,4330,4371,4329,3958,3967,3943,0}
<b>16</b>	2	{0,446,447,448,303,313,430,324,370,431,312,335,2417,2453,350,404,403,432,331,310,401,336,307,416,351,439,327,458,420,371,326,309,455,0}
	3	{0,426,402,7664,2423,7658,2437,7646,7662,389,422,460,440,372,365,390,308,385,2449,380,395,412,423,425,424,391,449,394,428,357,384,435,417,0}
	4	{0,368,438,419,405,406,342,311,333,2446,396,355,397,400,314,433,398,399,367,8031,8030,8020,8029,8040,8032,8027,8028,7939,7938,349,429,340,454,0}
	5	{0,6175,6097,6236,6210,6176,6226,6227,6096,6094,6095,6161,6177,6188,6104,6185,6183,6143,6139,6138,6203,6132,6131,6135,6133,6134,6204,6136,6137,6099,6191,6192,6189,0}
	6	{0,6145,6243,6219,6144,7940,8002,8001,7998,8000,7995,8035,7999,8039,7993,7994,6100,6193,6146,6194,6235,6208,6180,6195,6179,6209,6246,6101,6244,6102,6234,6103,6238,0}
	7	{0,7997,7996,8003,8023,8024,6178,6171,6164,6169,6165,6170,6214,6166,6213,6212,6228,6215,6239,6211,6216,6167,6168,6240,8051,8034,8025,7979,8033,8042,8041,7978,6163,0}
<b>17</b>	2	{0,588,544,566,613,600,637,552,547,556,565,512,621,535,513,8096,546,592,596,497,625,595,531,526,519,518,614,555,528,620,601,8109,470,0}



	3	{0,987,8104,982,974,979,983,1013,583,563,561,560,590,593,610,587,632,624,591,623,8116,557,558,502,577,493,648,539,574,575,576,573,609,0}
	4	{0,510,585,586,615,475,474,616,486,617,618,619,553,536,635,633,527,578,597,598,480,492,564,629,599,647,491,567,542,540,537,559,554,0}
	5	{0,465,631,611,622,589,562,612,523,514,472,503,548,549,551,499,525,550,628,464,584,636,638,504,641,640,642,505,506,478,496,498,532,0}
	6	{0,2617,2628,5334,5345,5359,5363,5382,2552,2551,2527,2610,959,1069,1070,1023,940,1049,1078,5011,8106,939,967,1016,993,1025,937,941,972,955,947,975,945,0}
	7	{0,1073,1082,7147,986,8100,997,1047,1046,1022,1045,1044,1043,1010,1087,1075,989,1029,1011,1050,8110,9044,9046,9043,8103,8101,1077,1076,1006,1028,1027,1012,1001,0}
<b>18</b>	2	{0,1204,1207,1187,1196,1186,1185,1161,1167,1176,1170,1245,1174,1298,1299,1322,1270,2709,2710,2585,958,1048,928,1030,1014,998,929,936,1057,992,991,944,1085,0}
	3	{0,2600,4428,2605,985,1040,1019,1042,4535,1041,1062,4509,969,8074,981,1061,994,1038,1072,995,952,1063,935,943,984,965,933,4420,964,988,930,931,4567,0}
	4	{0,1213,1216,1218,1217,1232,1311,1180,1206,1188,1208,1199,1193,1177,1200,1190,1198,1195,1202,1203,1205,1312,1226,1239,1233,1236,1320,1269,1230,1235,1210,1234,1219,0}
	5	{0,5306,1511,1212,1223,1222,1214,1224,1221,1238,1215,1220,1169,1166,1162,1164,1172,1178,1163,1165,1296,1294,1295,1301,1280,1278,1276,1277,1315,1168,1303,1211,1209,0}
	6	{0,2701,2556,2547,1257,2493,2500,2499,2489,2490,8063,2491,2492,2495,2496,2497,1266,2498,1251,1242,2494,1191,1273,1189,1181,1184,1182,1183,1179,1194,1192,2582,2584,0}
	7	{0,1197,1260,1252,1255,1313,1256,1259,1265,1264,1263,1247,1241,1243,1244,1248,1267,1201,1237,4456,4453,4435,4440,4570,4439,2609,2604,2607,2706,2589,2599,4426,2586,0}
<b>19</b>	2	{0,1120,1119,1121,1118,1117,1115,1116,1127,1125,1150,1149,1129,1135,1104,1111,1107,1109,1106,1139,1114,1136,1113,1112,1103,1159,1143,1137,1175,4443,1148,1144,1157,0}
	3	{0,1105,1097,1092,1095,1100,1089,1090,1102,1094,1091,1096,468,481,467,471,469,466,538,951,1021,970,1080,932,1068,934,1086,1051,1002,953,4450,4445,1099,0}
	4	{0,1093,4998,5001,484,533,515,604,534,495,482,605,634,607,571,606,516,608,487,524,477,511,485,494,4992,522,517,488,630,529,639,646,7665,0}
	5	{0,1318,4452,4441,4454,4449,4562,4571,4531,4438,4432,4560,4434,4559,4437,4436,4433,4442,4444,4448,4446,4451,4447,4455,4424,4430,4431,572,626,6641,6683,6689,8105,0}
	6	{0,1133,1275,1274,1271,1146,1297,1131,1132,1134,1140,1122,1123,1126,1152,1138,1147,1151,1153,1128,1130,1309,1124,1154,1142,1173,1158,1141,1156,1145,1160,1155,1171,0}
	7	{0,1017,1000,999,948,942,1081,1026,508,644,568,602,473,569,645,479,643,545,489,954,582,5018,570,500,521,501,520,627,483,603,541,4701,476,0}

<b>20</b>	2	{0,9117,9257,9064,8894,8838,8839,8899,8864,8863,8843,8850,8854,8824,8946,8841,4972,9023,8866,7651,579,580,543,581,4989,509,507,9060,8976,8875,8876,8869,8954,0}
	3	{0,9118,9178,8914,8905,8892,8909,9501,9027,8847,8961,8960,9078,8842,8878,8962,8877,8949,8958,8978,9495,9533,8994,490,530,4997,594,5008,5020,9536,8885,8998,9032,0}
	4	{0,9017,9290,9291,9180,9264,9237,8942,4690,9079,8945,8965,8870,9022,8943,8853,8910,8859,8904,8825,8871,9025,8935,8865,9061,9531,9491,9013,9014,9175,8860,8861,9235,0}
	5	{0,9058,9005,9159,8927,5016,5004,4990,4991,5013,5023,5022,5007,4985,5002,5012,4942,5015,4940,5003,5006,4943,4948,4980,4986,4987,4983,4984,4979,4988,4981,4993,4982,0}
	6	{0,6595,6653,6655,6680,6656,6658,6668,6666,6664,6663,6645,6661,6651,6650,6648,7934,9188,9187,9223,9189,9255,9130,8913,9190,9191,3778,9131,9116,9281,9282,9202,9183,0}
	7	{0,7976,7832,9279,9295,9146,9199,9194,9139,9151,9115,9266,9144,9124,9138,9105,7666,9238,9283,9267,9170,9296,9179,9231,9285,9108,9136,9214,9496,3871,9133,9240,9186,0}
	<b>21</b>	2
3		{0,2560,2569,2583,5346,5267,2516,2501,8075,2705,8089,2502,2505,2504,2503,2506,2507,2591,2592,4427,8108,8115,1249,8112,1246,1262,1254,1261,1258,1253,1250,1240,1316,0}
4		{0,6687,6686,2693,2687,6684,8097,8061,8092,2531,2535,2533,2537,2541,2542,2544,2545,2712,2581,2571,2572,2573,2577,2488,2688,2561,2566,2568,2564,6429,2574,2576,5272,0}
5		{0,2526,2546,2548,2549,2550,2532,2557,2555,2553,2543,2539,2511,2510,2508,2509,2520,8114,2523,2528,2529,2524,2522,2521,2518,2515,2514,2513,2517,2512,2538,2519,2525,0}
6		{0,6673,6678,6671,6706,6670,2708,2598,2595,2593,2590,2601,2596,2707,2588,2608,2594,2606,2603,2602,2587,2536,2559,2534,4425,8084,8069,2540,2554,2530,4429,2597,2558,0}
7		{0,2703,2692,2685,2702,2699,6521,2700,6707,6681,6679,6667,6677,6643,6646,2704,2686,2681,6685,2690,2698,2680,2697,2691,2711,2694,2689,2684,2695,2682,2696,2683,6694,0}
<b>22</b>		2
	3	{0,8656,8736,8681,8689,8725,8673,8687,1466,1487,1384,1423,1453,8708,8709,8751,8643,8678,8683,8692,8778,8810,8812,8811,8658,8688,1426,8646,8795,8750,8695,8657,8788,0}
	4	{0,5933,5898,6014,3363,5988,6028,5849,5850,5848,5851,3366,3380,3388,6009,5912,3428,5875,5949,5918,3429,8809,8754,8796,8711,8672,8710,8706,8729,8050,8731,7359,3817,0}
	5	{0,5748,5616,5765,5761,5806,5581,5614,5590,5582,5749,5615,8761,5833,699,8699,8779,5613,5682,5617,5618,8742,5687,8690,8653,8637,5688,5757,5620,5584,5756,5810,5700,0}

	6	{0,8730,8721,8759,8790,8679,8767,8684,8774,8713,5736,8797,8818,8640,8816,8784,8739,8631,8747,8770,8722,8768,8756,8799,8676,8697,8755,8696,8645,8642,8760,8800,8630,0}
	7	{0,8791,8801,8641,8675,8752,8753,8634,8632,8638,8666,8798,8628,8629,8717,8815,8611,8662,8746,8615,7367,7342,7240,4966,4941,8636,8749,8741,8792,8659,8738,8655,8765,0}
<b>23</b>	2	{0,893,837,661,735,827,869,751,871,1396,780,726,867,715,880,875,876,737,918,890,652,888,742,860,878,654,879,6955,847,877,693,696,819,0}
	3	{0,5010,809,788,864,685,831,883,710,794,800,781,761,787,894,882,826,3708,3751,3796,3667,3618,3836,3670,3624,3626,3860,3620,3627,889,7051,866,887,0}
	4	{0,9526,848,746,747,845,923,673,9528,720,914,900,822,921,762,8827,7046,807,6959,724,6957,906,920,922,911,904,902,7094,9148,706,708,905,844,0}
	5	{0,896,8972,3688,3682,9030,5224,3847,3625,3859,3685,3691,3749,3760,3830,3802,3804,3813,3671,3694,3712,3793,9534,3808,3733,3798,3800,3826,3807,3815,3858,3719,3848,0}
	6	{0,1447,1455,1444,1467,4938,1501,1490,6961,765,898,916,892,9508,717,792,926,718,874,924,4955,745,919,1326,1498,1479,1388,662,663,1428,782,1448,4946,0}
	7	{0,3730,3629,3721,3623,3709,3805,3674,3832,3706,9480,4935,9154,9457,9511,8073,3686,3649,3820,3630,3827,3828,3619,3661,3681,3814,8071,9210,9182,9227,9477,7096,7093,0}
<b>24</b>	2	{0,7226,8724,8703,8619,8612,8771,8776,834,805,8700,649,835,8621,8622,8737,8698,8624,1506,8614,8716,8733,8734,8651,8720,741,8613,8617,8732,8748,8694,8735,8616,0}
	3	{0,664,4952,852,801,812,802,855,903,711,686,714,775,784,797,658,660,8806,8623,674,672,4974,740,666,7299,8769,8783,8805,8789,8775,1387,1474,927,0}
	4	{0,668,854,808,825,727,828,3728,779,707,725,709,824,767,791,861,653,755,734,2848,912,832,698,842,776,655,651,915,684,815,850,700,783,0}
	5	{0,680,820,829,704,729,789,743,772,752,677,768,863,695,777,816,884,841,701,688,682,659,730,798,736,728,853,836,7390,857,773,881,702,0}
	6	{0,7231,7272,7250,7220,8620,8766,7221,8702,8618,8744,763,7358,774,7292,656,769,796,669,814,744,830,858,665,748,7352,3621,7238,7252,804,705,7401,7245,0}
	7	{0,7291,7191,7396,7189,7206,733,692,771,818,722,899,795,676,667,885,817,712,757,697,679,753,766,690,723,756,786,917,670,750,703,790,7357,0}
<b>25</b>	2	{0,9487,1368,1510,1378,1411,1494,1472,1334,1365,1449,1471,1505,1504,1382,1483,1366,1380,1360,1484,1385,1491,1514,1515,2450,6960,7088,9164,1482,1480,1486,1485,9506,0}
	3	{0,1460,1425,1391,1508,1433,1435,1386,1404,1329,1507,1503,7099,7047,1488,4969,891,1432,1436,1340,1452,1392,1374,1350,1381,1478,1406,1502,1492,716,1335,1457,1405,0}
	4	{0,1339,1346,1499,1331,1355,1324,1419,1407,1367,1363,1362,1509,1408,1469,4950,1325,1465,1476,7159,4976,1454,1458,1461,913,897,873,1418,1336,1414,1323,1446,1327,0}

	5	{0,7886,8018,4994,7917,7860,7898,1429,1357,1434,1333,1495,1424,2854,1354,1473,1330,1364,1377,1358,1373,1349,4939,1389,1416,1438,1437,1403,9518,7955,7921,7920,9466,0}
	6	{0,7935,7881,7841,7868,7972,7870,8013,7937,7846,1328,7867,1402,7844,1431,1445,1375,1443,7859,5005,7849,1372,1401,1398,1420,8822,1345,1342,1379,1417,1415,1359,1442,0}
	7	{0,7866,8019,4626,7933,7883,7838,7842,7973,1464,4937,7916,7899,7670,7931,7845,7878,7865,7863,7879,7946,7923,7924,7839,3123,3126,3122,7950,8006,7668,7944,8021,7926,0}
<b>26</b>	2	{0,1796,1777,1874,1888,1890,1871,1791,1920,1832,1790,1792,1779,1738,1852,3191,3144,3145,3230,3211,3136,3204,1654,1635,1714,1630,3179,3155,3154,3213,1650,294,3235,0}
	3	{0,1020,938,1039,1065,960,1037,8083,963,977,1055,950,1053,1054,990,1018,1031,1032,1052,1079,1056,1064,1088,957,1007,6708,946,8093,8091,8065,8098,1009,1071,0}
	4	{0,4472,4480,4422,4547,4475,7663,4554,4553,4485,4495,4529,4550,4584,4587,4588,4513,4561,8082,8081,8107,7643,973,1005,1074,1067,1066,1035,976,1059,1083,1036,1058,0}
	5	{0,3259,1540,1655,1704,1538,1536,1535,1537,8095,996,1033,1003,1004,949,1034,978,966,961,962,4463,4537,4423,4564,4555,4536,4519,4552,4572,4518,4494,272,286,0}
	6	{0,1881,1848,1778,1847,1844,1929,1851,1932,1916,1765,1845,1928,1859,1837,1839,3212,3210,3332,3183,3178,3142,1614,1734,1735,1939,1843,1752,1732,1846,1915,1853,1760,0}
	7	{0,4522,4512,4577,4500,4484,4421,4483,4474,4479,4492,4473,4534,4568,4580,4508,4565,4556,4581,4566,4575,4576,7145,971,968,1024,8079,956,1060,8076,1015,980,1084,0}
<b>27</b>	2	{0,3221,3289,3170,3165,3163,3166,3313,3216,3153,3331,3321,3152,3158,3252,3161,3251,3245,3157,3214,3336,3276,3205,3329,3335,3219,3151,3215,3173,1708,3156,3169,3168,0}
	3	{0,1877,1753,1783,1736,1922,1937,1924,4686,4962,1876,1842,1776,1838,1840,1733,1941,1769,1940,1943,4645,1731,1740,1739,4671,3229,262,3209,3174,3177,3323,3255,1757,0}
	4	{0,3135,1585,1522,1521,1592,1519,1677,1586,1641,1638,1523,1631,1694,1615,1533,1534,1658,1622,1659,1604,1623,1695,1671,1710,3225,3164,3140,3139,53,2174,89,217,0}
	5	{0,3330,1636,1608,1715,1678,1539,4700,5321,296,274,1314,1307,8094,1279,1272,1281,1284,1282,1288,1285,1317,1304,1300,1308,5518,5517,5575,5568,5390,5521,5557,1287,0}
	6	{0,1921,1865,1809,1560,1862,1727,1728,1841,1756,1925,1667,1665,1551,1679,1780,1556,1680,1692,1557,1911,1602,1688,3133,1524,4670,3207,4661,3176,3175,3172,3337,3181,0}
	7	{0,3268,3206,3244,3146,3318,3222,3227,3234,3238,3258,3171,3162,3270,3188,3147,4688,3218,3232,3208,3148,3160,3256,3159,3167,3315,3137,3341,3267,3319,3263,3266,3149,0}
<b>28</b>	2	{0,4804,4885,3414,1923,1775,1762,1723,1835,1834,3389,4892,4635,3411,1645,1637,1578,1685,4895,2173,2095,1289,3343,4893,1544,4656,1919,1918,1833,1721,1908,1720,1754,0}

	3	{0,1787,1786,1798,1898,1875,1870,1864,1880,1926,1885,1887,1726,1825,1830,1886,1868,1722,1917,1938,1800,1719,1893,1927,1799,1763,1718,4852,4898,4849,4890,4791,4930,0}
	4	{0,4827,3182,4866,4737,4733,4896,4747,4933,4721,4739,4822,4860,4920,4743,4726,4741,4742,4911,1909,1831,1829,1761,1755,4735,4875,1891,4825,4732,4903,4923,4929,4731,0}
	5	{0,4858,4869,4833,4846,4872,4823,4880,4887,4883,4749,4717,4851,4855,4854,8246,4720,4919,4795,4724,1883,4738,4740,4925,4864,4722,4884,4816,4850,4881,4871,4832,4748,0}
	6	{0,4826,4715,4905,4734,3370,2164,7857,5950,2115,3399,2145,2111,2072,2112,2029,2189,1995,2074,1971,2132,2168,2071,3404,2147,1990,2009,1970,2170,3426,3396,2154,3406,0}
	7	{0,4759,4729,7164,4785,4922,4821,4793,4750,4834,4819,1849,4798,4818,4799,4843,3192,4845,4844,4745,4932,4790,4797,4912,4810,4913,4815,4751,4763,4767,4754,4909,4928,0}
<b>29</b>	2	{0,5268,231,8067,1108,1098,1101,1110,8072,4569,5343,5400,5389,5376,5404,5384,5378,5377,5371,5367,5360,5351,5347,5344,5340,2678,2575,2677,5485,5497,5495,5510,5476,0}
	3	{0,2673,2640,2639,2629,2633,2648,2627,2563,2567,2570,2565,2612,2616,2621,2624,2634,2635,2637,2653,2562,2580,2578,5358,5342,5338,5335,5333,5331,5320,5305,5277,4457,0}
	4	{0,2632,2631,2643,2650,2620,2625,2630,2651,5381,2647,5372,5329,5326,5307,5294,5289,5330,5323,5322,5314,5292,5283,2666,5303,5288,2674,2670,2668,2642,2638,2614,2619,0}
	5	{0,5567,5462,5468,5458,5467,5524,2713,5480,5490,2672,2676,5285,5399,5396,5365,5355,5350,5337,5295,2652,5357,5353,5405,5403,5395,5375,5328,2665,2659,2649,2611,5464,0}
	6	{0,5562,5394,5308,5560,2579,5520,2645,2644,2646,4502,5559,5514,5516,5571,5324,5362,5352,5339,5336,5327,5300,5298,5297,5291,5290,5287,5284,2641,2613,5325,5348,5349,0}
	7	{0,5555,5445,5442,2669,2662,2667,2636,5508,5391,5383,5379,5374,5368,5366,5341,5280,2654,5293,5356,5354,2623,2675,5481,2658,2664,2661,2660,2663,2656,2657,2655,2671,0}
<b>30</b>	2	{0,3372,5959,8626,5953,8780,8635,8781,8705,8704,8712,8714,8819,8663,8633,8745,8686,8649,9481,3408,283,3418,5978,3351,3357,7313,3364,3377,6004,6003,5935,6005,5876,0}
	3	{0,8667,8785,7971,7834,7936,7874,7880,8059,4934,1353,1439,1351,1348,1413,8817,1347,8764,8813,8803,8669,8787,8808,8685,8691,8727,7847,8017,1332,8820,1489,7861,7888,0}
	4	{0,8043,7873,8058,7958,8011,8012,8054,8022,1344,7952,1427,1341,1352,1399,1463,1356,1369,1338,1421,1397,1422,1456,1477,1337,1468,1451,1450,1462,1441,1440,7165,8010,0}
	5	{0,8644,8625,8814,8652,8758,8757,8660,8664,8762,8654,8763,8648,8680,8650,8782,8668,8802,8777,8693,8786,8677,8627,8723,8665,8743,8804,8773,8718,8674,8728,8807,8794,0}
	6	{0,5937,5951,6031,5907,5984,5917,5980,6007,5963,5983,5982,5924,6036,6015,6032,5947,6012,5942,5979,5927,5948,5952,6019,5913,6010,5945,6027,3360,9150,5866,7869,3393,0}

	7	{0,8772,8670,8661,7836,7856,8055,7835,8007,7941,8004,8044,7876,7872,4957,7833,8049,7871,8016,7918,7843,7848,7932,7885,8015,7864,7948,8005,7945,8009,8008,8671,8719,0}
<b>31</b>	2	{0,5151,5231,3481,300,195,284,189,7,208,214,42,9,6,8,10,11,14,12,22,5,2,1,54,46,2717,3463,3475,3465,3486,3456,2419,3478,0}
	3	{0,21,26,23,18,15,19,44,43,36,25,49,24,17,45,13,37,20,39,38,35,2479,186,137,3466,3545,3489,3490,2788,3998,3550,3484,2863,0}
	4	{0,182,4491,4459,4486,4557,4505,4585,4514,4540,4497,4563,4468,4586,4481,4524,4496,4539,4538,6795,6730,6825,6848,3549,3506,3599,3116,3118,3542,3130,3471,3508,6746,0}
	5	{0,2480,3528,3504,3476,3107,3128,2424,3604,3603,3512,3541,3511,3581,3488,3551,4354,4397,4355,4356,4396,4357,4359,3454,4404,4300,3513,3609,3473,3469,6823,6824,4027,0}
	6	{0,4558,4578,4465,4464,4510,4476,4488,4523,4542,4469,4470,4527,4528,4549,4541,4546,4482,4477,4489,4511,4543,4467,4573,4521,4525,4526,4487,4503,4498,4504,4466,4499,0}
	7	{0,4517,4478,4532,4471,4574,4533,8086,4545,8099,4490,4493,4501,4544,4582,4507,4579,4458,4515,4462,4461,4583,4530,4516,4460,5914,4548,2484,4506,4520,164,138,55,0}
<b>32</b>	2	{0,5223,5113,5245,5192,5061,5116,5072,5236,5112,5176,5200,5101,5095,4947,5089,5240,5252,5118,5163,5216,5074,5226,5251,5045,5031,5253,5123,5242,5260,4070,4710,4176,0}
	3	{0,4406,4324,4375,4291,4351,4350,4304,4299,4318,4314,4305,3537,3562,3449,3507,3440,3491,3606,3452,3591,3519,3439,3447,3474,3516,3532,3544,4278,4306,4368,4276,4361,0}
	4	{0,5182,5205,5065,5093,5054,5124,5202,5222,5052,5158,5185,5064,5233,5149,5034,5196,5219,5157,5256,5117,5139,5198,6875,3441,161,6808,6760,6880,223,3583,3576,3559,0}
	5	{0,171,140,3487,3457,3510,3514,3560,4340,4342,4339,4279,4341,4283,4280,4387,3531,4353,4395,3612,4034,3611,3605,3589,3587,3598,3561,3578,3529,3582,4290,3496,3530,0}
	6	{0,5030,5154,3588,3601,3438,3495,3610,3594,3505,3595,3596,3563,3498,3546,3565,3602,6794,3548,3564,3593,3509,3577,4316,4392,4417,4325,4343,4413,4349,4373,4393,4310,0}
	7	{0,3547,2448,3525,3590,3503,3615,3540,3574,3580,3518,3569,3502,3477,3521,3597,4295,4399,4409,4419,4416,4400,4418,4315,4360,4407,4410,4321,4348,4301,4402,4307,4401,0}
<b>33</b>	2	{0,1609,1661,241,3356,252,232,251,2327,2352,2402,2401,1594,4,295,270,297,2414,302,7641,285,299,2415,7882,291,2416,280,104,287,4630,8062,4711,4692,0}
	3	{0,5470,5438,5411,5434,5419,5526,5527,5432,5413,5418,5525,5430,5564,5429,5281,5276,5273,5310,5311,5302,5566,5565,5421,5530,5433,5436,5439,5415,5529,5441,5528,5409,0}
	4	{0,5424,5423,5427,8113,5414,5407,5425,5459,5466,5453,5463,5461,5576,5447,5444,5446,5455,5443,5369,5286,2622,2626,5279,5296,5304,5278,5275,5274,5301,5309,5312,5316,0}
	5	{0,5543,5448,5469,6431,5554,5471,5457,6462,5452,5553,3555,5504,5507,5506,6474,3493,5271,3492,5474,5431,5408,3523,5482,5483,5544,3526,3566,6480,5451,5551,5552,6432,0}

	6	{0,5538,3468,5519,5532,5266,5317,5393,5385,5282,2618,2615,5572,5406,5373,5370,5388,5380,5570,5574,5318,5398,5392,5573,5387,5386,5509,6484,5523,3524,5522,5531,5269,0}
	7	{0,4620,4708,261,276,5550,5545,5536,5539,5535,5450,6466,5513,3571,5533,5534,5547,5548,5501,5549,5270,5265,5541,5264,5540,6433,5460,5542,5537,5546,5511,5465,5512,0}
<b>34</b>	2	{0,2277,2843,2340,2397,2363,2324,2332,2319,6578,6599,6597,6596,6709,6600,6598,6612,6606,6605,6601,6579,6693,6586,6582,6581,6584,6585,6583,6591,6592,6590,244,2347,0}
	3	{0,2188,2285,2368,2293,2220,2250,2252,2260,2275,2331,2276,2330,2322,2229,2272,2278,2279,2339,2338,2259,2356,2358,234,249,2388,2389,2410,2306,6473,6475,6467,6464,0}
	4	{0,6468,6448,6477,6470,6525,6530,6527,6524,6523,6526,6696,6535,6520,6537,6622,6617,6481,6626,8085,6603,6607,8078,6609,6604,6610,6608,6613,6611,6482,6478,6453,6457,0}
	5	{0,6528,6529,6531,6533,6519,6548,6547,6543,6542,6540,6544,6561,6559,6566,6563,6570,6564,6571,6572,6573,6574,6568,6569,6553,6554,6552,6551,6555,6577,6575,6556,6576,0}
	6	{0,6522,6549,6546,6538,6545,6539,6541,6550,6624,6690,6620,6557,6562,6565,6616,6654,6649,6652,6567,6688,6558,6560,6636,6637,6634,6665,6618,6640,6659,6633,6631,6536,0}
	7	{0,6629,6695,6621,6639,6638,6627,6644,6642,6662,6635,6660,6682,6676,6672,6647,6669,6675,6674,6657,6632,6630,6619,6614,6628,6623,6580,6589,6588,6602,6587,6594,6593,0}
<b>35</b>	2	{0,7397,7345,7370,7266,7304,7265,7163,7361,7294,7373,7264,7353,7262,7261,7296,7208,7201,7382,7320,7207,7242,7263,7276,7253,770,657,7219,7254,7228,7237,7255,7234,0}
	3	{0,7298,7329,7200,7204,7190,7295,7213,7289,7247,7196,7188,7203,7212,7392,7318,7199,7348,7336,7340,7314,7354,7214,7337,7198,7195,868,689,7333,7326,7360,7344,7288,0}
	4	{0,7347,7285,7341,7303,7244,8283,7346,7278,7316,7378,7322,7321,7389,2890,7325,7365,7343,9424,9402,9456,9417,9404,9394,9415,7284,7332,7260,7259,7387,7338,7257,9299,0}
	5	{0,5666,5754,5667,4956,5702,5729,5691,5686,5588,5683,5722,5586,5793,7393,7380,7279,7286,7339,7381,7312,7324,7243,5744,7371,5791,5737,5583,7335,7330,7377,7300,4951,0}
	6	{0,8134,9331,9405,9362,9347,9332,9340,9344,9343,9393,9342,9396,9431,9432,9357,9358,9454,9412,9301,9300,9430,9303,9328,9436,9435,9437,9368,9439,4963,9425,9387,9304,0}
	7	{0,7287,7297,7305,7277,7308,7192,7311,7210,7283,7364,7376,7205,7398,7193,7310,7290,7258,7383,7209,7211,7355,7334,7366,7236,7327,7194,7187,7306,7356,7351,7319,7268,0}
<b>36</b>	2	{0,5628,5602,5627,7281,7372,5626,5606,4953,8258,8350,8242,8331,8323,8144,8321,8326,7112,7105,7066,7102,6922,7078,8119,7116,8166,8313,7104,8357,7077,8137,8120,5591,0}
	3	{0,9475,8345,6917,6928,6929,6938,6943,6948,6967,6995,6947,7072,7123,7005,7107,7122,7121,7073,7059,7057,7061,7060,8173,7056,6968,7058,7125,8264,8176,8175,5587,5612,0}

	4	{0,5684,5716,5717,5675,5707,5663,5671,5674,5601,5629,5631,5630,5694,5678,5600,5731,5646,6899,7083,7084,7117,6918,7087,7068,7079,6985,8243,6896,7081,6942,7020,5642,0}
	5	{0,5624,5603,8160,8162,8163,8315,8304,8136,8133,8324,8130,8138,8128,8129,8290,8140,8143,8358,8329,8276,8256,8254,8252,8251,8250,8155,8344,8231,8139,8158,865,851,0}
	6	{0,8241,8512,8595,7716,7724,7798,7824,7689,7685,7715,7801,7688,7696,7693,7722,7827,7706,7674,7814,7728,8270,8239,8267,8271,8265,8145,8234,8322,8157,5623,5625,764,0}
	7	{0,8189,8257,7678,8259,7786,7745,7691,7748,7747,7746,7712,7806,7719,7829,7713,7757,7687,7690,7682,7723,7805,7681,7698,7718,7714,7680,7785,7707,7787,7817,8229,9083,0}
<b>37</b>	2	{0,833,754,731,840,713,760,823,806,843,870,846,3762,671,856,732,3622,3680,9467,3790,3856,3818,3834,3707,3811,3702,3718,3705,3628,3789,650,799,691,0}
	3	{0,839,739,721,759,872,910,7161,4961,758,687,886,859,749,862,681,907,810,785,675,738,778,3695,694,678,719,8026,908,813,683,3665,811,803,0}
	4	{0,3855,3711,3738,9505,3701,3816,3825,3732,3753,3746,3713,3822,3743,3797,3792,3744,3742,3696,3799,3714,9355,3768,9327,9366,9317,9370,7293,3643,9322,9530,3756,9326,0}
	5	{0,9308,9367,9348,9312,9389,9325,9324,3764,9329,9337,9330,9338,9321,9336,3645,9323,3835,9363,9335,9373,9452,9334,9410,9440,9318,9315,9371,9360,9399,9316,9351,9447,0}
	6	{0,9333,9361,9365,9356,9352,9297,9434,9397,9449,9350,9448,9306,9395,9443,9307,9382,9313,9305,9345,9451,3647,9409,9359,9311,9411,9349,9310,9416,9401,9309,9302,9341,0}
	7	{0,9429,9428,9353,9354,9445,9427,3754,4954,9446,9433,3745,3720,3785,3722,3734,3700,3698,3740,3829,3821,3737,3690,3710,3748,3763,3741,3771,3717,3715,3786,3716,3823,0}
<b>38</b>	2	{0,8356,8180,8192,8146,8165,8226,5697,5699,5809,3108,5664,5660,5669,7089,5808,5609,5787,5611,5659,5733,5800,5708,5706,5821,5723,8280,8224,8225,4965,8715,5592,5804,0}
	3	{0,8355,8141,8148,7111,7110,8337,8342,3119,8233,8191,8245,8362,8171,8236,5621,8309,8170,8227,8206,8181,8338,8185,8335,8261,8149,8221,8282,8147,8197,8200,8207,8220,0}
	4	{0,6909,7050,6910,6912,6916,6998,7063,7006,6934,5772,7064,8293,8212,8211,8210,8208,8209,4958,8219,8318,8319,8347,8228,8346,8202,7065,7113,6988,7002,7126,7080,6914,0}
	5	{0,7015,7000,8333,8168,8361,8352,8121,8195,8167,8174,8152,8159,8153,8150,8235,6908,8281,8343,8156,8296,8151,7124,7049,6923,8184,8161,5816,5589,5739,5578,8193,8164,0}
	6	{0,7040,7021,7004,7010,6924,7048,7114,6963,7003,7016,7120,8215,8214,7070,7106,7067,8154,8118,6966,7115,6941,6970,7007,6907,7001,5579,6905,7157,4973,7041,7036,7082,0}
	7	{0,5693,5668,5786,5681,5792,5710,7362,7232,7218,7270,7274,7394,7275,7399,7233,7227,7248,7384,7269,7224,7225,7230,7249,7271,7273,5698,5608,5704,5665,5834,7731,5610,0}



<b>39</b>	2	{0,7927,7919,4949,7884,7953,7947,1497,1496,1400,9149,9155,9085,9104,9103,9165,901,7103,7100,793,4964,7101,6962,925,1393,1470,1512,7141,1409,1412,4999,1343,1430,0}
	3	{0,3663,3675,3841,3806,9515,3838,3656,3868,3870,3873,3875,3650,3783,3787,3781,3652,8934,3683,3704,3703,3676,3755,3780,3654,3850,3777,3637,3766,3775,3769,3648,3113,0}
	4	{0,3779,3631,3782,3747,3776,3840,3866,3642,3784,3853,9034,9009,9509,9465,8844,9029,3632,8832,8922,9065,9512,8359,8939,3809,9076,9075,9062,8936,3664,8940,3761,9461,0}
	5	{0,9063,9070,9069,8836,8959,8917,8916,8918,8855,9072,8851,8856,9068,8830,9056,8837,8852,8828,8834,8921,9077,8835,8950,8969,8919,9073,9499,9468,8897,8953,8999,8889,0}
	6	{0,9319,9414,9339,9320,3758,3640,3750,3774,3634,3810,3677,3633,3635,3849,3803,3801,3657,3638,3863,3864,3837,3842,3854,3845,3669,9497,3636,3699,3851,3852,3641,9504,0}
	7	{0,9096,9287,9271,9101,9274,9273,9156,9114,9134,8925,8993,8987,9074,8979,8980,8985,9033,8924,9217,9216,9084,8923,9036,8988,9236,9229,9275,9492,9205,9204,9218,9539,0}
	<b>40</b>	2
3		{0,3869,3729,3843,3725,9460,9525,9177,8974,3752,3673,3765,3679,3684,3651,3862,3678,3689,3739,8975,3653,3672,3846,3644,3857,8957,9513,9472,3865,9123,9097,9197,9113,0}
4		{0,3697,3772,3773,3666,3692,3788,3833,3727,9527,3687,9523,3867,3668,3655,3767,3757,3660,3794,3795,3819,3723,3736,3831,3724,3735,3726,3731,3693,3824,8973,3791,3812,0}
5		{0,9176,9192,9185,9087,9162,9253,9112,9126,9121,9219,9127,9158,9122,9120,9529,9292,9125,9111,9245,9163,9474,9093,9209,9092,7132,9459,8080,9532,9089,4970,9147,9128,0}
6		{0,6949,9478,9473,7098,9220,9270,9167,9166,9249,9269,9286,9169,9212,9168,9213,9243,9268,9247,9294,9458,9211,9244,9102,9203,9470,9248,9242,9086,9193,9088,6958,7097,0}
7		{0,9482,9181,9160,9100,9098,9284,9135,9172,9507,9228,9252,9173,9222,9145,9109,9258,9276,9254,9140,9110,9293,9259,9278,9277,9119,9195,9246,9503,9106,9091,9201,3759,0}
<b>41</b>		2
	3	{0,2005,1978,1984,1982,2141,2070,2015,2045,2003,1957,2118,2082,2026,5908,5872,5955,6033,5986,5864,5868,5856,5867,5854,2028,2193,1958,2027,2144,1959,2138,1960,2096,0}
	4	{0,4926,4897,4714,4718,4921,4736,4769,5997,4723,4725,4812,4817,4824,4882,2051,2140,2114,1964,2149,2007,1966,1991,1965,2033,1968,2157,2041,2048,3415,4712,4904,4803,0}
	5	{0,1967,1979,2037,2135,2143,2047,1980,2133,2165,2084,2100,2186,2126,2024,2025,2004,1987,2150,1993,4968,6913,7903,7902,7153,7129,5853,2085,1986,2159,2137,2187,2042,0}

	6	{0,4927,4772,4774,4773,4874,4746,4916,4762,4777,4811,4837,4924,4778,4728,4806,4902,4842,4779,4899,4820,4727,4853,4839,4776,4775,4805,4931,4906,4713,4838,4813,4801,0}
	7	{0,8398,8426,8424,8412,8382,8363,8383,8597,8409,8593,8408,8413,8390,8429,8385,8377,8440,8435,8387,8433,8393,8596,8425,6925,8432,8589,8434,8386,8585,8407,6969,8586,0}
<b>42</b>	2	{0,8060,4960,9519,9502,9489,1390,1459,1394,1370,1500,9239,9200,9174,9157,9241,9196,9107,9289,9198,9184,9099,9221,9153,9161,9224,9171,9251,9141,3861,9250,9471,7954,0}
	3	{0,3839,3872,9010,9054,9045,9059,9006,9007,8845,8997,8933,9031,8996,8995,8956,9004,8941,8893,8990,9021,8944,8991,9012,8931,8930,8947,9003,8868,8903,8970,8872,8848,0}
	4	{0,9057,8821,3844,8846,9050,9008,8882,8881,8887,8880,8879,8891,8888,8920,9018,8901,8986,8915,8874,8886,8955,8932,8829,8928,8964,9026,9066,8883,9520,9055,8890,9051,0}
	5	{0,9288,9272,9215,9207,9206,9494,9208,9132,9152,9143,9142,9137,9095,9094,9129,9232,9265,9230,8862,9280,9225,9256,9015,9234,9233,9260,9261,9262,9263,8983,8967,9071,0}
	6	{0,8951,8966,8968,8896,9042,8826,8823,9040,8858,9016,8867,9002,8849,8938,8912,8898,8902,9019,9080,8984,8952,8948,8906,8907,9035,9049,8908,9037,8857,8992,9039,9041,0}
	7	{0,9048,9067,8937,9081,8840,8833,8982,8895,9000,9001,8831,9082,8977,8971,9052,9053,8989,8884,9011,8873,8929,9024,9020,9047,8981,8926,8911,9038,9028,8963,8900,9226,0}
<b>43</b>	2	{0,2878,2783,2842,2766,2815,2849,2910,2792,2794,2835,2750,2722,2874,2861,2773,7013,7148,7135,6992,6999,7033,7026,6915,7017,6973,6990,7181,7136,6980,7180,5561,2860,0}
	3	{0,6815,6814,6813,6812,6869,6883,6862,3055,6776,2892,2735,5884,2804,2816,2761,2791,2778,2762,2826,2898,2894,2911,2909,2889,7022,2891,2726,2858,2749,2776,2780,2756,0}
	4	{0,2827,7154,2770,2828,2775,2790,2875,2868,2741,2836,2883,2758,2859,2789,2825,2786,2817,2821,2764,2844,2760,2879,2784,2813,7031,7034,7029,2774,2769,2785,2807,2754,0}
	5	{0,6854,6860,6861,7175,6804,6874,6816,6764,6805,6890,6778,6780,6781,6783,2923,6425,3034,3089,3062,6774,3030,7176,3047,3027,6871,6839,6847,6838,6857,6833,6784,6806,0}
	6	{0,6975,7018,8421,8375,8549,7023,7028,6971,7030,6982,8447,8364,8365,8389,8411,8400,8419,6944,7014,7044,7039,7052,6901,6940,7042,6939,8222,8183,8272,8223,8190,5701,0}
	7	{0,8551,8381,8397,8384,8371,8420,8496,8448,8376,8514,8379,8391,8414,8374,8403,8366,8519,8380,7828,7374,7831,7803,7267,7804,7385,7256,7215,7391,7854,7386,7350,8563,0}
<b>44</b>	2	{0,7823,7703,7704,8260,7821,7700,7795,7701,7711,8188,8186,7820,7721,7702,7710,7759,7705,8196,7807,8468,8469,8580,8446,8472,8515,8517,8497,8470,8518,8516,8584,7761,0}
	3	{0,9384,3658,9364,3770,9346,3639,3646,3662,3659,8473,8476,8444,8475,8474,8445,8564,8609,8477,8443,8471,8442,7818,8605,7796,7774,7794,7732,7752,7734,7751,7777,7802,0}

	4	{0,5705,5640,5767,5648,5598,5734,5634,5647,5732,5818,5670,5817,5752,7388,7363,7379,8452,7775,7776,7736,7735,7676,9379,9381,9380,9386,9298,9369,9418,9374,9403,9378,0}
	5	{0,5651,5652,5009,5622,5827,5643,8334,5649,5828,5650,5712,5813,5597,8266,8255,8232,7840,5730,5655,5657,5658,5814,5812,8253,8284,5594,5654,5811,5593,5764,5599,5771,0}
	6	{0,8240,8238,7725,7729,7684,7797,7753,7793,7717,7750,7741,7740,7686,8070,7739,7738,7737,7742,7756,7815,8286,8287,7799,2900,7792,7744,7743,7726,7673,7754,7749,7781,0}
	7	{0,8466,8478,8493,8464,8487,8604,8485,8483,8484,8480,8486,8601,8482,8599,8488,8481,8492,8491,8490,8479,8489,8600,8499,8494,8462,8498,8451,8467,8450,8522,8465,8463,0}
<b>45</b>	2	{0,4607,4602,4592,4603,4628,4605,8038,4621,6129,4616,6155,6156,6186,7988,4598,7959,7990,8057,7983,7991,7992,7960,8046,8047,7986,8048,7980,7964,7985,7981,7963,7951,0}
	3	{0,8036,7965,7982,7962,7987,7961,4623,8037,7984,7967,7989,6128,6190,7537,6068,6114,6115,6116,6231,6174,6122,6121,6125,6117,6118,6119,6126,6237,6127,6120,6218,7966,0}
	4	{0,6078,8553,8610,8579,6079,6160,6197,6172,6065,7789,6159,6141,6081,6158,6207,6080,6233,7816,7790,7764,7765,7770,7763,7772,7822,7825,7808,7773,6061,7788,7783,4604,0}
	5	{0,8529,8531,8533,8534,8528,8554,8525,8524,8523,8527,8594,8559,8520,8513,8568,8532,8526,6130,7635,7636,7630,7634,7612,7627,7629,7550,7638,7628,7579,7594,7578,6084,0}
	6	{0,6220,6077,6157,6221,6149,6217,6142,6196,8045,6187,6150,6151,6230,6154,6152,6205,6153,6229,6241,7524,7518,7559,7536,7552,7602,7640,7558,7516,7522,7520,7614,7519,0}
	7	{0,8536,4618,6123,6069,6124,6067,4593,6206,6076,6073,6072,6232,6140,6074,6075,7766,7771,7782,7810,7809,7768,7767,7762,7769,8608,8540,8539,8556,8538,8537,8530,8535,0}
<b>46</b>	2	{0,5994,5995,5993,5992,5977,5970,2847,2885,2873,2876,259,8179,6997,8194,6904,7144,7140,6991,7008,4627,2748,2830,2730,7019,2834,6895,2882,2862,2718,2823,2757,2822,0}
	3	{0,7177,4959,6758,6841,6744,6777,6832,6779,6846,6782,6803,6818,6845,6884,6817,6802,6891,7168,6865,6766,6769,6762,6775,3068,8726,3059,3070,3092,6763,2893,5998,6017,0}
	4	{0,6894,8422,8441,7009,8607,6965,6897,6993,8368,8370,8392,8378,8416,8418,8417,8369,8410,8415,7720,7813,8367,7692,7730,4617,4589,4609,4611,4596,4615,4622,7733,7779,0}
	5	{0,7800,4613,4601,4595,4599,8172,7755,7791,7677,8399,7683,7780,7708,7699,7758,7695,7819,7811,7826,7830,7675,7709,7694,7679,7727,7784,7697,7760,7778,4594,9488,4600,0}
	6	{0,6920,6976,7071,6933,6906,6927,6979,6978,7043,6926,6911,7119,6964,8336,8230,8178,8177,8327,8244,5619,4971,8340,8339,8263,8273,8300,8341,8351,8294,8169,8349,8328,0}
	7	{0,8430,8405,6931,8707,8274,8268,8285,7038,7118,7055,6972,6062,7045,7037,7131,6921,6984,5014,8406,8495,6900,6936,8372,8373,6987,8401,6937,6919,6989,7011,6930,6983,0}

<b>47</b>	2	{0,6047,4619,9464,4590,8573,8574,8576,8500,8572,8575,8571,8578,8577,8602,8570,8606,8544,8547,8569,7402,8461,8455,8453,8460,8501,8456,8508,8502,8506,8457,8598,8591,0}
	3	{0,4597,4624,4625,7592,7616,7468,7478,7599,7565,7472,7598,7534,7548,7547,7475,4614,7488,7606,7596,7617,7624,7480,7474,7593,7482,7512,7466,7465,7464,7418,7605,7609,0}
	4	{0,7409,7589,4612,7481,7570,7404,7456,7411,7413,7571,7452,7457,7455,7415,7469,7454,7407,7414,7556,7416,7575,7566,7479,7539,7406,8248,7494,7530,7582,7484,7458,7487,0}
	5	{0,7489,7504,7503,7500,7509,7597,7600,7611,7604,7505,7567,7491,7502,7549,7461,7471,7501,7541,7467,7525,7538,7470,7462,7490,7510,7511,7477,7460,7551,7476,7526,7560,0}
	6	{0,7459,7499,7410,7572,7486,7527,7419,7495,7463,7473,7483,7453,7498,7506,7542,7408,7528,7451,7529,7412,7590,7496,7557,7561,7497,7493,7591,7585,7405,7417,7603,7492,0}
	7	{0,9450,8458,8454,8587,8505,8459,8511,8583,8565,8555,8566,8510,8567,8582,8509,8550,8504,8507,8548,8603,8581,8503,9444,9314,9413,9438,9426,9422,9406,9392,9442,9421,0}
	<b>48</b>	2
3		{0,6946,7012,8204,6994,6986,7062,7074,6898,7075,7076,5585,6945,6935,8316,8305,7069,6893,7054,7053,7108,8117,8131,8125,8217,8303,8216,8198,8317,8360,6996,8353,8262,0}
4		{0,8132,8310,8308,5653,5595,5596,8142,8292,8275,8269,8307,5741,5769,8279,8122,8291,8306,8299,8301,8312,8332,8354,8325,8289,8278,7172,8237,8297,8348,8123,8311,5745,0}
5		{0,5638,5795,5685,5725,5746,5753,5720,5783,5775,5641,5776,5832,5777,7302,5778,5780,5715,5781,5782,5718,5726,5719,5830,5820,5819,5784,5690,5672,5635,5831,5794,5632,0}
6		{0,8320,8277,8127,8199,8314,8124,9493,5835,5662,5661,5677,5673,5803,5762,5807,5728,5790,8126,5676,8302,8187,5760,8201,8288,8247,8203,8205,8218,8213,8135,8298,5823,0}
7		{0,5680,5713,5679,5743,5692,7197,7400,7202,7349,7375,7331,7317,5766,5711,5689,5727,5763,5735,5639,5779,7323,7185,7184,7282,7301,7280,7328,7368,7369,7395,7183,7186,0}
<b>49</b>		2
	3	{0,5724,5773,5750,5774,9400,9376,9390,9453,9372,9377,9420,9388,9419,9441,9385,9375,9423,9383,9407,9398,9408,9391,9455,6045,6044,6198,6107,6147,6105,6106,6042,8182,0}
	4	{0,6112,6222,6200,6055,6056,7485,6113,6202,6184,6201,6071,7531,6058,6046,7584,7421,7439,7610,8542,8561,8541,8558,8557,8560,8590,8521,8543,8552,8546,8562,8545,8592,0}
	5	{0,7607,7580,7581,7432,7437,7426,7436,7621,7564,7563,7423,7403,7438,7533,7620,7622,7513,7515,7425,7573,7543,7546,7619,7608,7521,7535,7618,7514,7586,7440,7587,7532,0}

	6	{0,6173,6093,6223,6090,6091,6182,7562,6199,6089,7553,6088,7626,7623,7625,7631,7588,7577,7576,7595,7632,7639,7555,7554,7637,7633,7615,6087,7583,6092,6242,6086,6085,0}
	7	{0,7424,7545,7420,7430,7429,7428,7601,7427,7568,7431,7574,7422,7613,7433,7434,7435,7507,7544,7441,7450,7517,7569,7442,7449,7448,7508,7523,7445,7444,7446,7443,7447,0}
<b>50</b>	2	{0,2455,2457,2472,2456,2482,2426,2444,2431,2432,2438,2467,2427,2460,2458,6892,6831,6849,6796,6826,6797,6827,2459,0}
	3	{0,4381,4352,4391,4382,4323,4383,4408,4377,4055,4041,3920,3980,4030,2434,4042,3918,3970,4044,3968,4051,4298,4049,0}
	4	{0,3613,3614,6828,6798,6799,4551,2440,2435,2483,2474,2462,2463,2486,4384,4385,2461,4379,4293,4029,3899,4036,4050,0}
	5	{0,4346,4308,4317,4358,4345,4347,4313,3616,3554,3617,3573,3553,6750,6751,6747,6887,6748,6749,6765,6850,6800,6879,0}
	6	{0,2441,4311,4365,4344,4294,4303,4378,4023,4018,4370,4369,2451,4388,4288,4386,4389,4411,2428,4412,4362,4380,2445,0}
	7	{0,2473,4024,4028,2477,2476,2454,2465,2470,8068,2487,2485,2429,2430,2433,2439,2442,7672,2436,4277,2475,2469,0}

## APÉNDICE D

Distancia total recorrida por ruta

<b>Ruta</b>	<b>Día</b>	<b>Recorrido (km)</b>
<b>1</b>	2	29,3
	3	37,3
	4	29,1
	5	22,4
	6	17,7
	7	19,2
	<b>2</b>	2
3		17,8
4		18,2
5		14,3
6		15,8
7		13,8
<b>3</b>		2
	3	85,8
	4	56,2
	5	66,4
	6	50,6
	7	93,5
	<b>4</b>	2
3		11,1
4		14,1
5		11,5
6		14,3
7		15,3
<b>5</b>		2
	3	32,1
	4	25,3
	5	16,4
	6	12,1
	7	28,0
	<b>6</b>	2
3		11,5
4		13,0
5		13,4
6		13,1
7		13,2
<b>7</b>		2
	3	32,4
	4	31,2
	5	33,0
	6	33,9
	7	33,6

<b>8</b>	2	13,0
	3	12,4
	4	15,1
	5	13,1
	6	15,5
	7	10,5
	<b>9</b>	2
3		26,8
4		29,8
5		30,4
6		26,6
7		27,8
<b>10</b>		2
	3	27,8
	4	18,6
	5	20,4
	6	22,7
	7	21,2
	<b>11</b>	2
3		11,7
4		9,8
5		12,0
6		10,7
7		14,0
<b>12</b>		2
	3	19,3
	4	10,7
	5	15,7
	6	12,1
	7	11,5
	<b>13</b>	2
3		4,9
4		8,3
5		10,9
6		7,8
7		10,5
<b>14</b>		2
	3	84,4
	4	46,8
	5	29,4
	6	33,9
	7	44,7
	<b>15</b>	2
3		102,4
4		78,0
5		120,1

	6	80,3
	7	113,9
<b>16</b>	2	141,0
	3	168,2
	4	296,4
	5	240,8
	6	244,6
	7	256,9
<b>17</b>	2	203,6
	3	197,8
	4	226,6
	5	243,3
	6	146,6
	7	273,8
<b>18</b>	2	100,9
	3	112,4
	4	41,3
	5	46,5
	6	45,6
	7	54,0
<b>19</b>	2	57,9
	3	160,7
	4	213,6
	5	55,4
	6	48,2
	7	256,3
<b>20</b>	2	257,4
	3	258,6
	4	47,3
	5	367,0
	6	53,4
	7	45,5
<b>21</b>	2	46,8
	3	48,6
	4	41,9
	5	43,2
	6	48,8
	7	38,3
<b>22</b>	2	24,5
	3	25,7
	4	24,6
	5	22,6
	6	23,7
	7	24,8
<b>23</b>	2	30,4
	3	31,0



	4	34,6
	5	33,6
	6	30,5
	7	35,4
<b>24</b>	2	27,3
	3	27,8
	4	28,7
	5	26,2
	6	25,7
	7	28,7
<b>25</b>	2	33,4
	3	31,8
	4	28,5
	5	31,4
	6	29,6
	7	28,5
<b>26</b>	2	19,9
	3	89,3
	4	85,1
	5	87,3
	6	15,4
	7	141,7
<b>27</b>	2	15,8
	3	14,4
	4	23,7
	5	50,4
	6	15,6
	7	15,1
<b>28</b>	2	15,4
	3	10,1
	4	10,7
	5	8,2
	6	11,8
	7	7,8
<b>29</b>	2	35,9
	3	39,2
	4	37,0
	5	32,7
	6	37,7
	7	36,0
<b>30</b>	2	23,9
	3	26,2
	4	25,7
	5	23,1
	6	20,9
	7	24,7

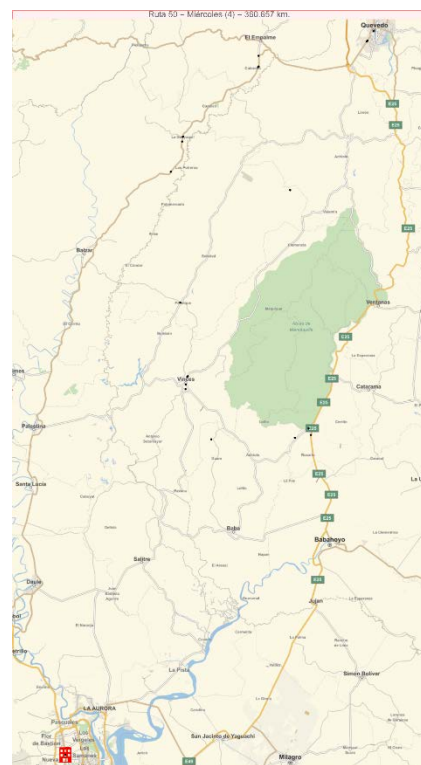
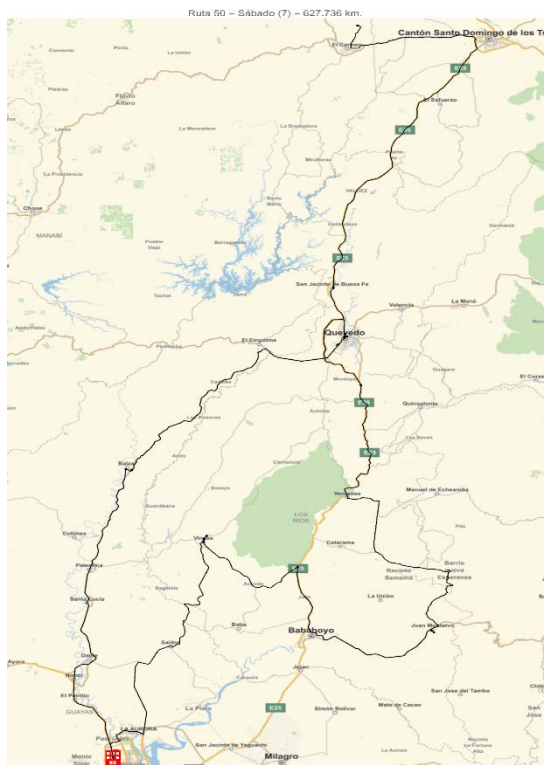
<b>31</b>	2	127,2
	3	137,3
	4	212,9
	5	187,5
	6	247,0
	7	135,4
	<b>32</b>	2
3		193,0
4		60,5
5		127,5
6		103,5
7		149,4
<b>33</b>		2
	3	32,9
	4	34,3
	5	30,5
	6	34,7
	7	30,8
	<b>34</b>	2
3		32,8
4		33,7
5		35,6
6		36,4
7		37,1
<b>35</b>		2
	3	30,5
	4	31,5
	5	26,2
	6	32,7
	7	28,2
	<b>36</b>	2
3		32,4
4		28,4
5		32,1
6		69,3
7		80,8
<b>37</b>		2
	3	29,8
	4	35,1
	5	39,1
	6	34,0
	7	34,9
	<b>38</b>	2
3		22,4
4		21,9
5		23,0

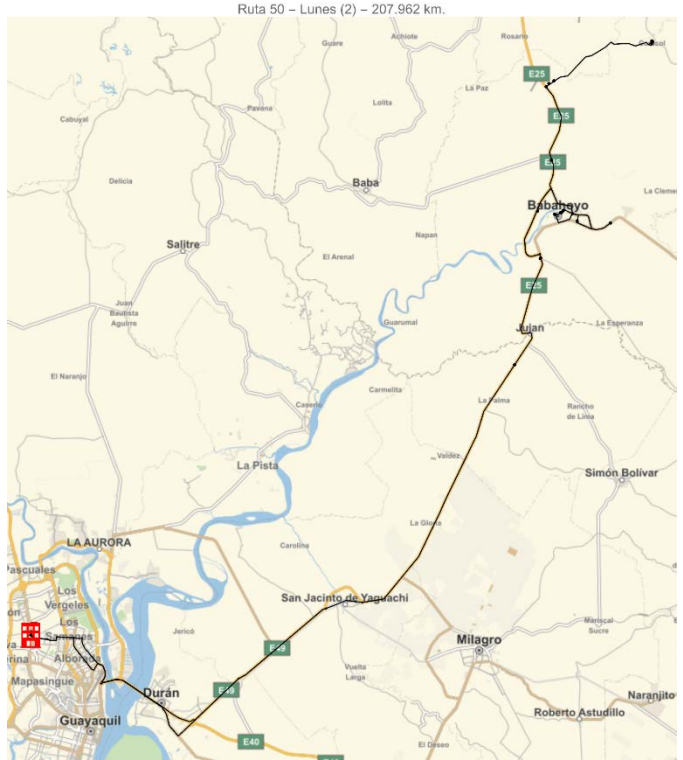
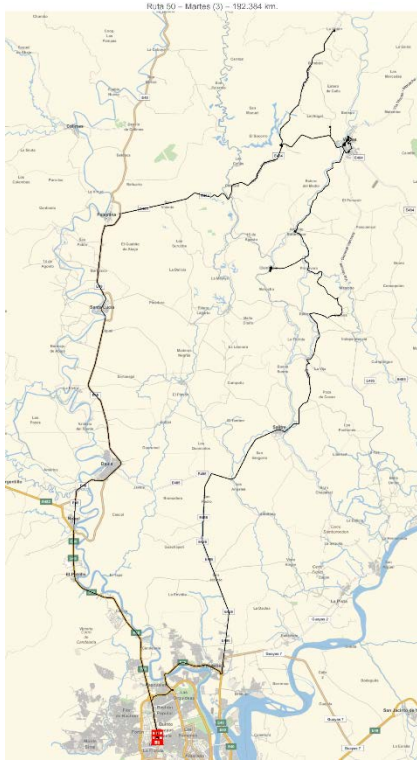
	6	22,2
	7	25,0
<b>39</b>	2	40,5
	3	41,2
	4	44,4
	5	43,5
	6	40,1
	7	40,8
<b>40</b>	2	34,1
	3	41,3
	4	36,9
	5	38,7
	6	36,8
	7	40,6
<b>41</b>	2	21,4
	3	11,1
	4	9,2
	5	11,4
	6	5,2
	7	17,3
<b>42</b>	2	41,2
	3	48,0
	4	45,3
	5	44,0
	6	45,9
	7	44,7
<b>43</b>	2	20,1
	3	21,7
	4	11,5
	5	13,7
	6	23,1
	7	24,0
<b>44</b>	2	145,9
	3	104,9
	4	76,6
	5	29,7
	6	74,6
	7	133,6
<b>45</b>	2	254,0
	3	258,4
	4	238,1
	5	248,0
	6	279,7
	7	262,4
<b>46</b>	2	27,9
	3	21,9

	4	31,8
	5	41,0
	6	21,0
	7	23,3
<b>47</b>	2	192,4
	3	238,6
	4	216,6
	5	219,7
	6	219,7
	7	155,5
<b>48</b>	2	28,8
	3	25,4
	4	28,0
	5	26,5
	6	27,5
	7	28,3
<b>49</b>	2	288,3
	3	203,1
	4	228,7
	5	265,5
	6	203,0
	7	260,7
<b>50</b>	2	208,0
	3	192,4
	4	360,7
	5	187,9
	6	252,6
	7	627,7
<b>DISTANCIA TOTAL RECORRIDA (km)</b>		20346,0

# APÉNDICE E

## Gráficas de recorrido por ruta y por día





Ruta 49 - Sábado (7) - 260.687 km.



Ruta 49 - Viernes (6) - 203.004 km.



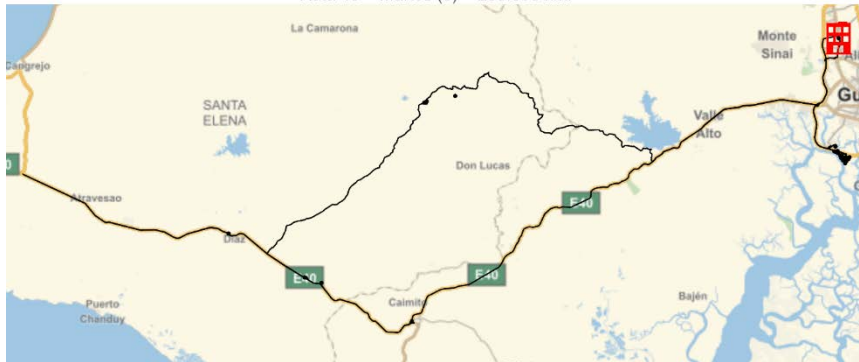
Ruta 49 – Jueves (5) – 265.511 km.



Ruta 49 – Miércoles (4) – 228.683 km.

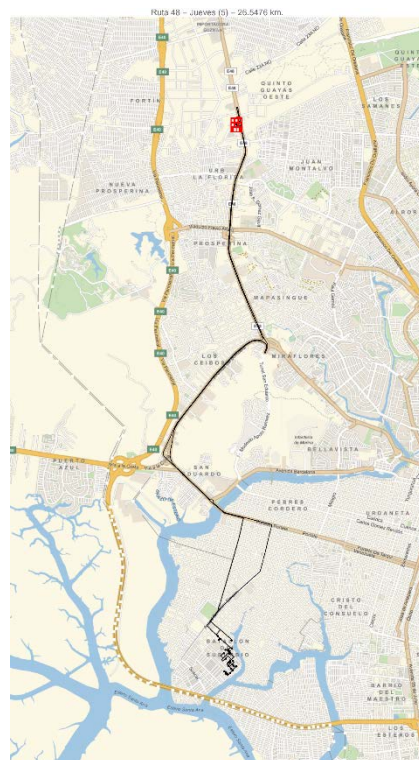
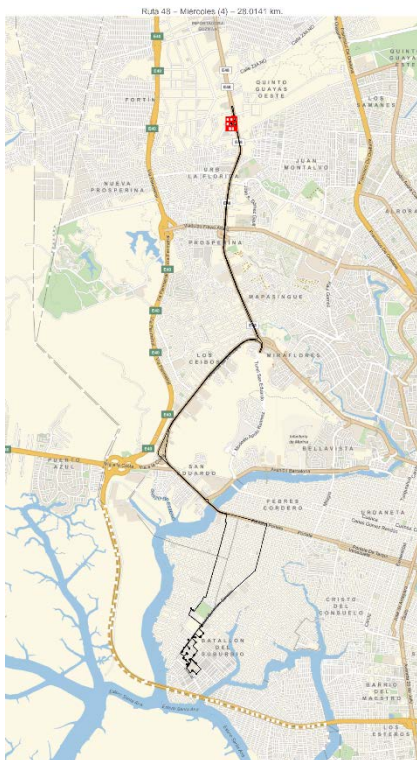
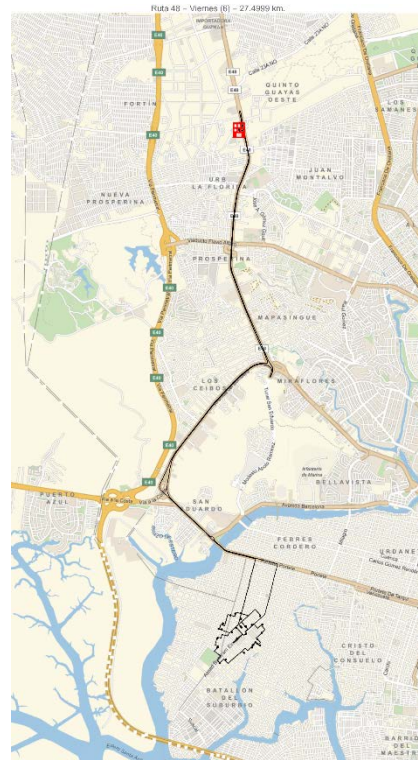
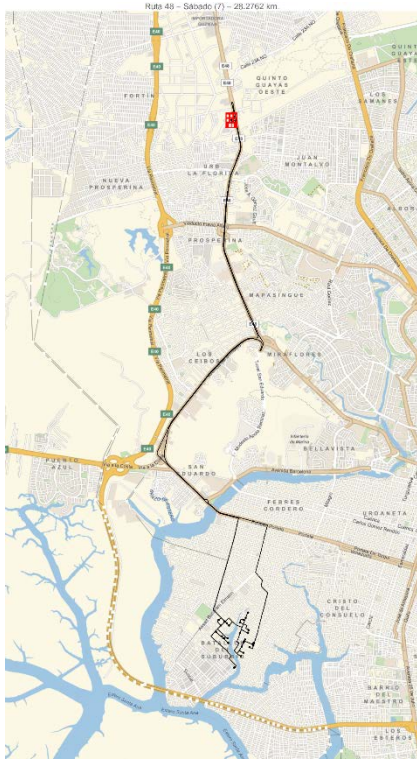


Ruta 49 – Martes (3) – 203.096 km.

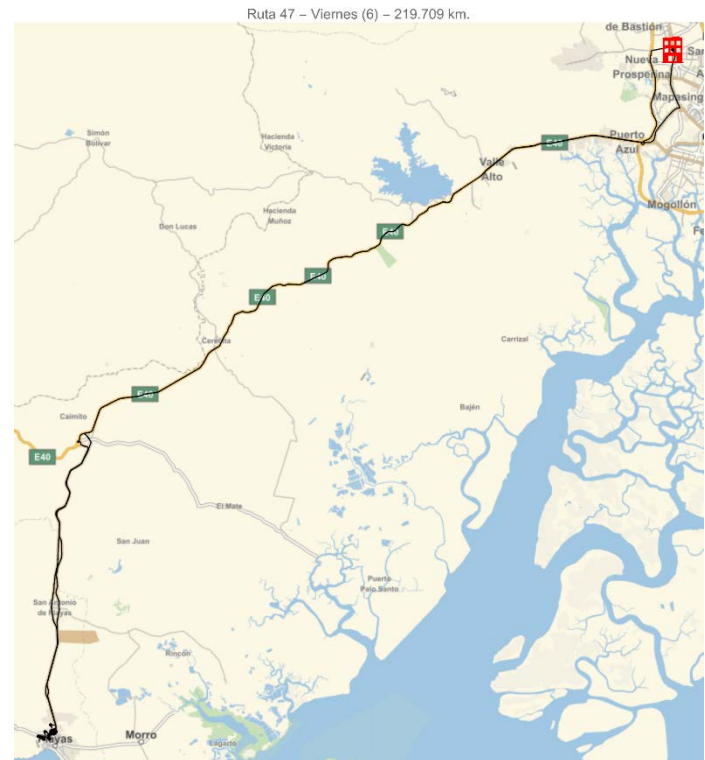
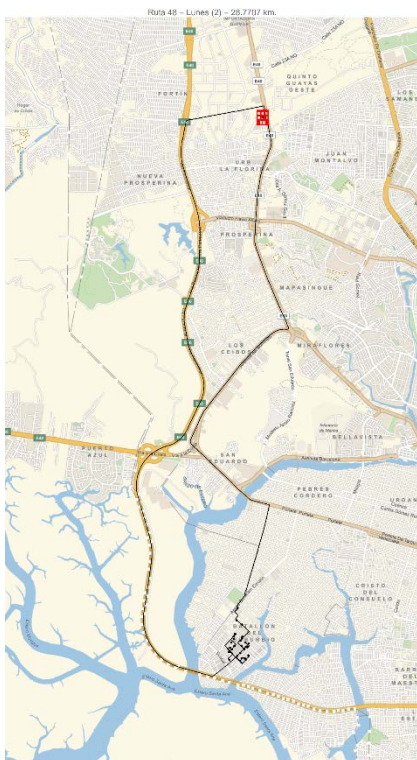


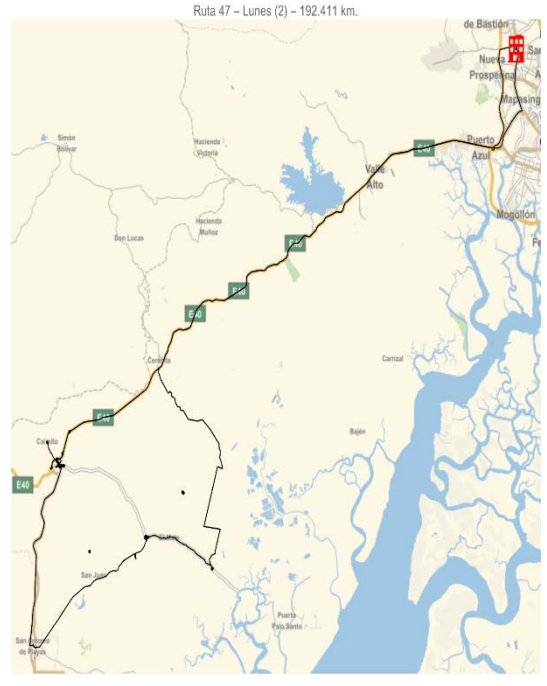
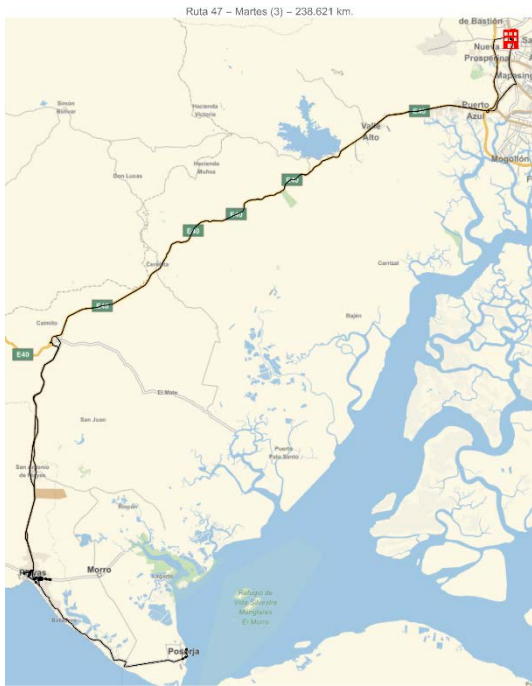
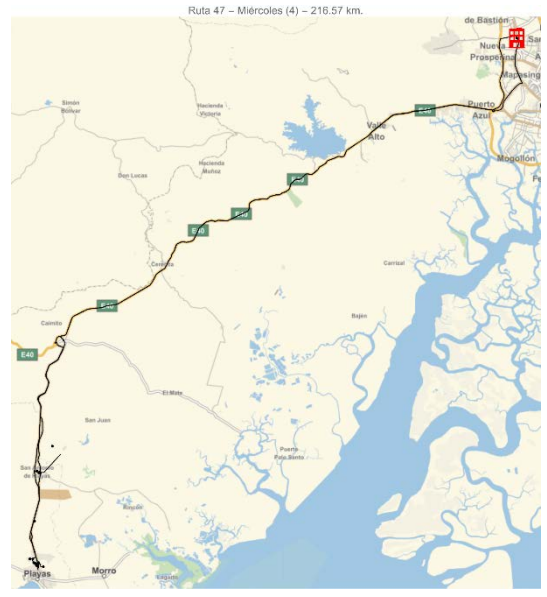
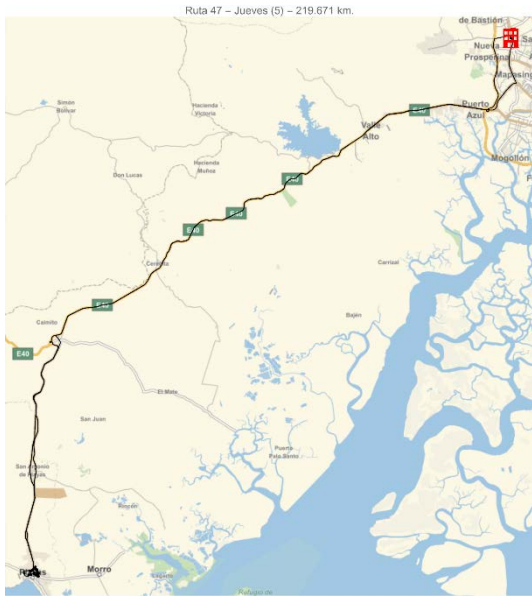
Ruta 49 – Lunes (2) – 288.333 km.

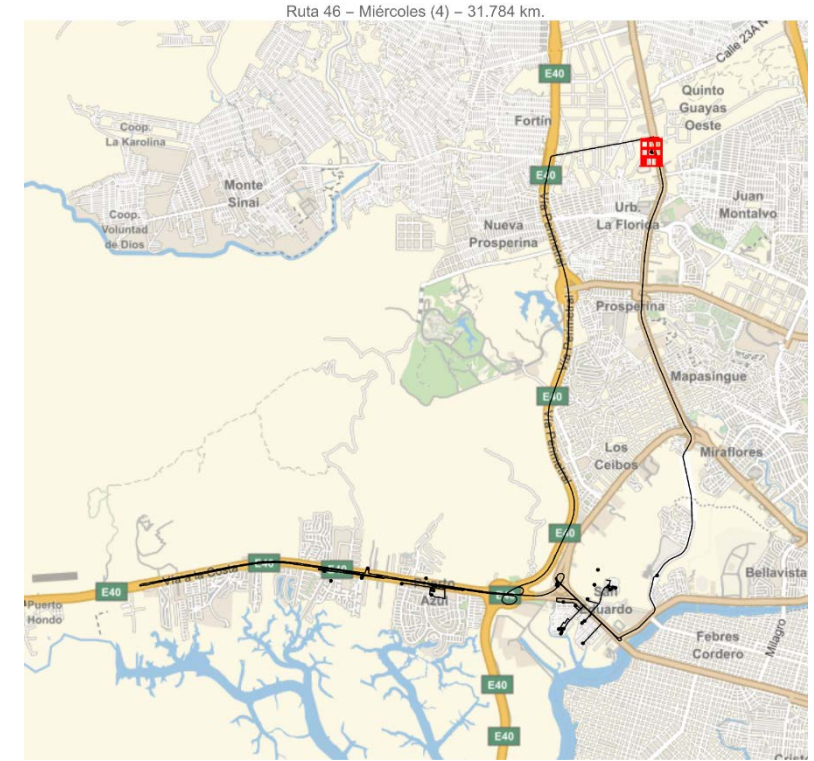
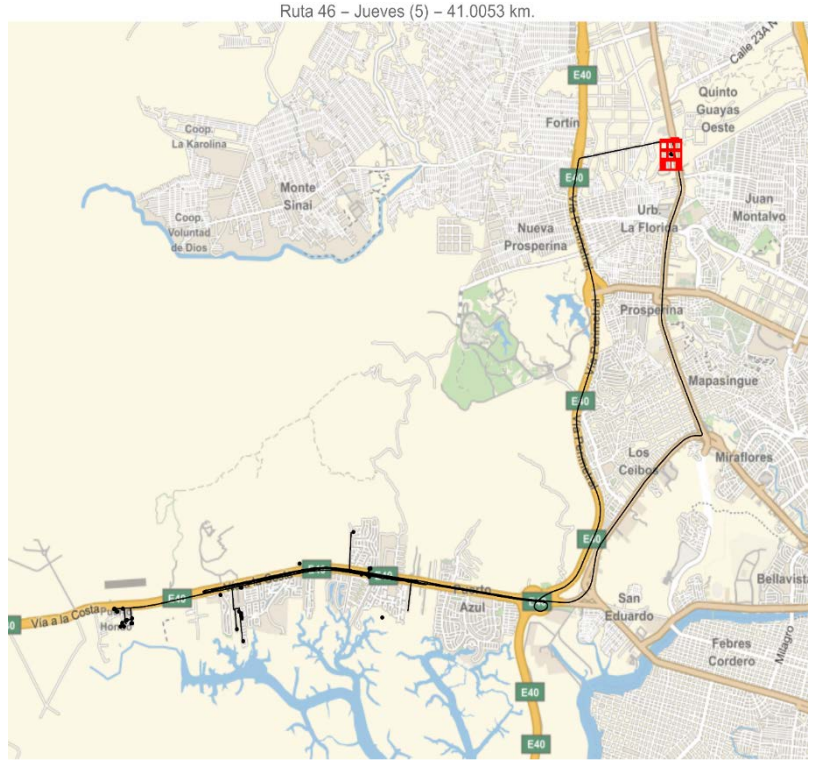




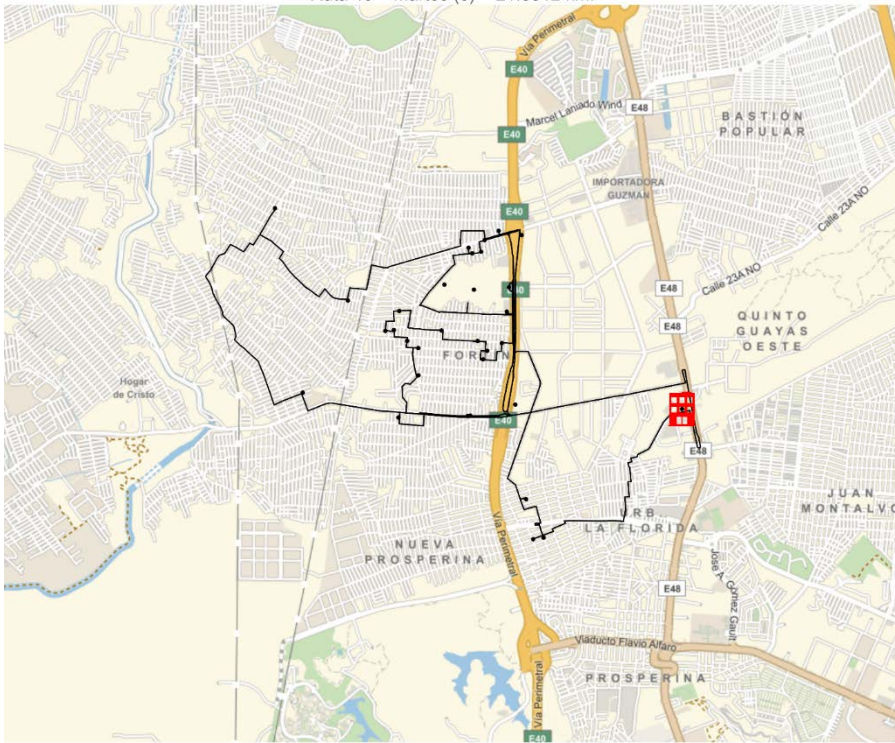




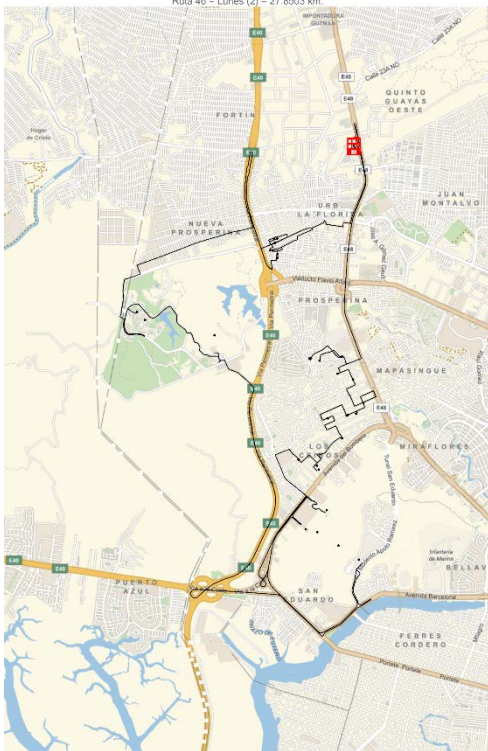




Ruta 46 - Martes (3) - 21.8512 km.



Ruta 46 - Lunes (2) - 27.8603 km.



Ruta 45 – Viernes (6) – 279.673 km.



Ruta 45 – Sábado (7) – 262.412 km.



Ruta 45 – Jueves (5) – 248.022 km.



Ruta 45 – Miércoles (4) – 238.062 km.



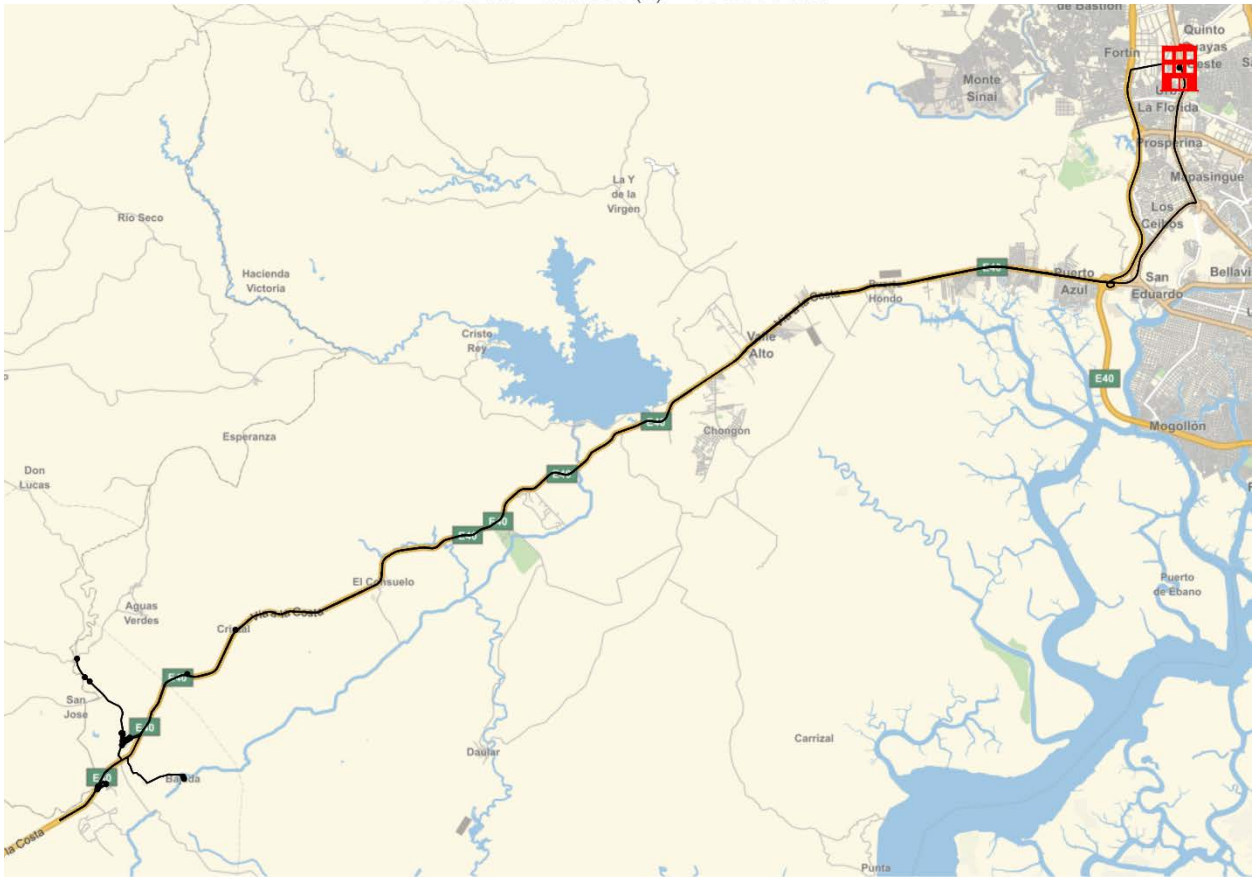
Ruta 45 – Lunes (2) – 253.964 km.



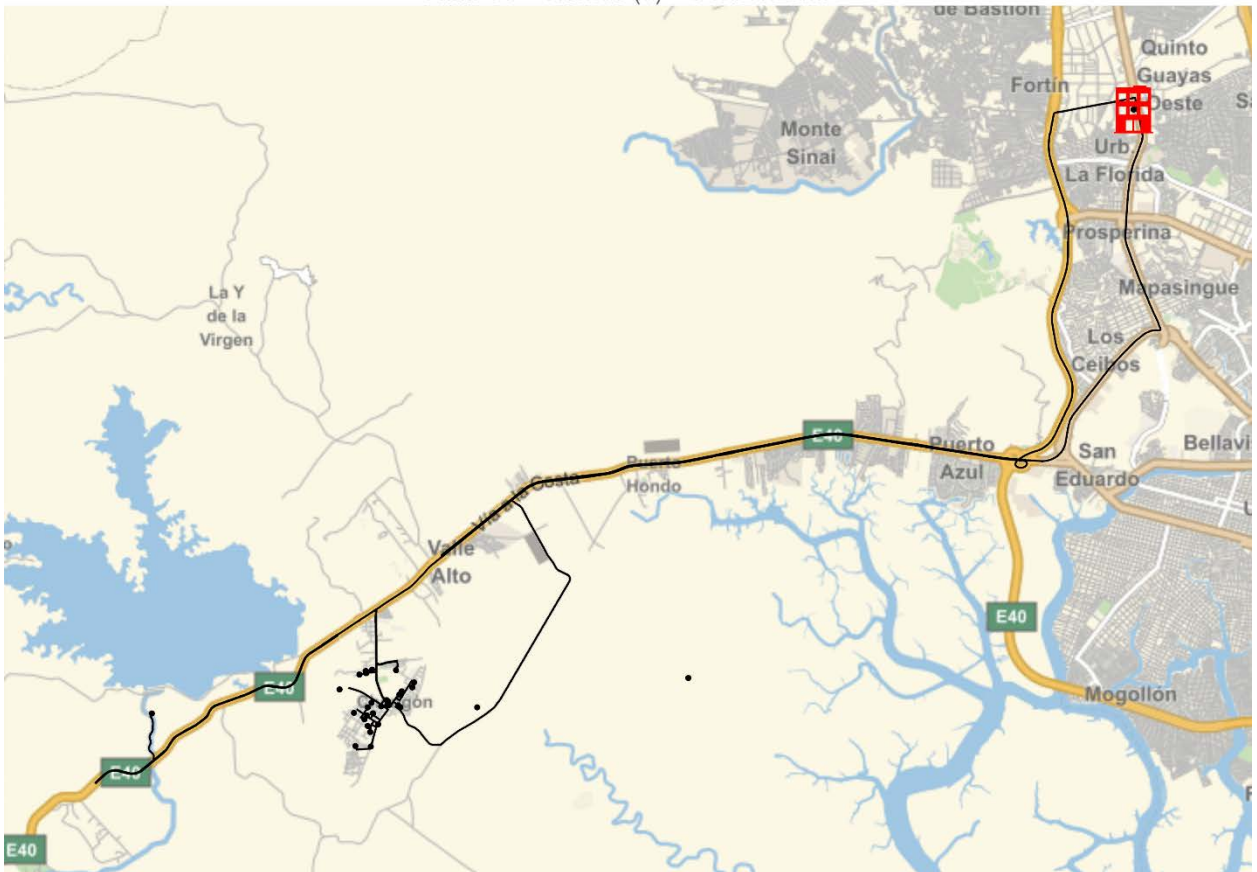
Ruta 45 – Martes (3) – 258.357 km.

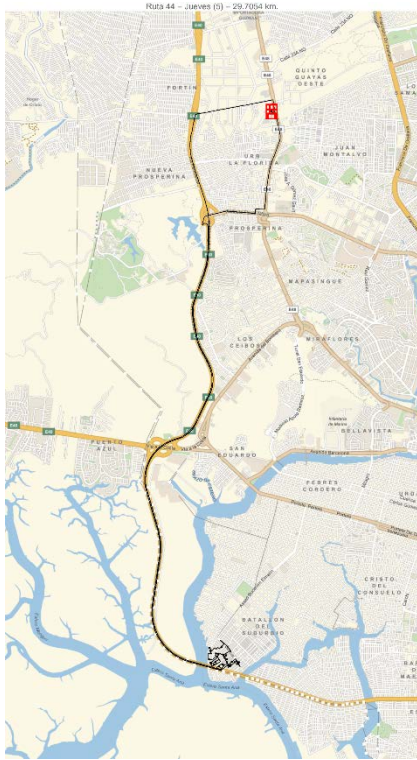


Ruta 44 – Sábado (7) – 133.601 km.

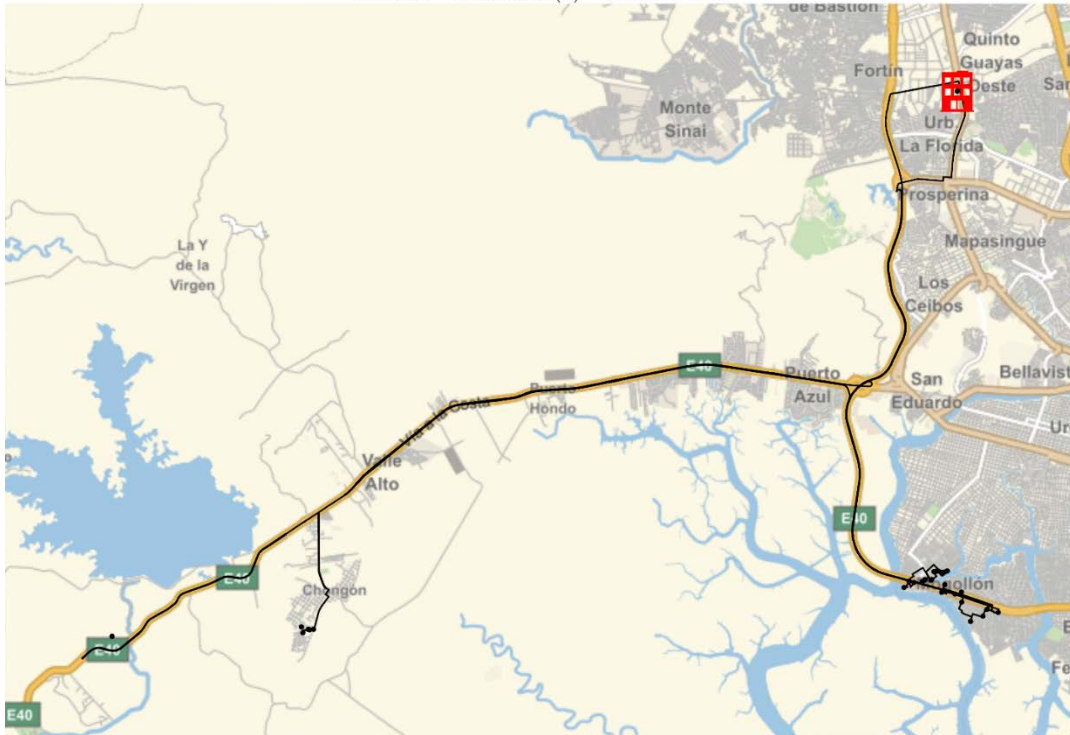


Ruta 44 – Viernes (6) – 74.6247 km.





Ruta 44 - Miércoles (4) - 76.5616 km.



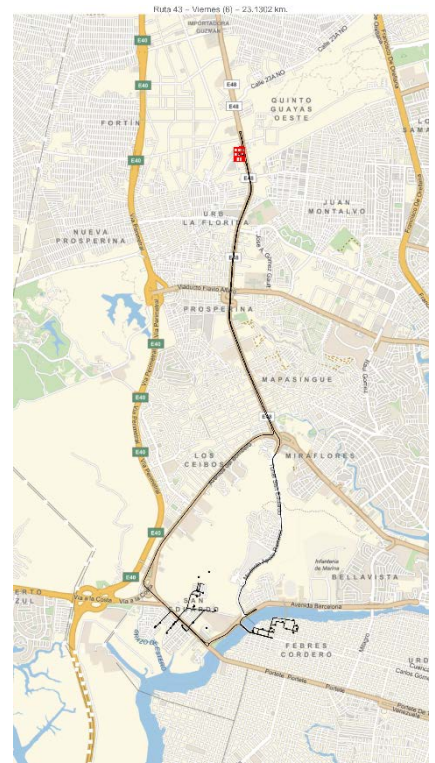
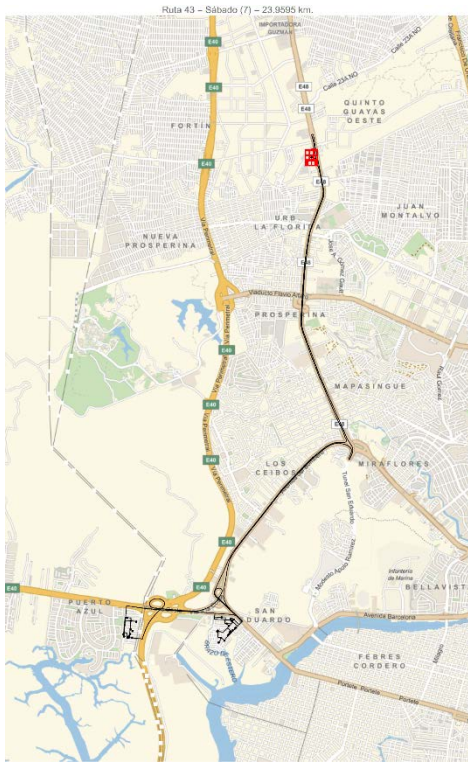


Ruta 44 – Martes (3) – 104.945 km.

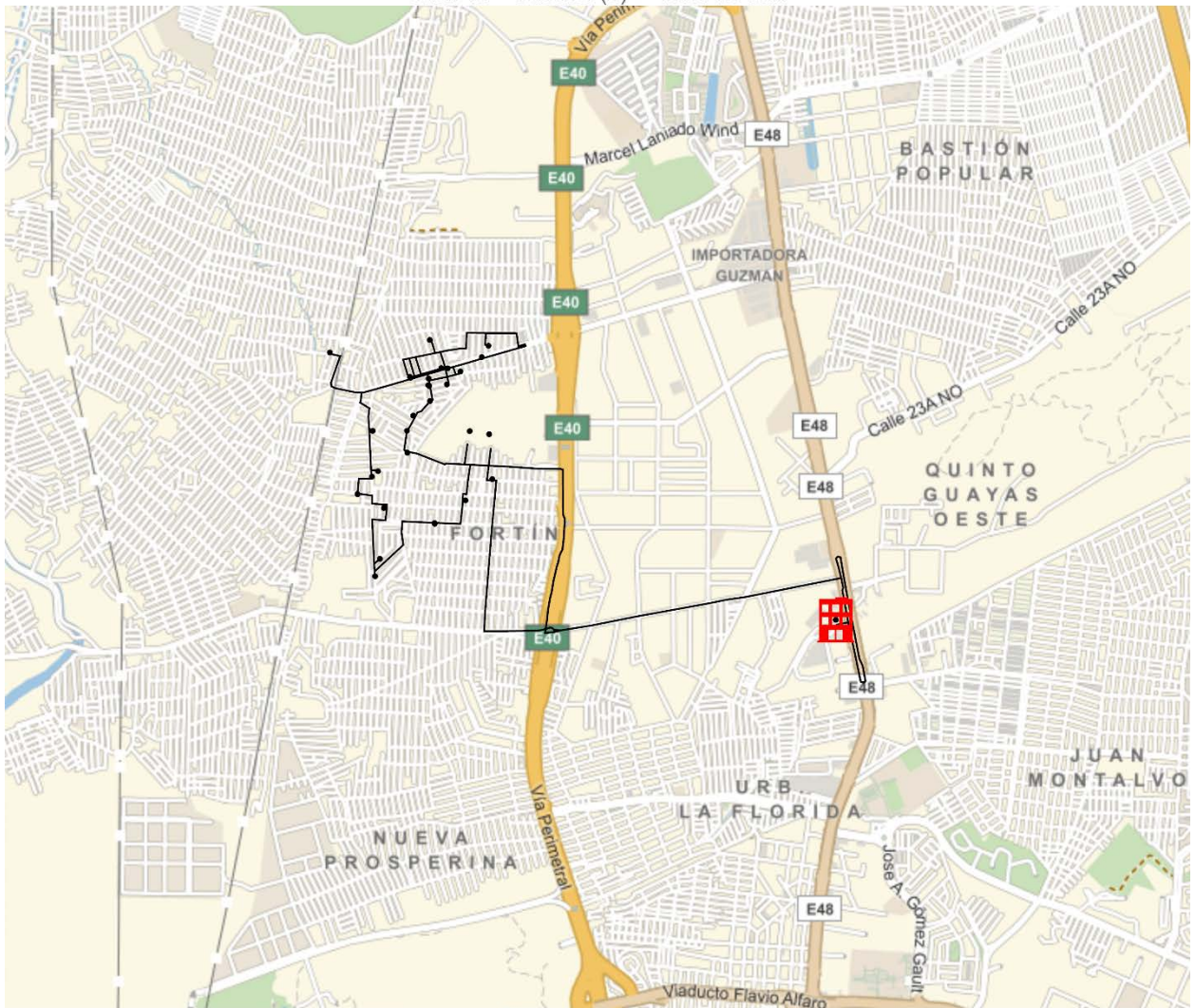


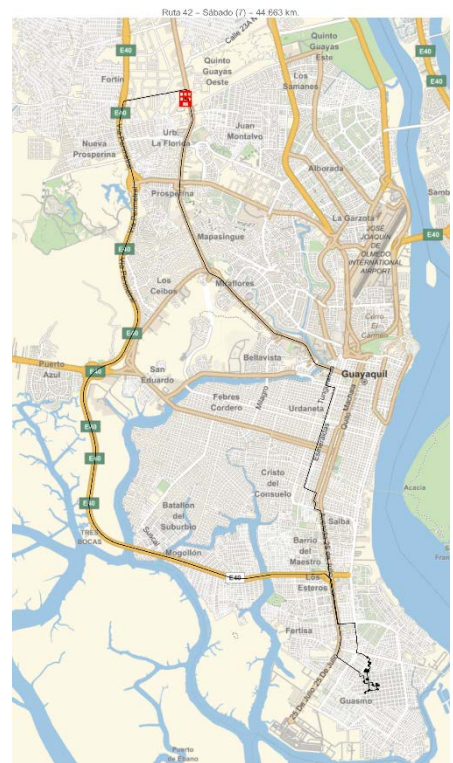
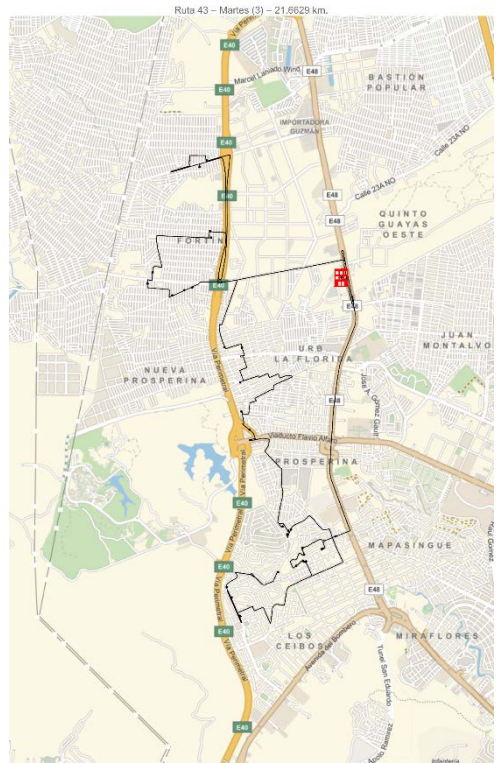
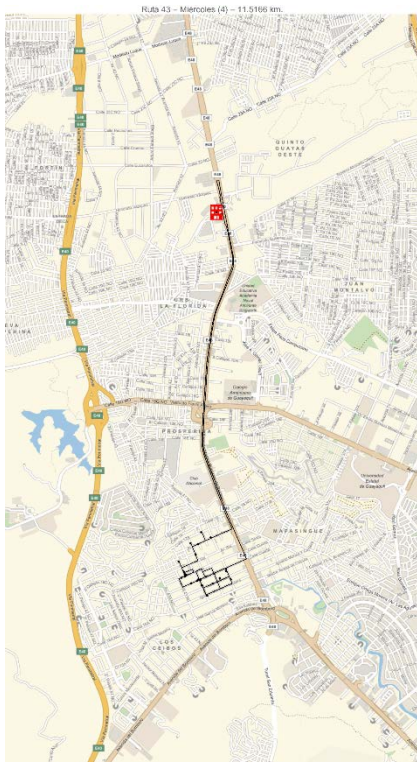
Ruta 44 – Lunes (2) – 145.9 km.

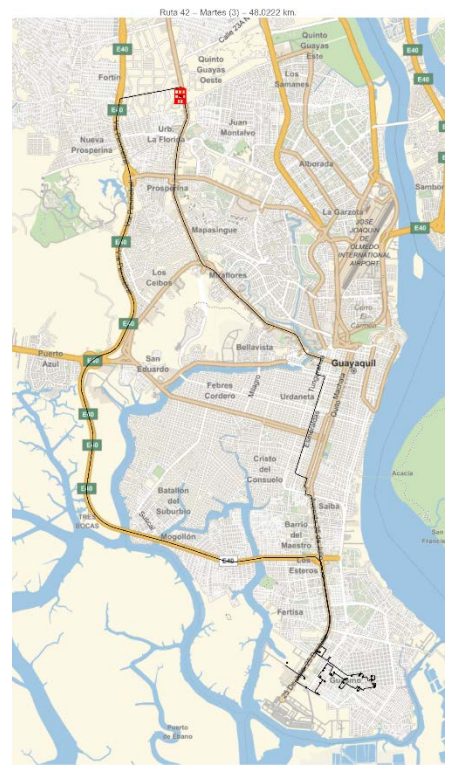
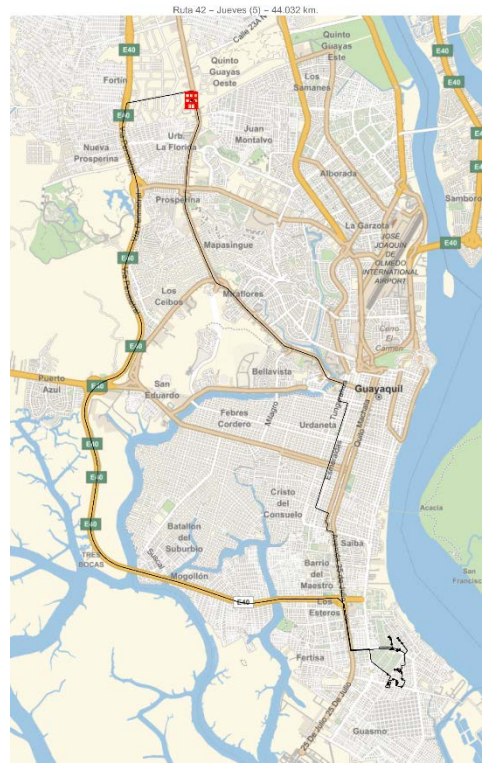
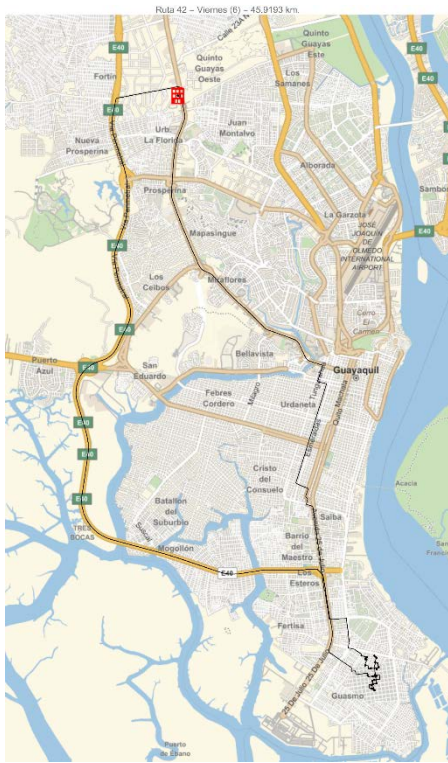


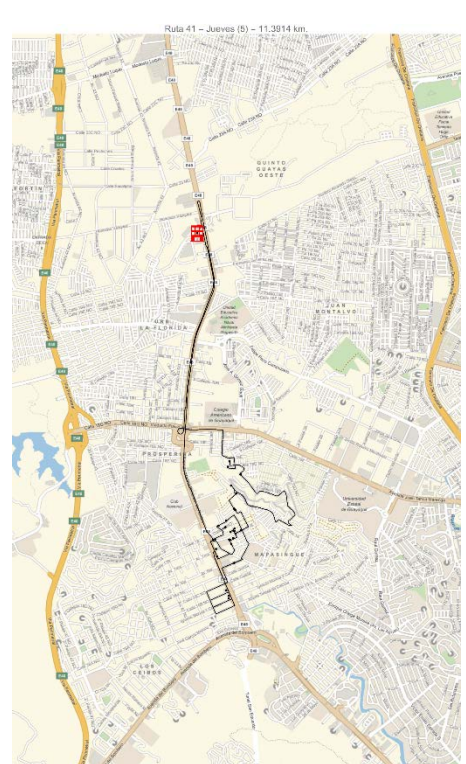
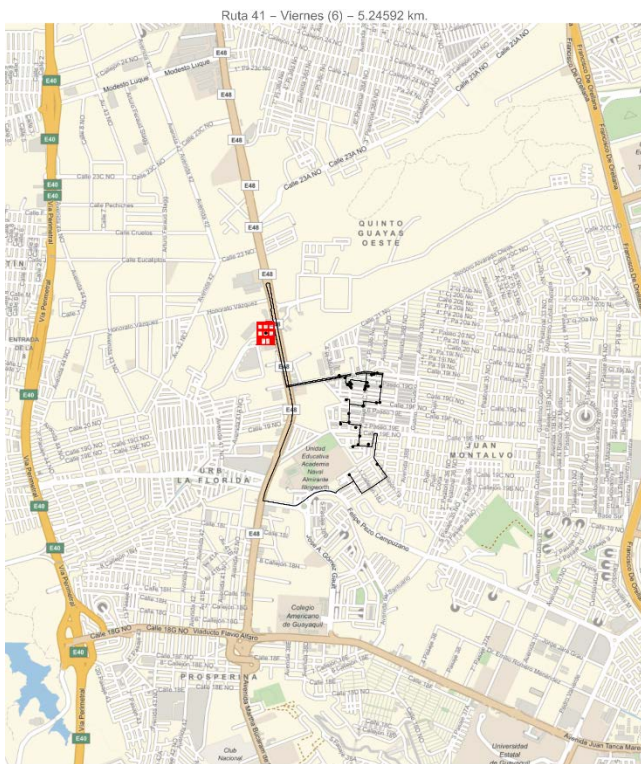
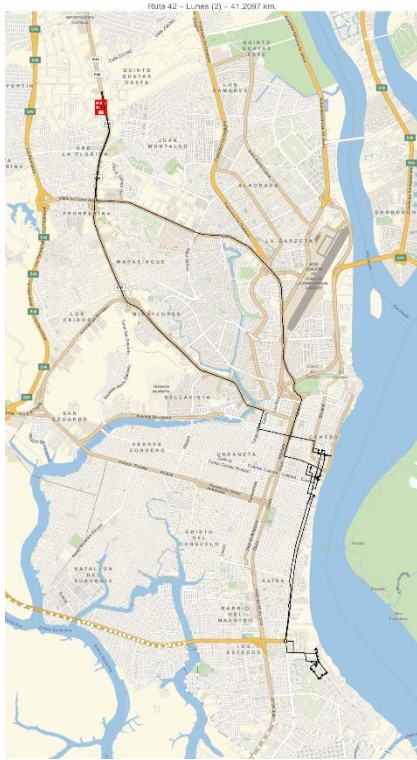


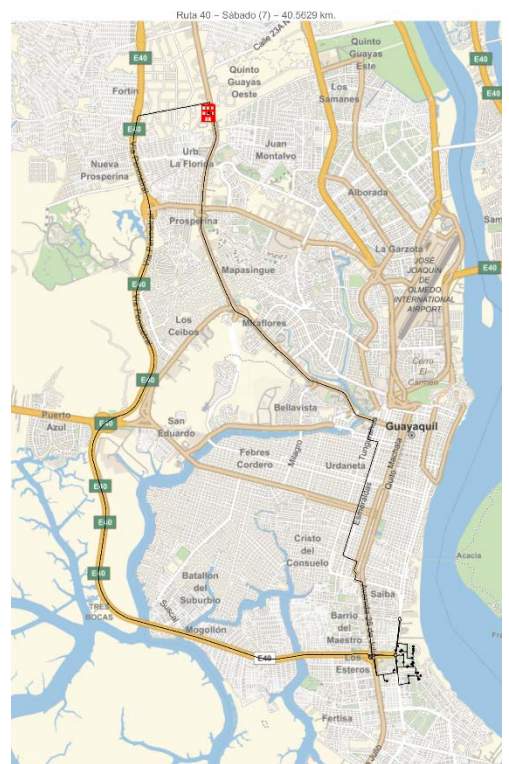
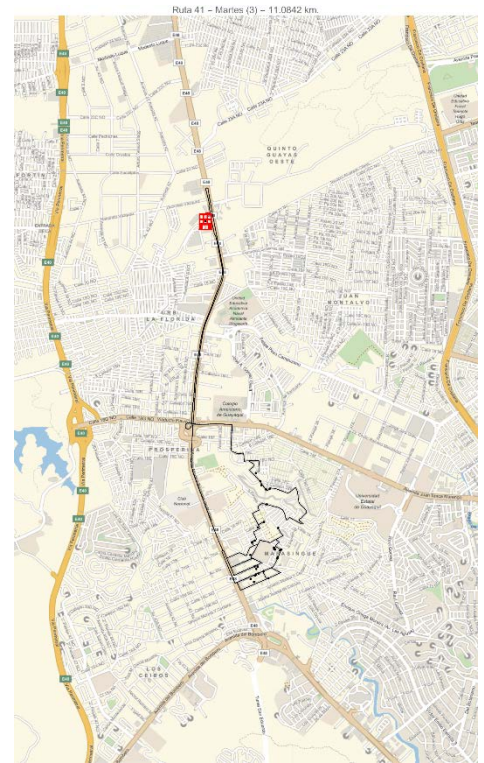
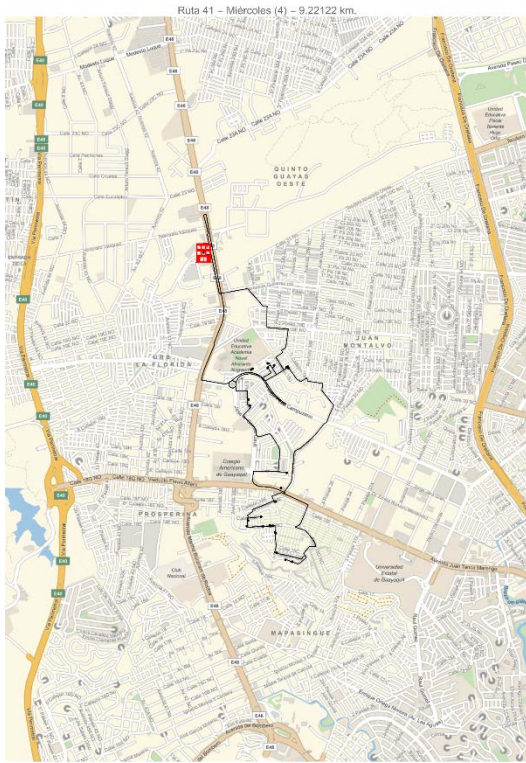
Ruta 43 - Jueves (5) - 13.6756 km.

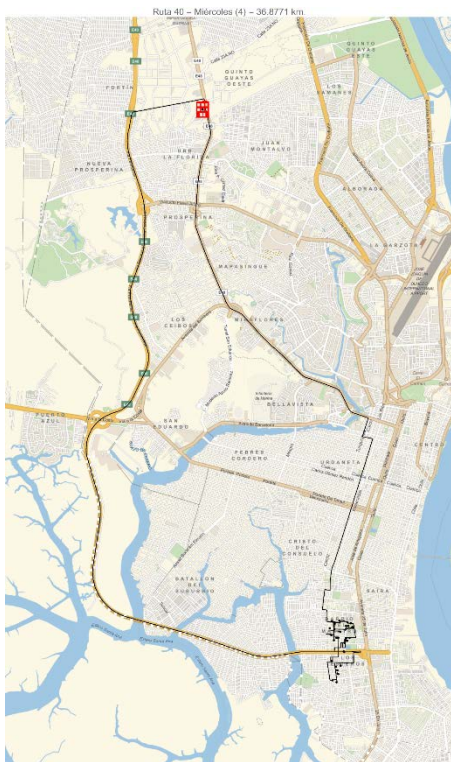
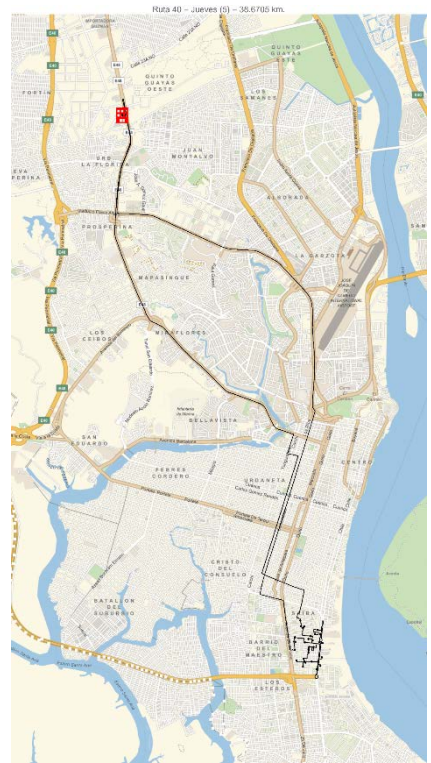


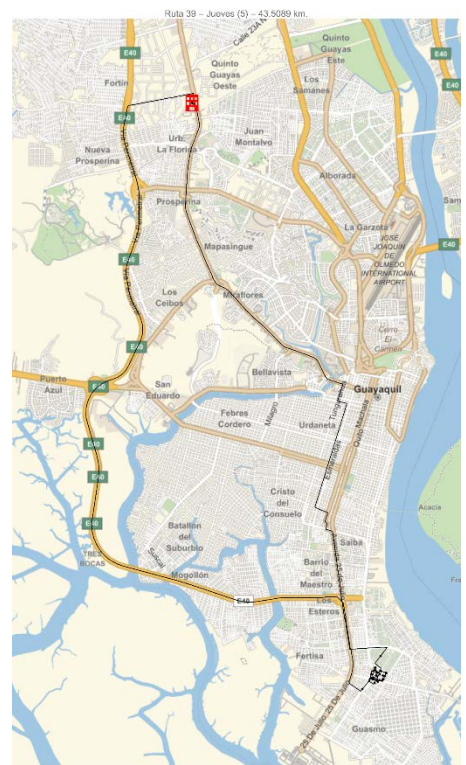
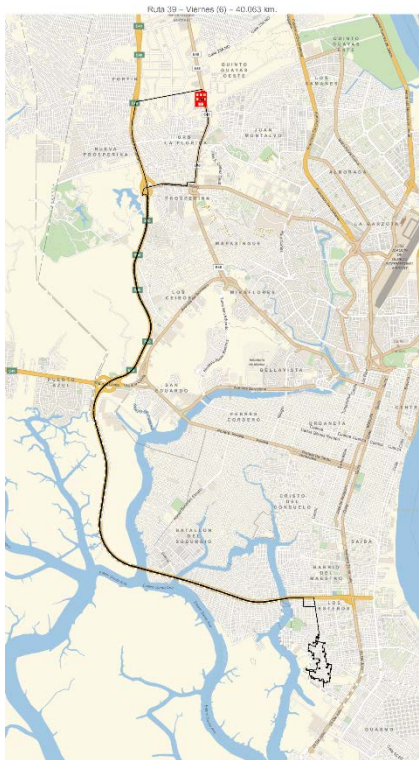
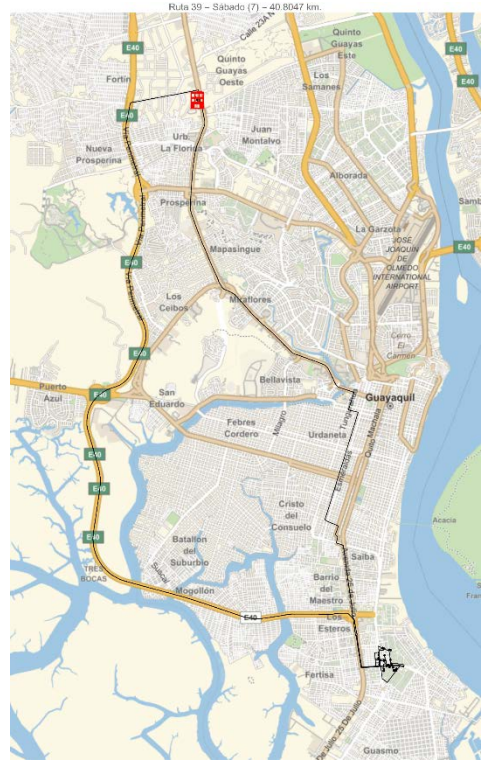




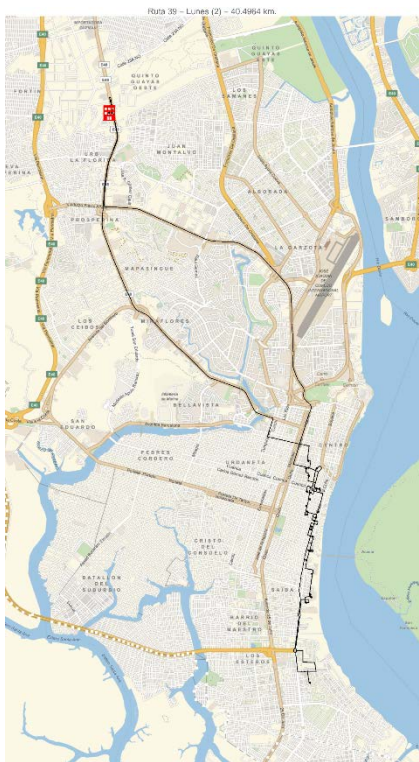
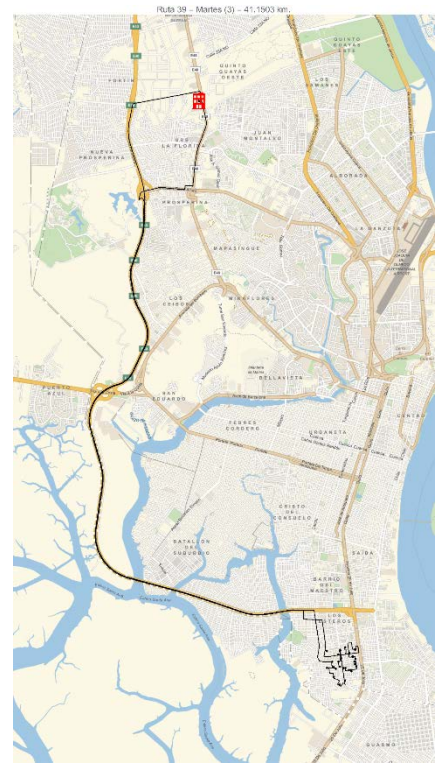
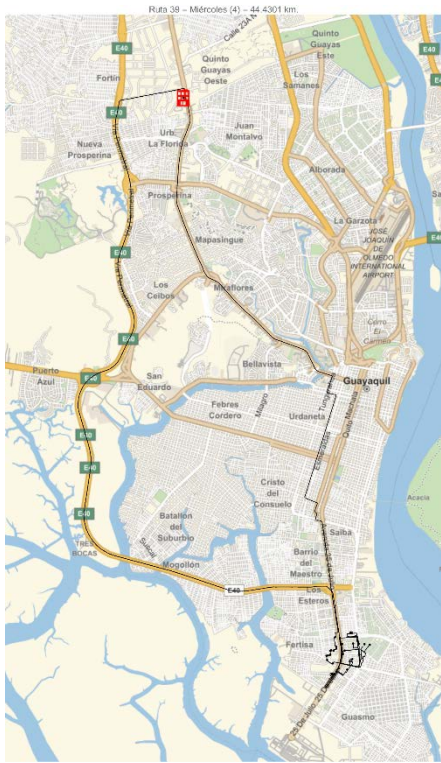




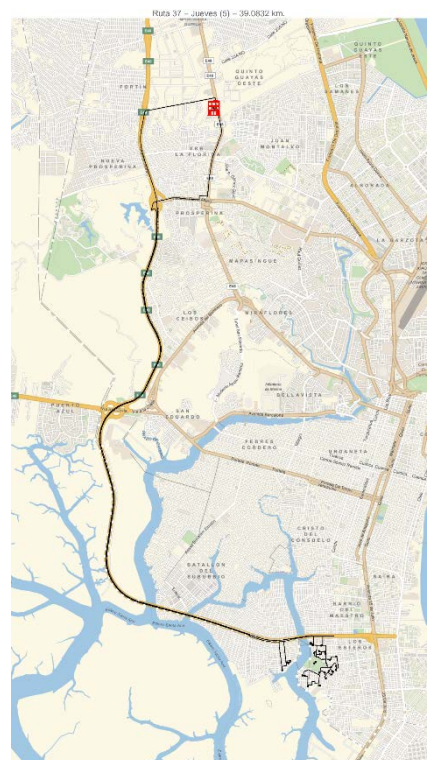
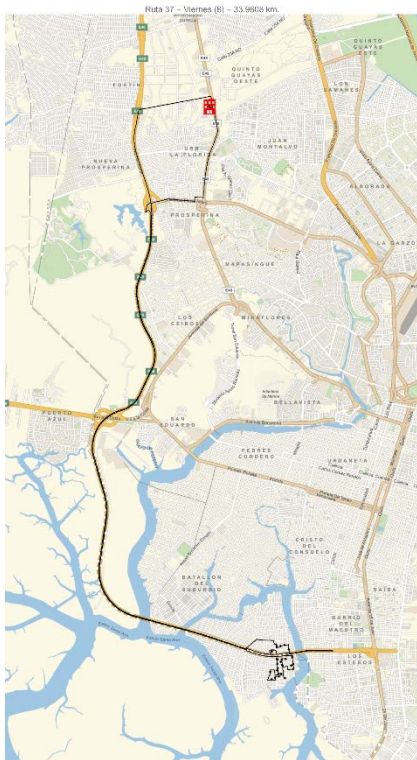
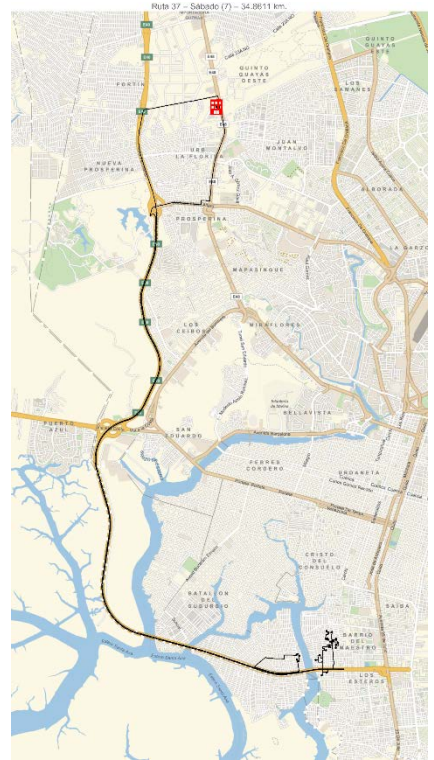


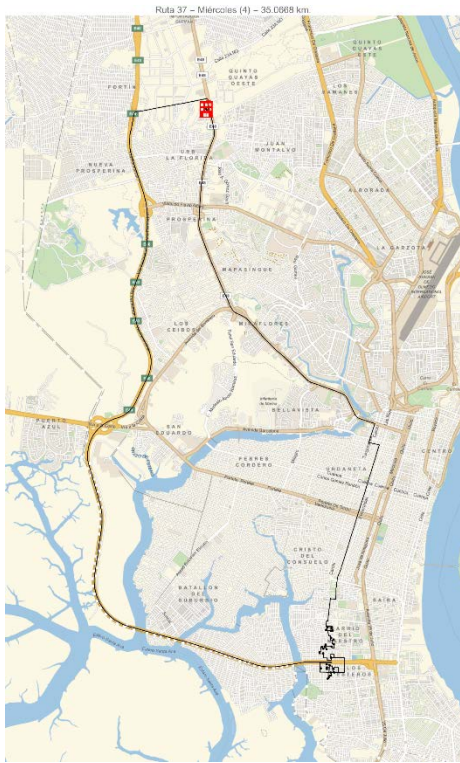






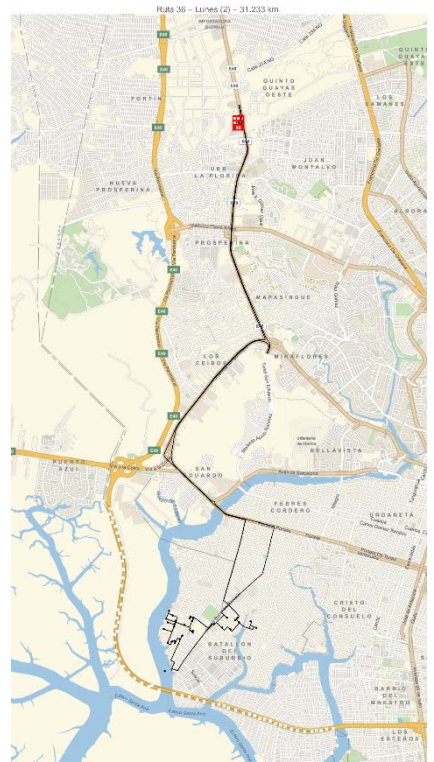
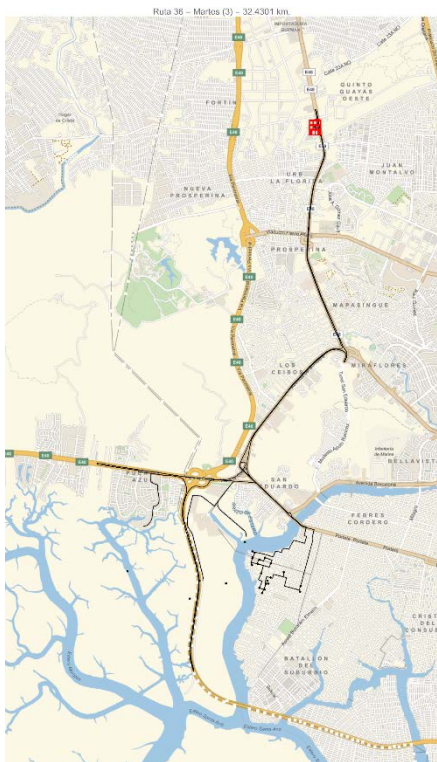
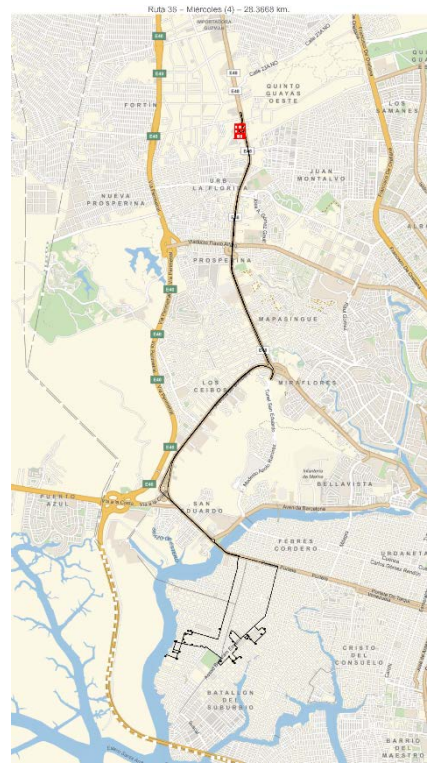
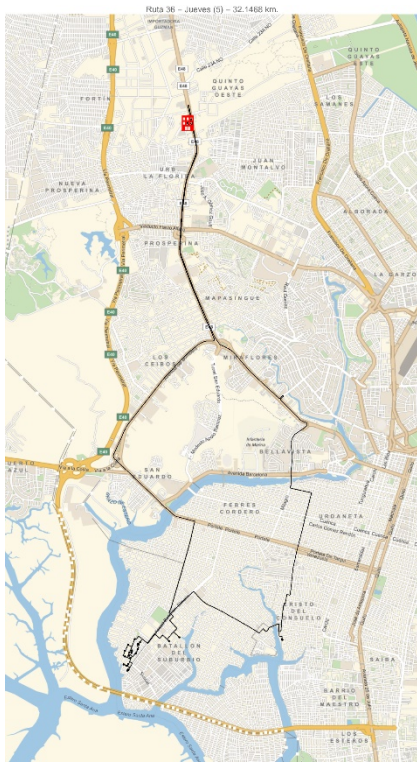


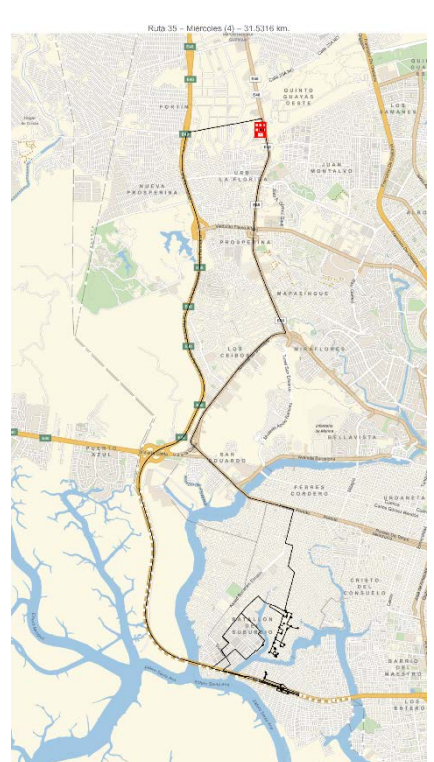
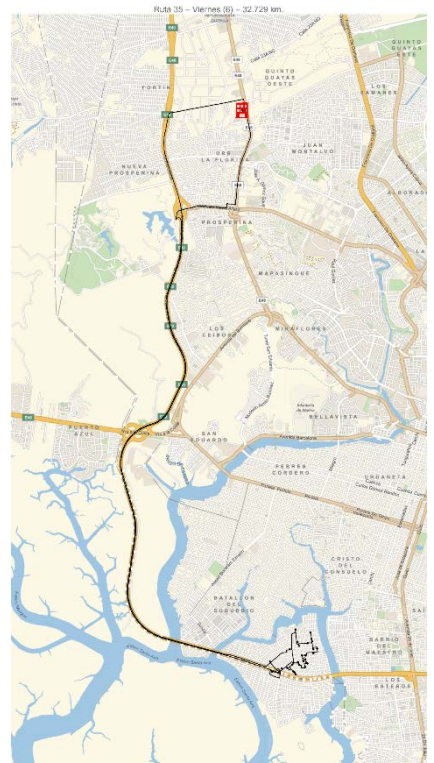
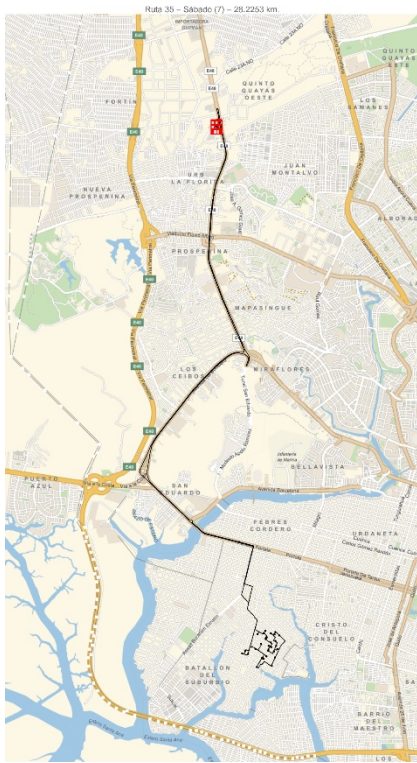


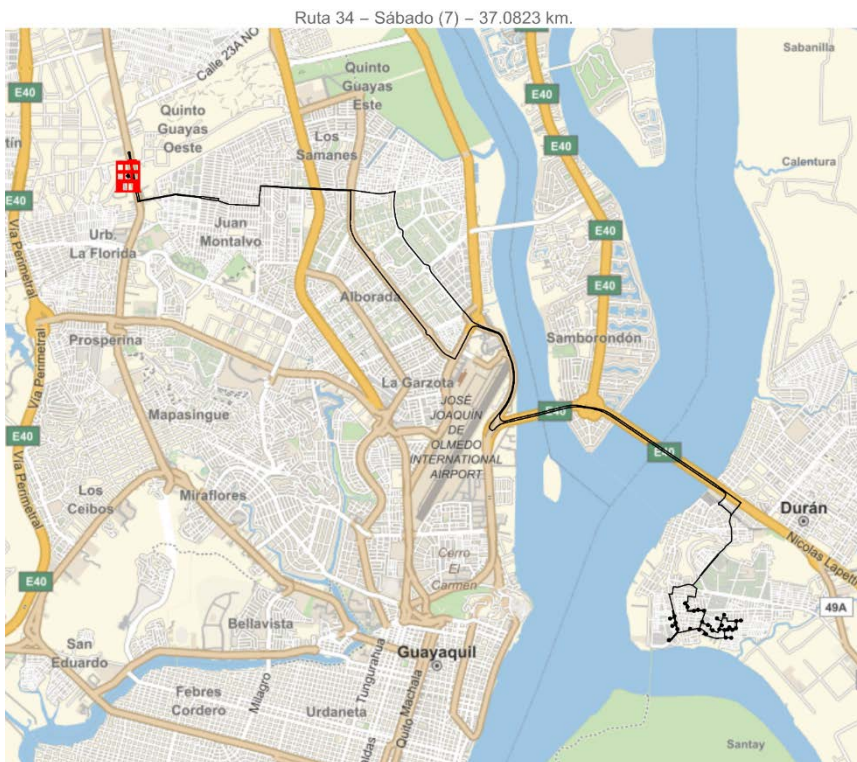


Ruta 36 – Sábado (7) – 80.8493 km.



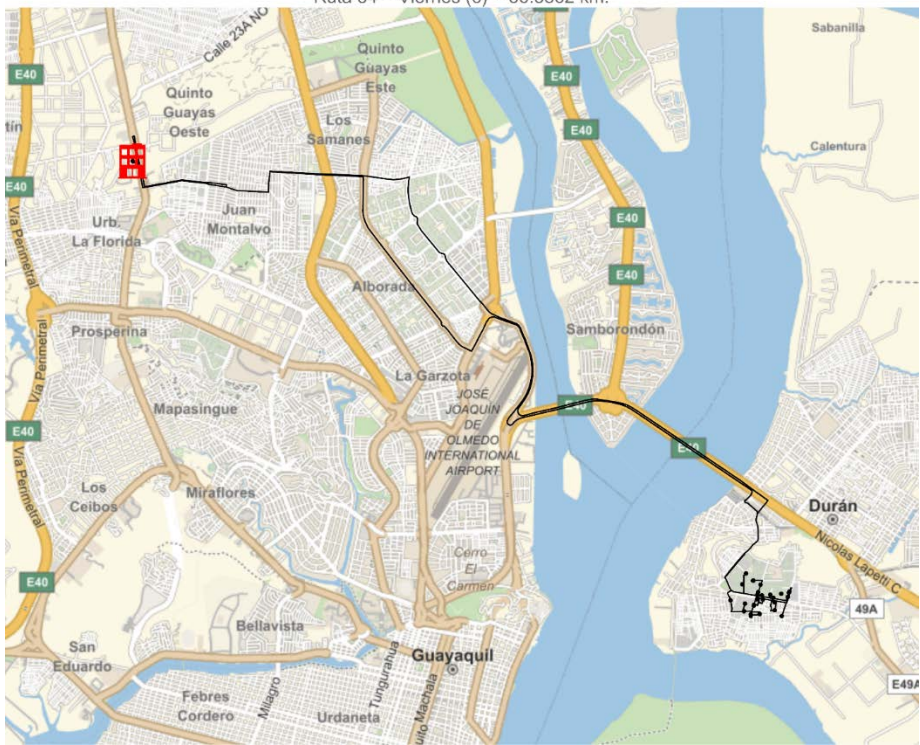








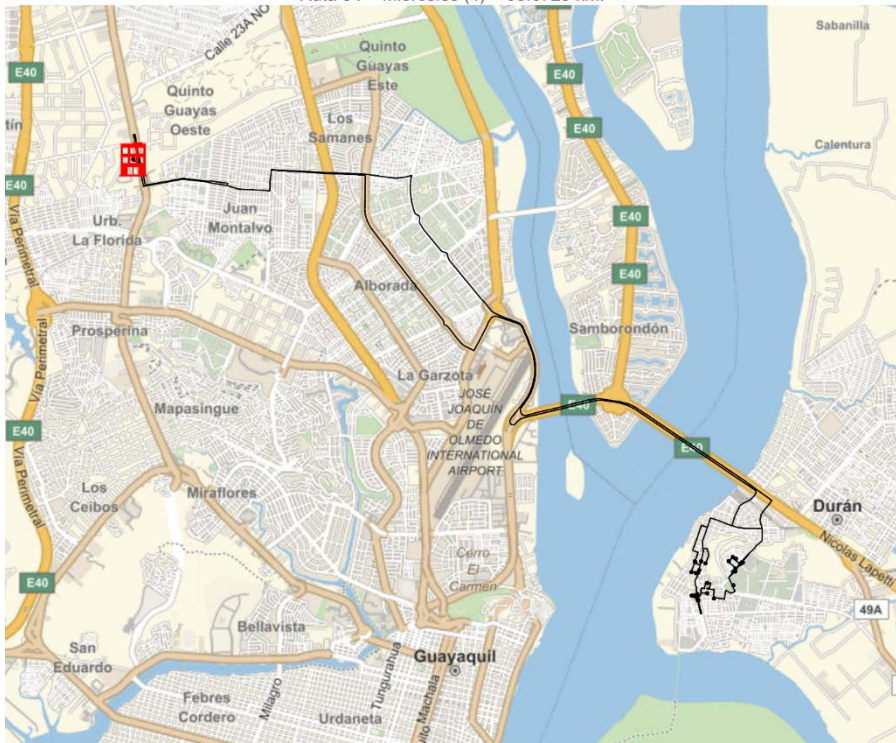
Ruta 34 – Viernes (6) – 36.3862 km.



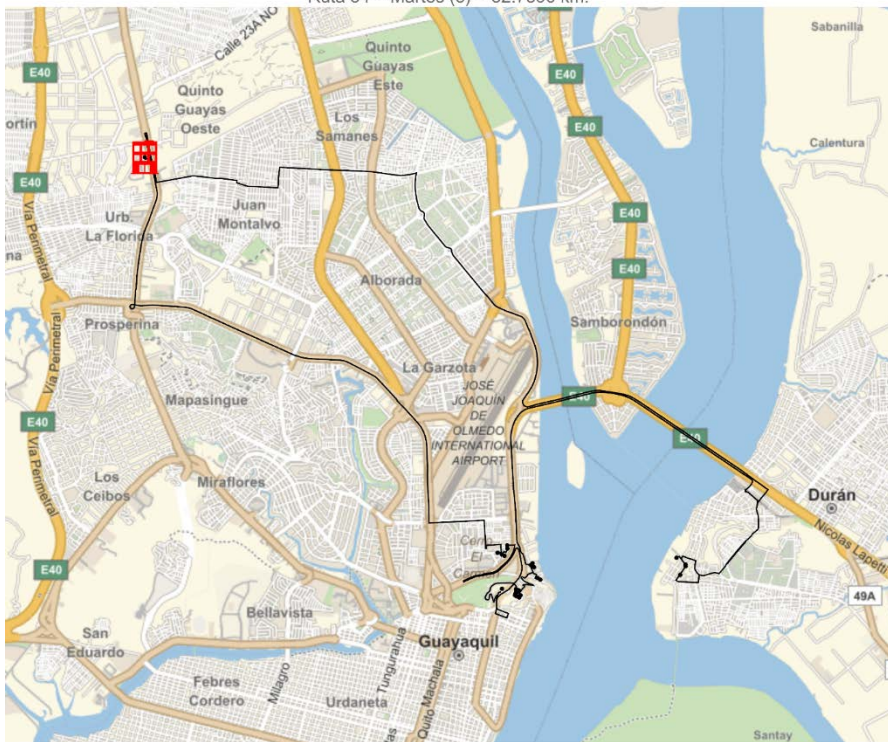
Ruta 34 – Jueves (5) – 35.6386 km.



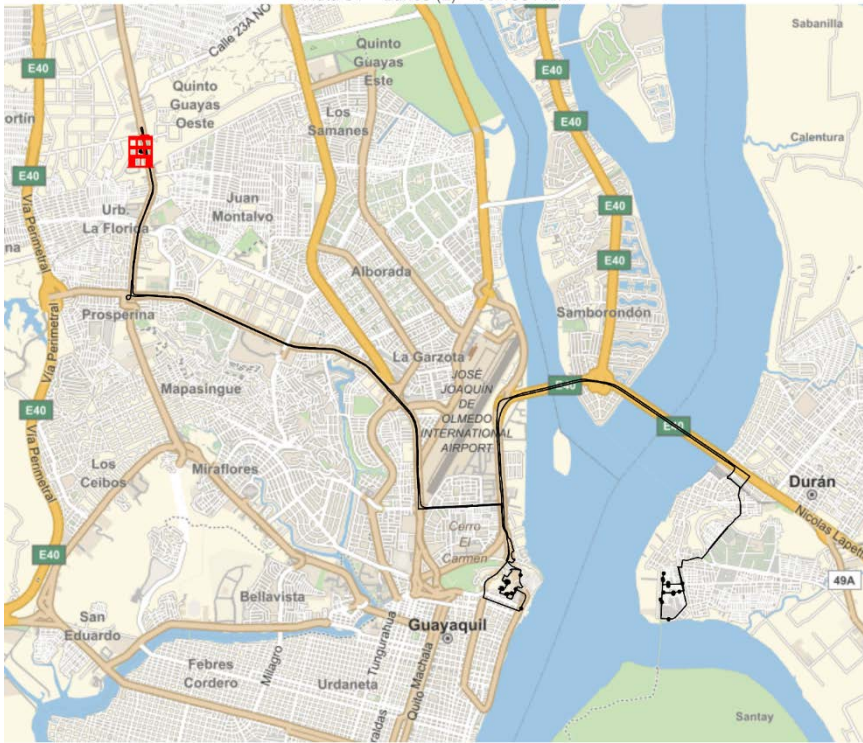
Ruta 34 – Miércoles (4) – 33.6729 km.



Ruta 34 – Martes (3) – 32.7896 km.



Ruta 34 – Lunes (2) – 35.1031 km.



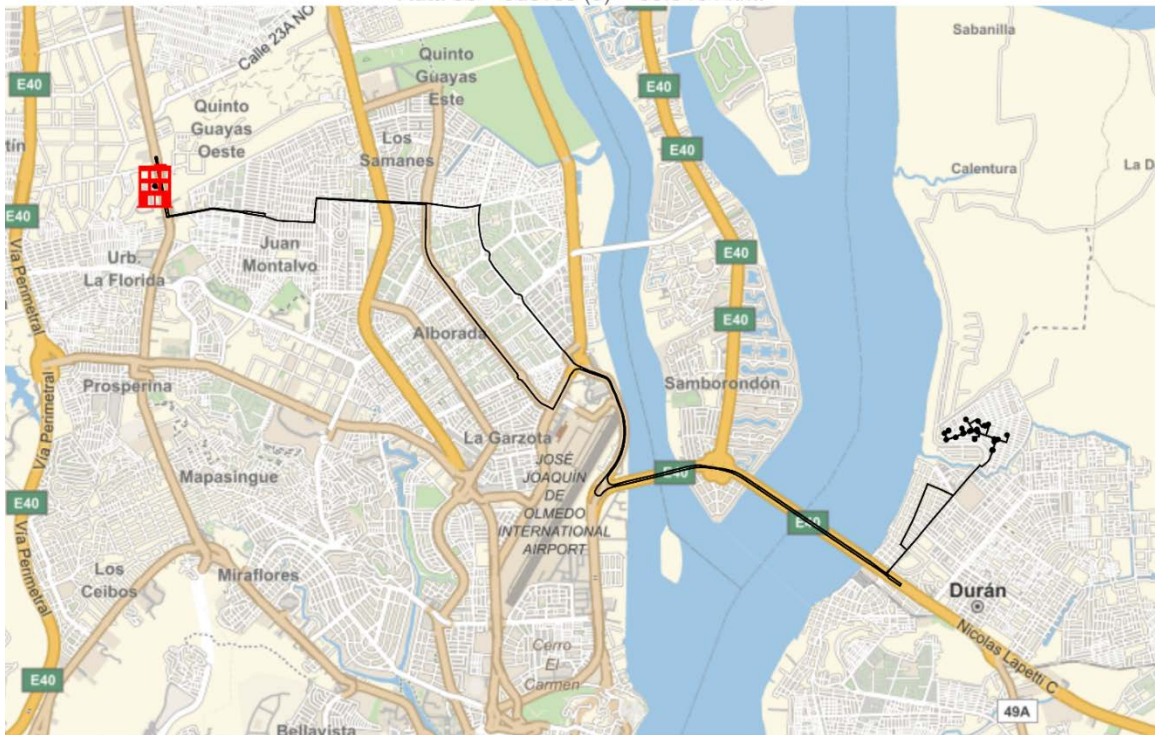
Ruta 33 – Sábado (7) – 30.7545 km.



Ruta 33 – Viernes (6) – 34.6691 km.



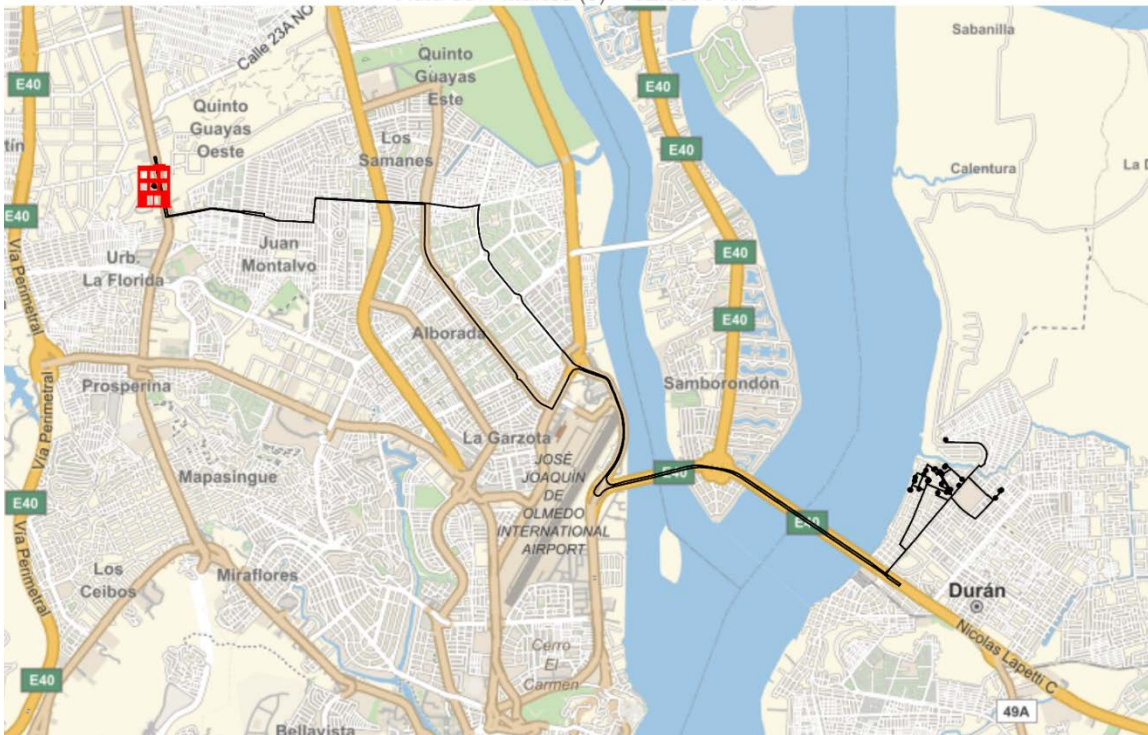
Ruta 33 – Jueves (5) – 30.5484 km.



Ruta 33 – Miércoles (4) – 34.345 km.



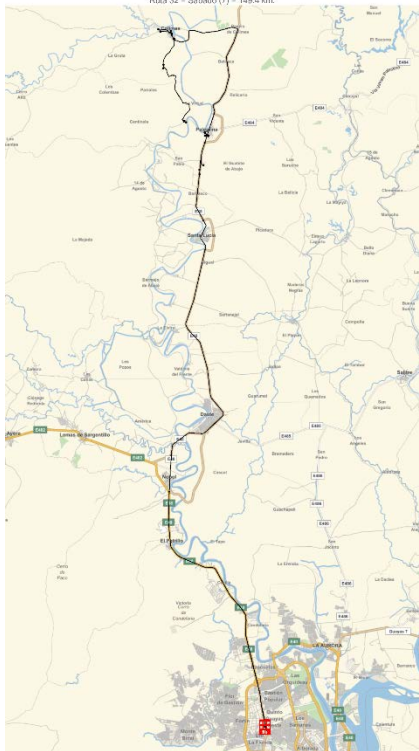
Ruta 33 – Martes (3) – 32.9373 km.



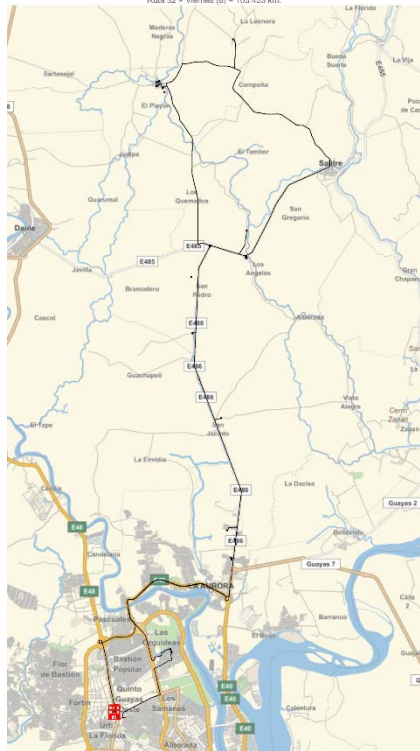
Ruta 33 – Lunes (2) – 24.523 km.

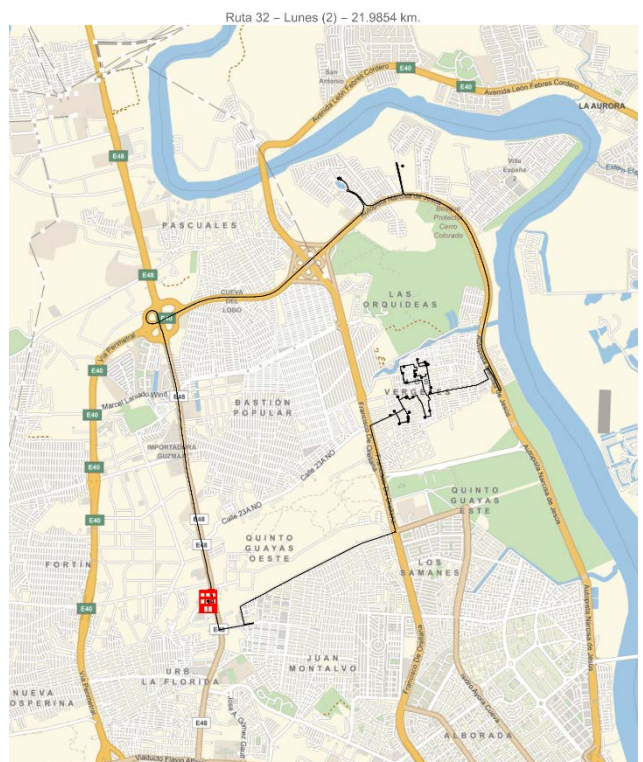
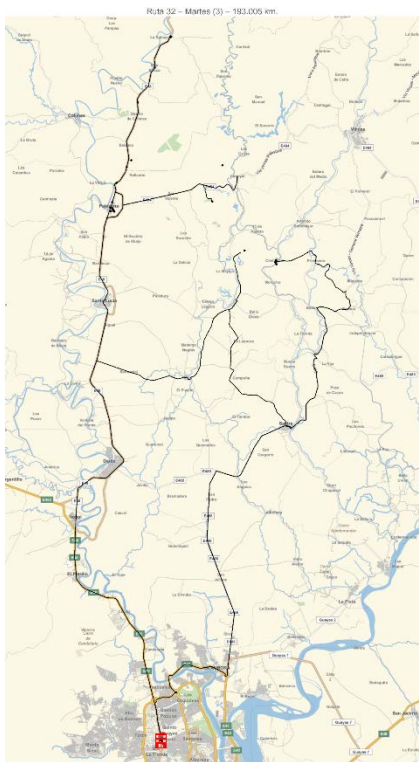
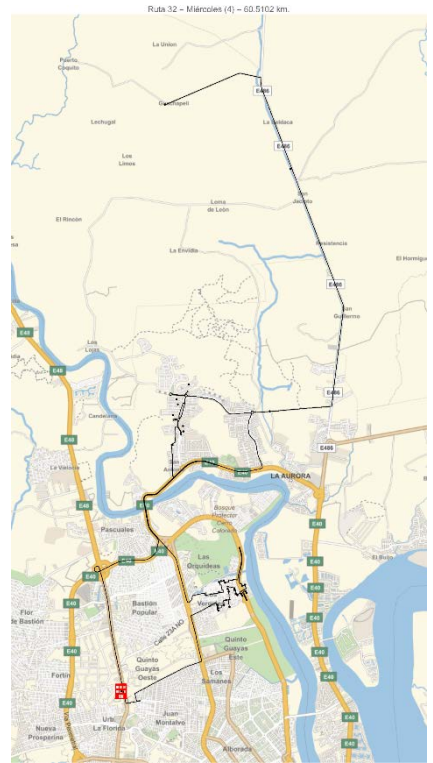
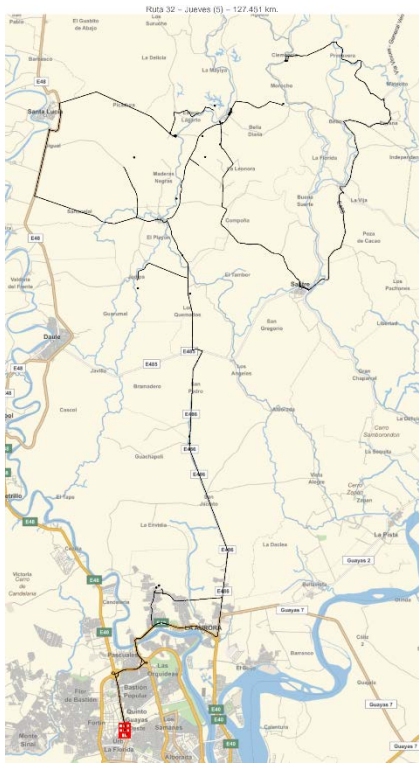


Ruta 32 – Sábado (7) – 149.4 km.

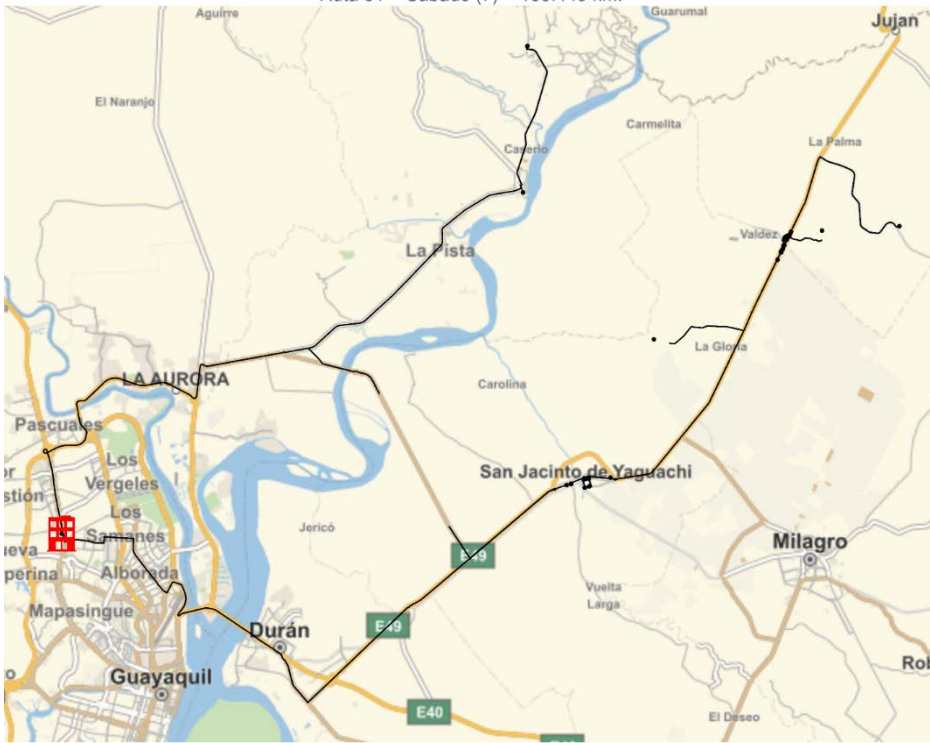


Ruta 32 – Viernes (8) – 103.455 km.

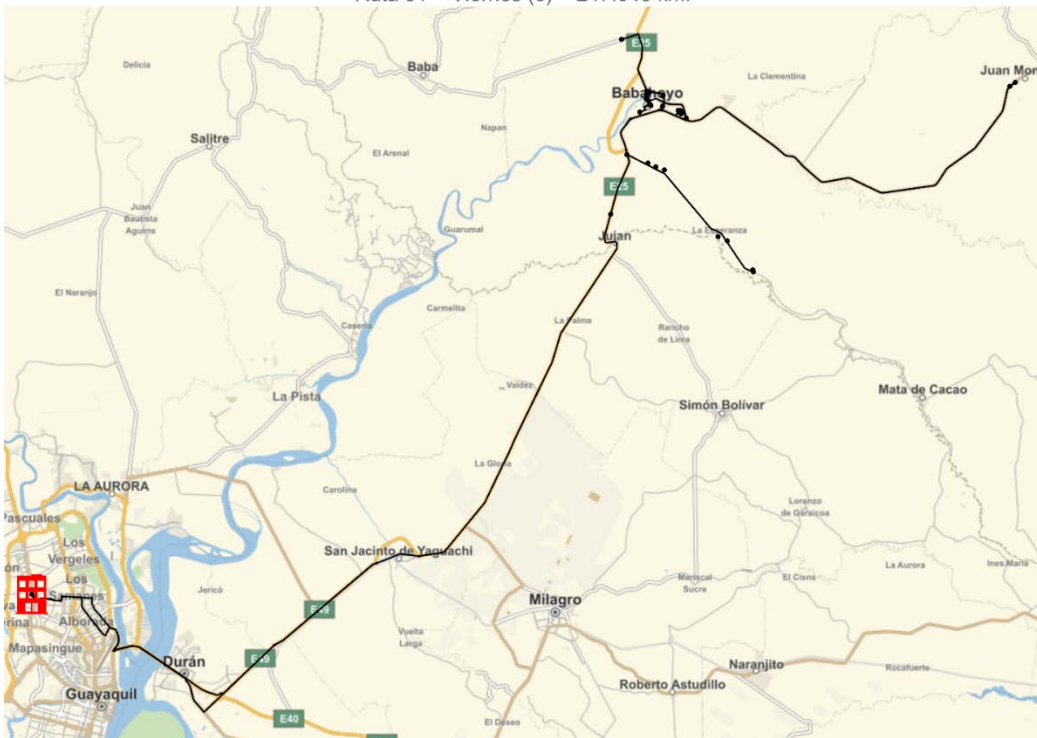




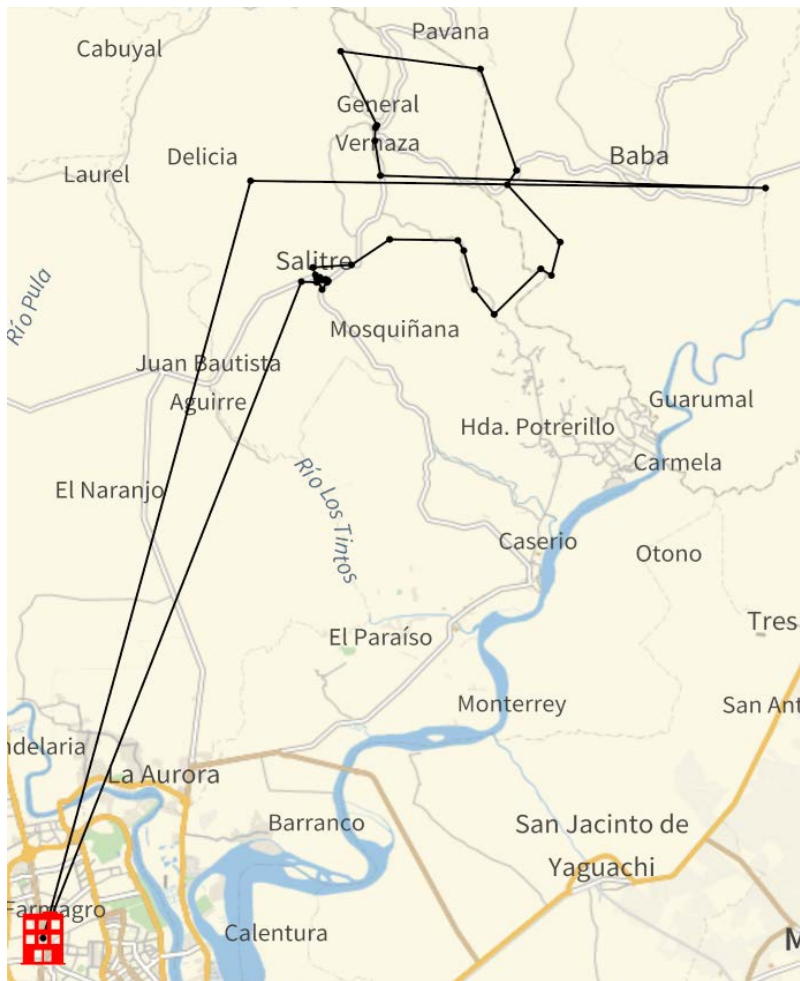
Ruta 31 – Sábado (7) – 135.449 km.



Ruta 31 – Viernes (6) – 247.046 km.



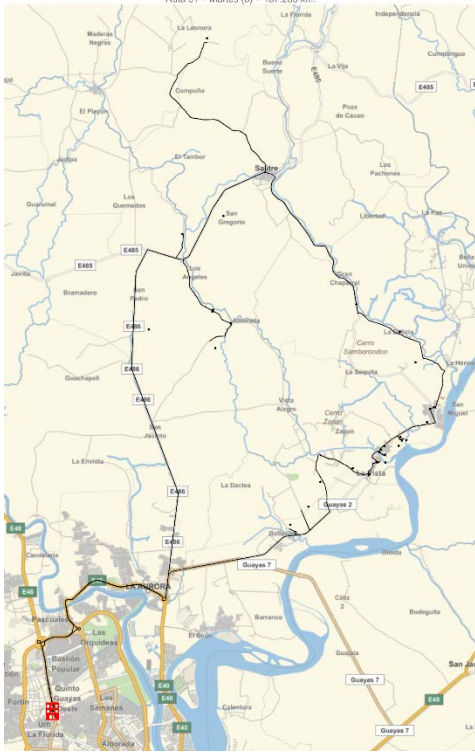




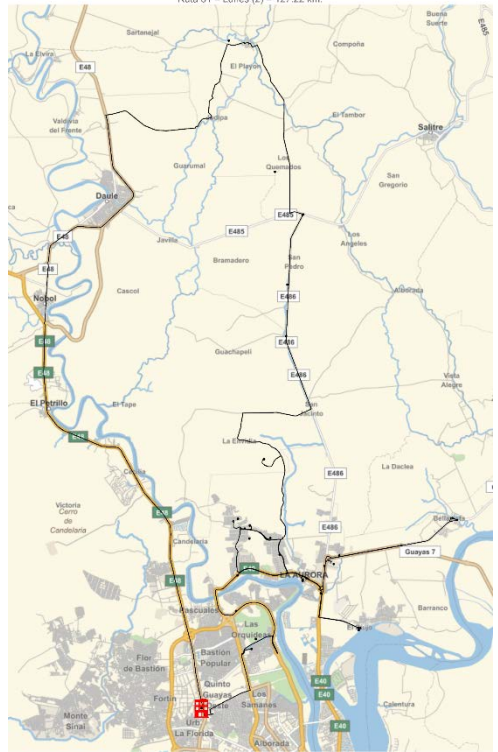
Ruta 31 – Miércoles (4) – 212.857 km.



Ruta 31 – Martes (3) – 137.265 km.



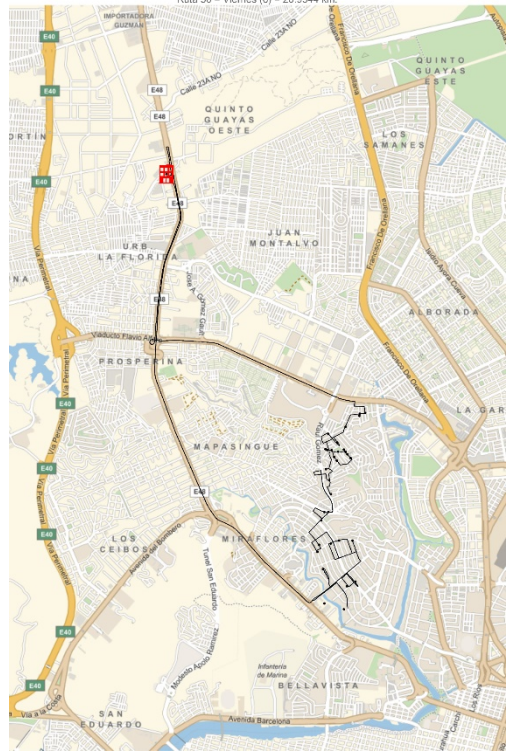
Ruta 31 – Lunes (2) – 127.22 km.

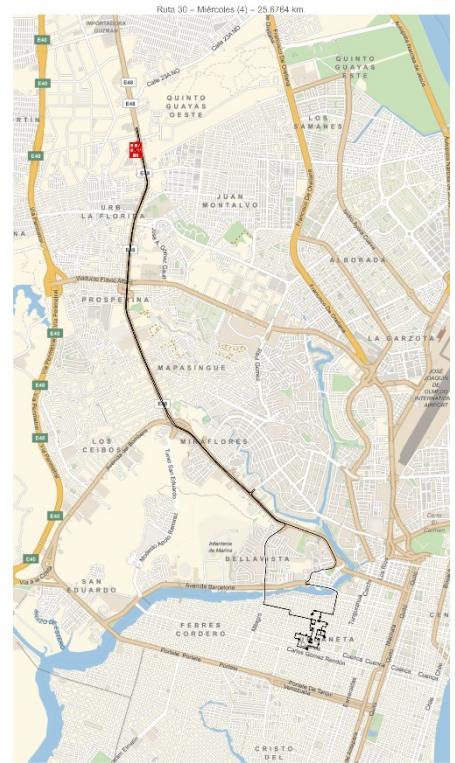


Ruta 30 – Sábado (7) – 24.7321 km.



Ruta 30 – Viernes (6) – 20.9344 km.





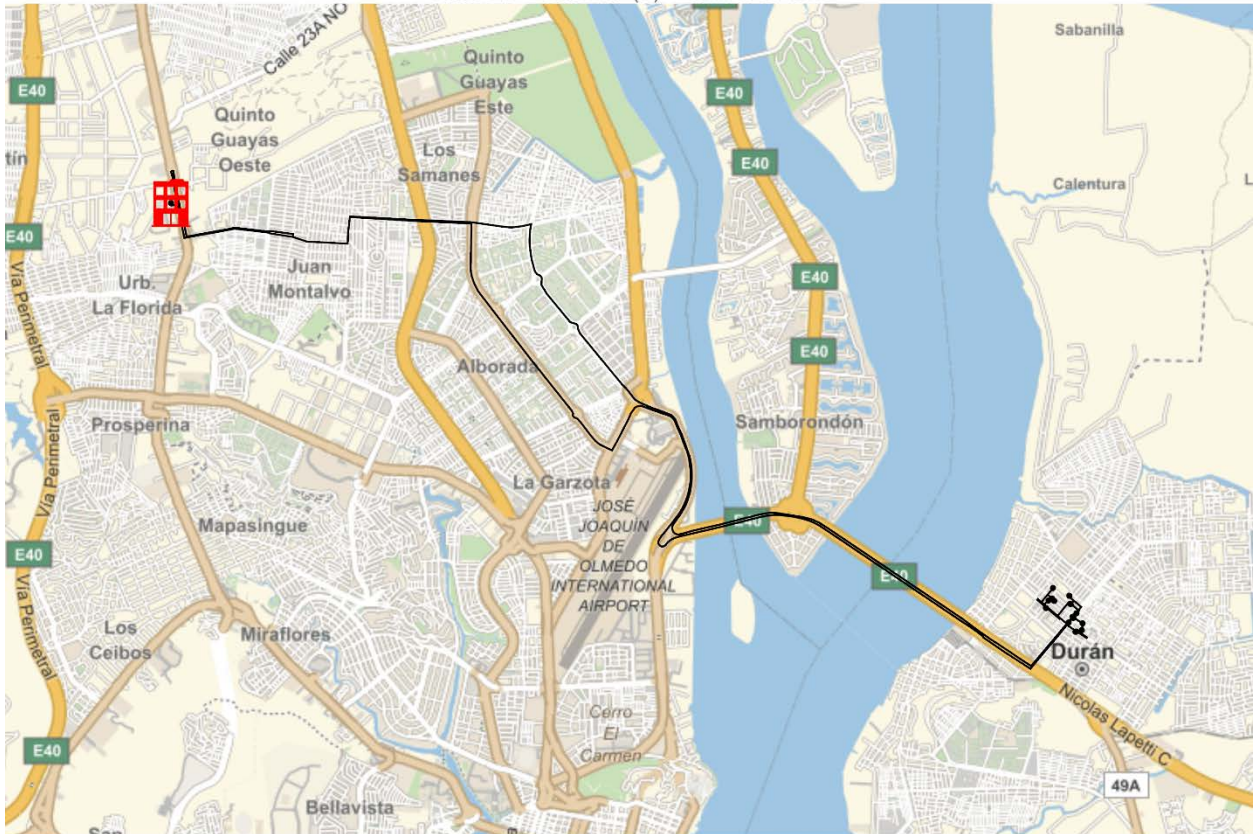
Ruta 29 – Sábado (7) – 35.9605 km.



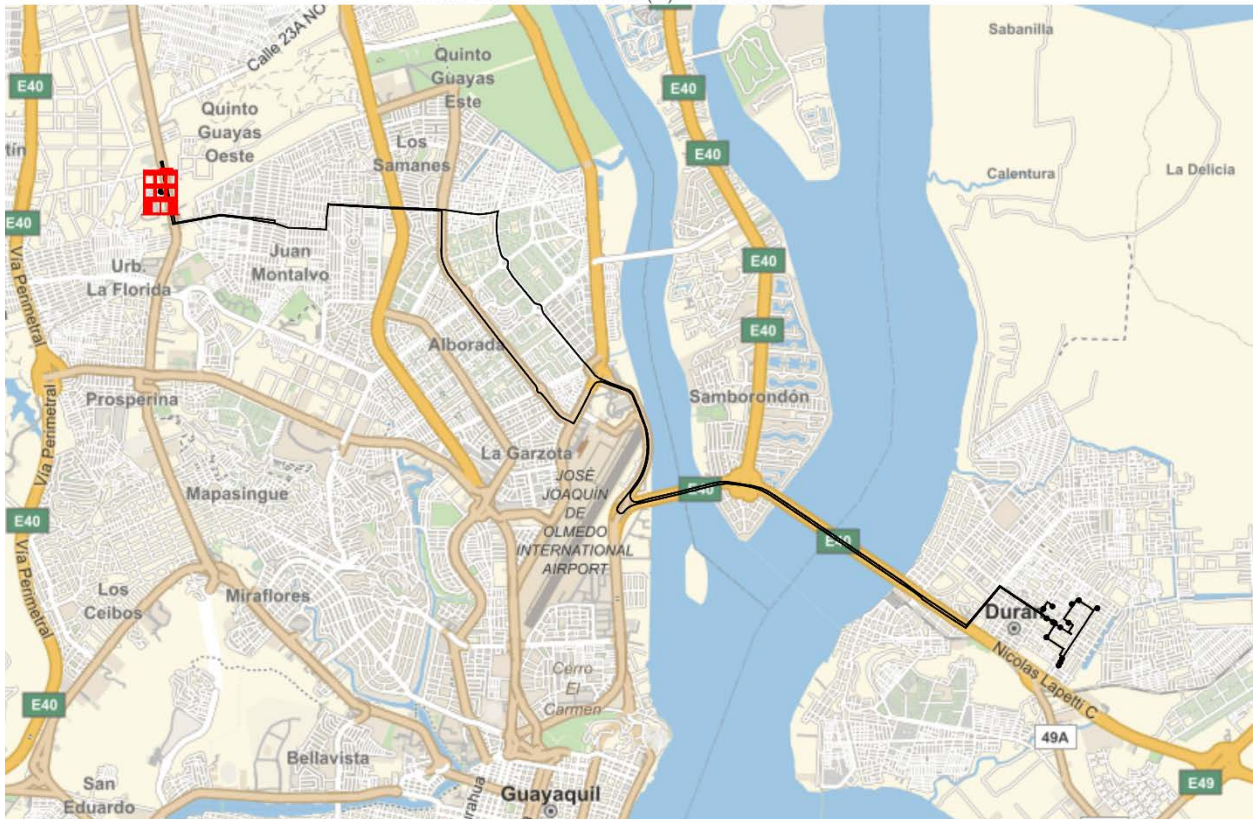
Ruta 29 – Viernes (6) – 37.7473 km.



Ruta 29 – Jueves (5) – 32.7043 km.



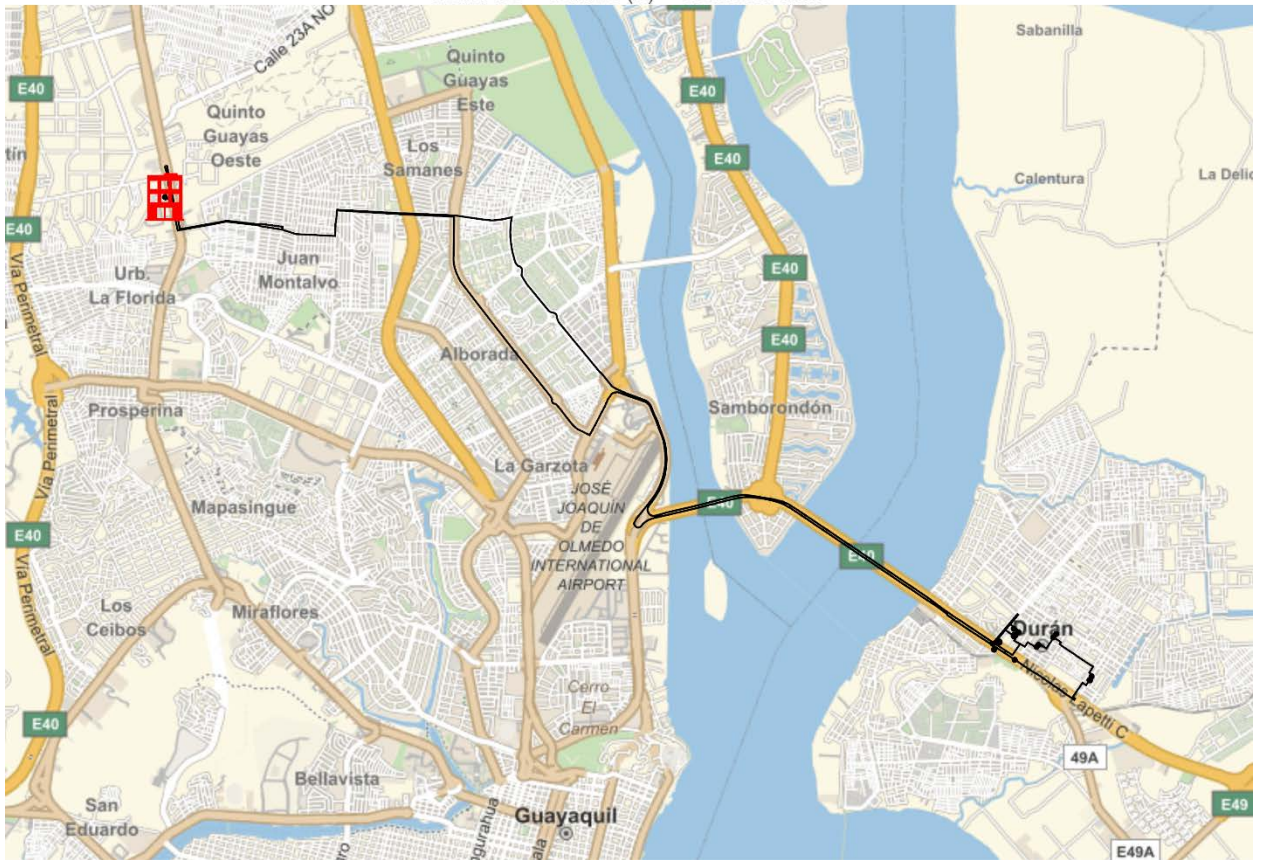
Ruta 29 – Miércoles (4) – 37.0387 km.



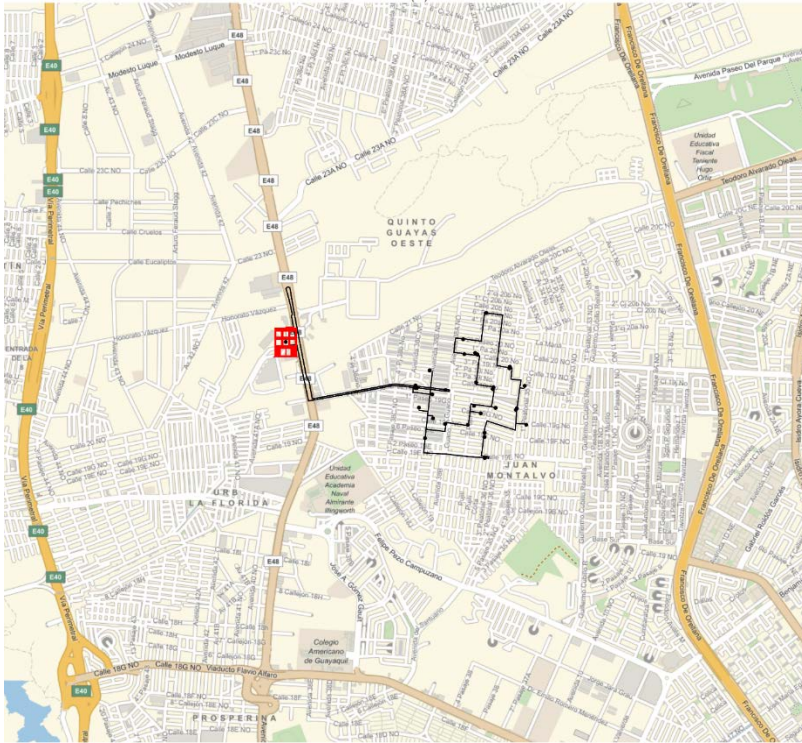
Ruta 29 – Martes (3) – 39.1735 km.



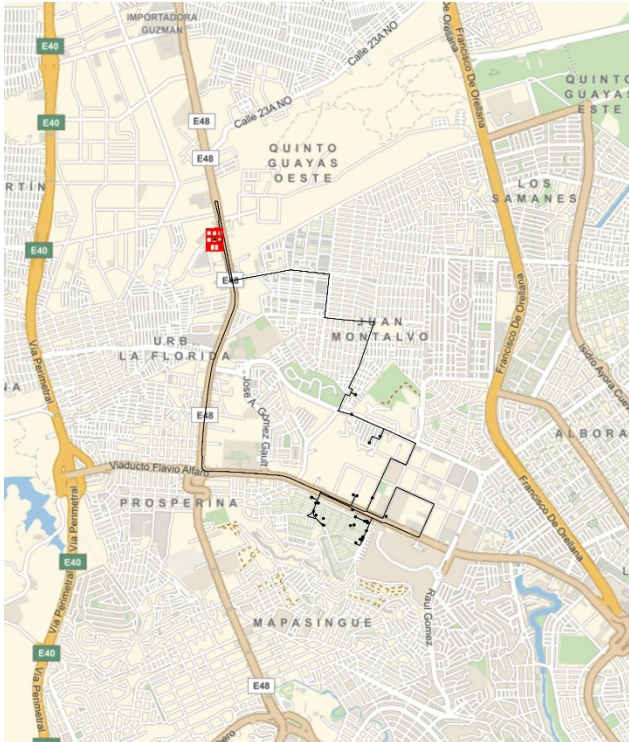
Ruta 29 – Lunes (2) – 35.8936 km.



Ruta 28 – Sábado (7) – 7.79468 km.



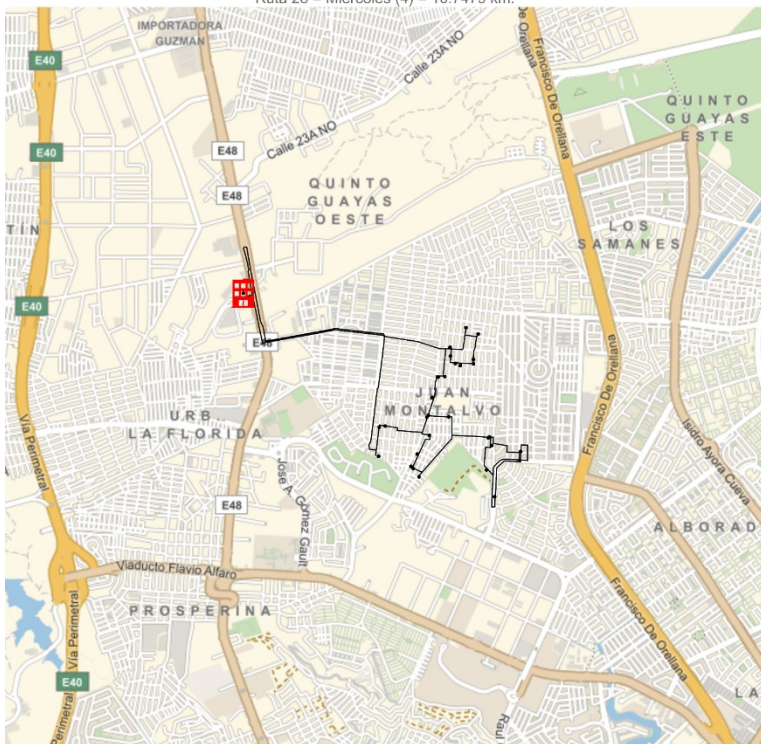
Ruta 28 – Viernes (6) – 11.7701 km.



Ruta 28 – Jueves (5) – 8.18281 km.

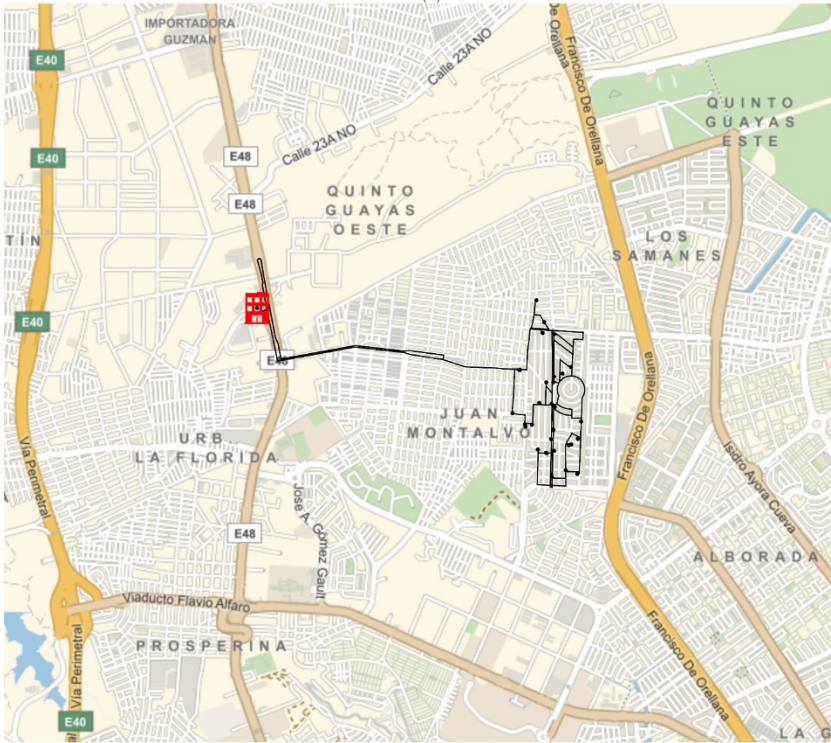


Ruta 28 – Miércoles (4) – 10.7479 km.

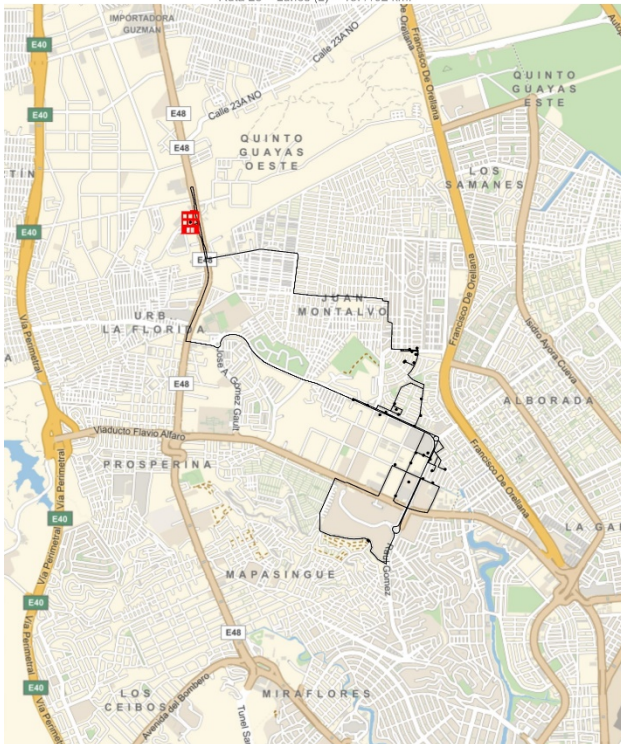




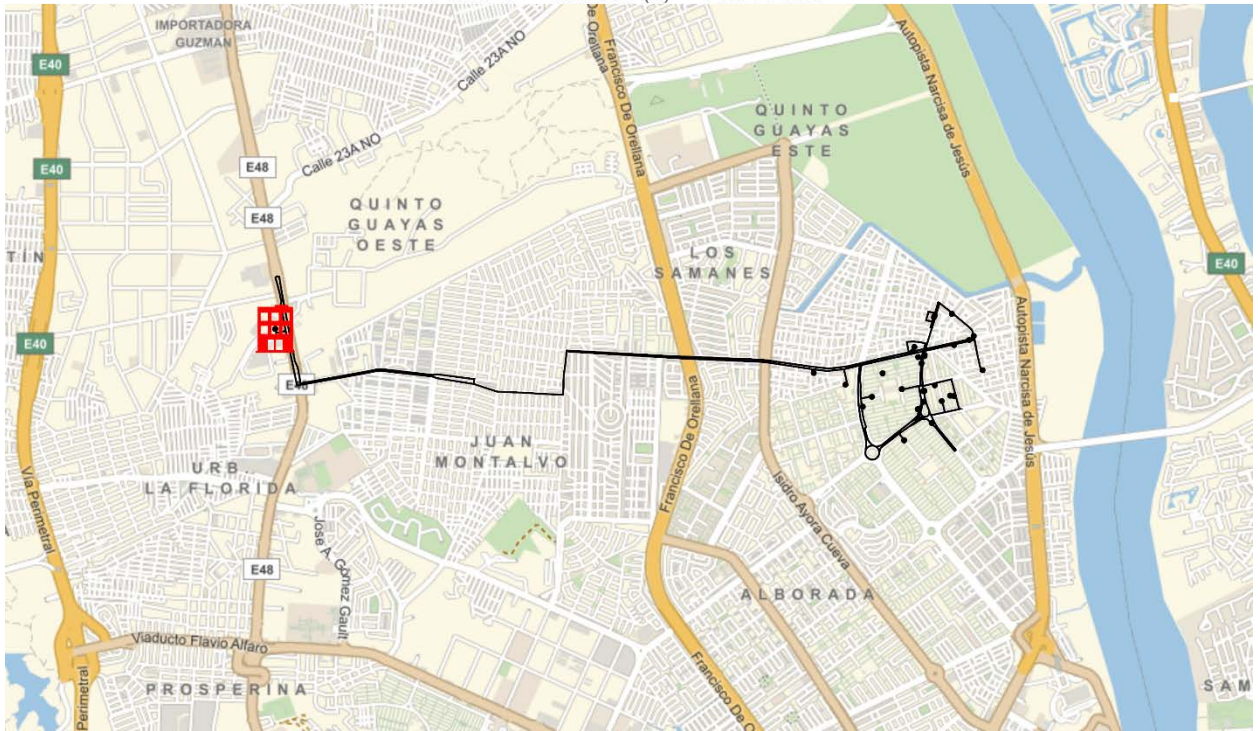
Ruta 28 - Martes (3) - 10.1452 km.



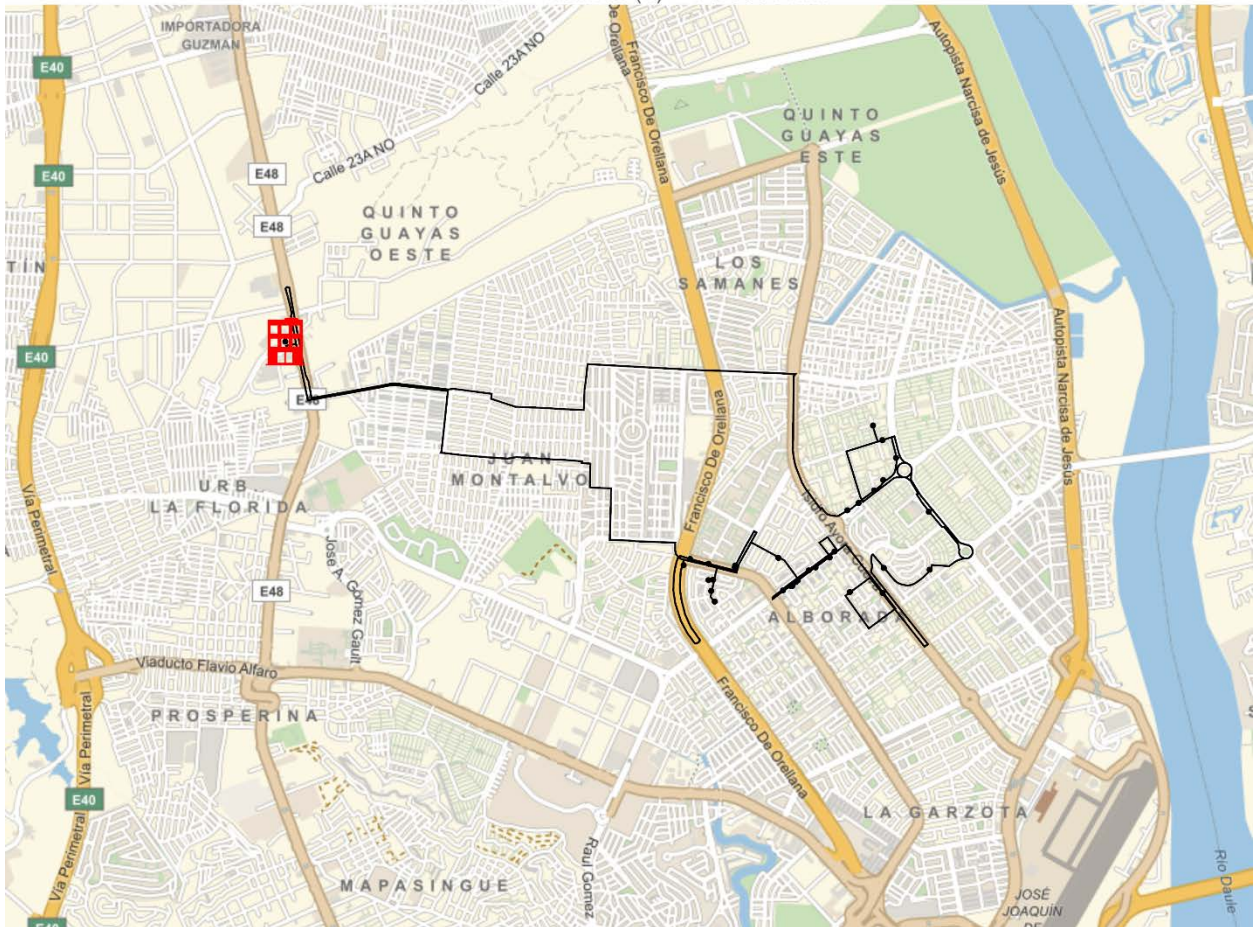
Ruta 28 - Lunes (2) - 15.4102 km.



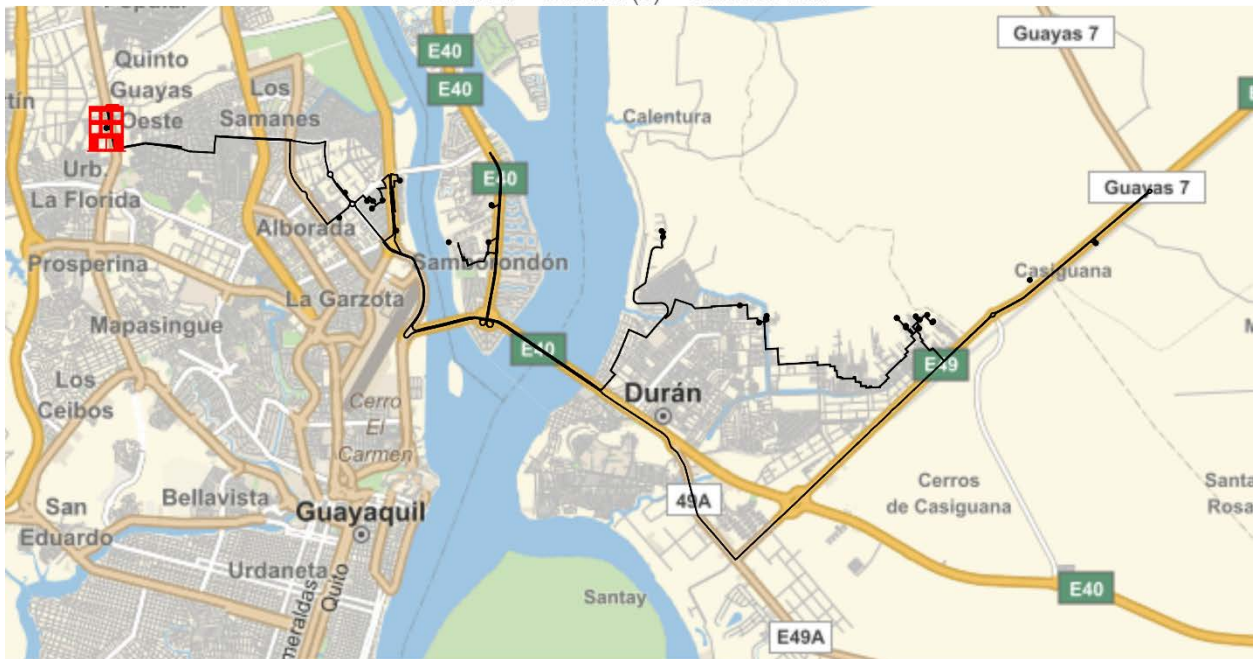
Ruta 27 – Sábado (7) – 15.139 km.



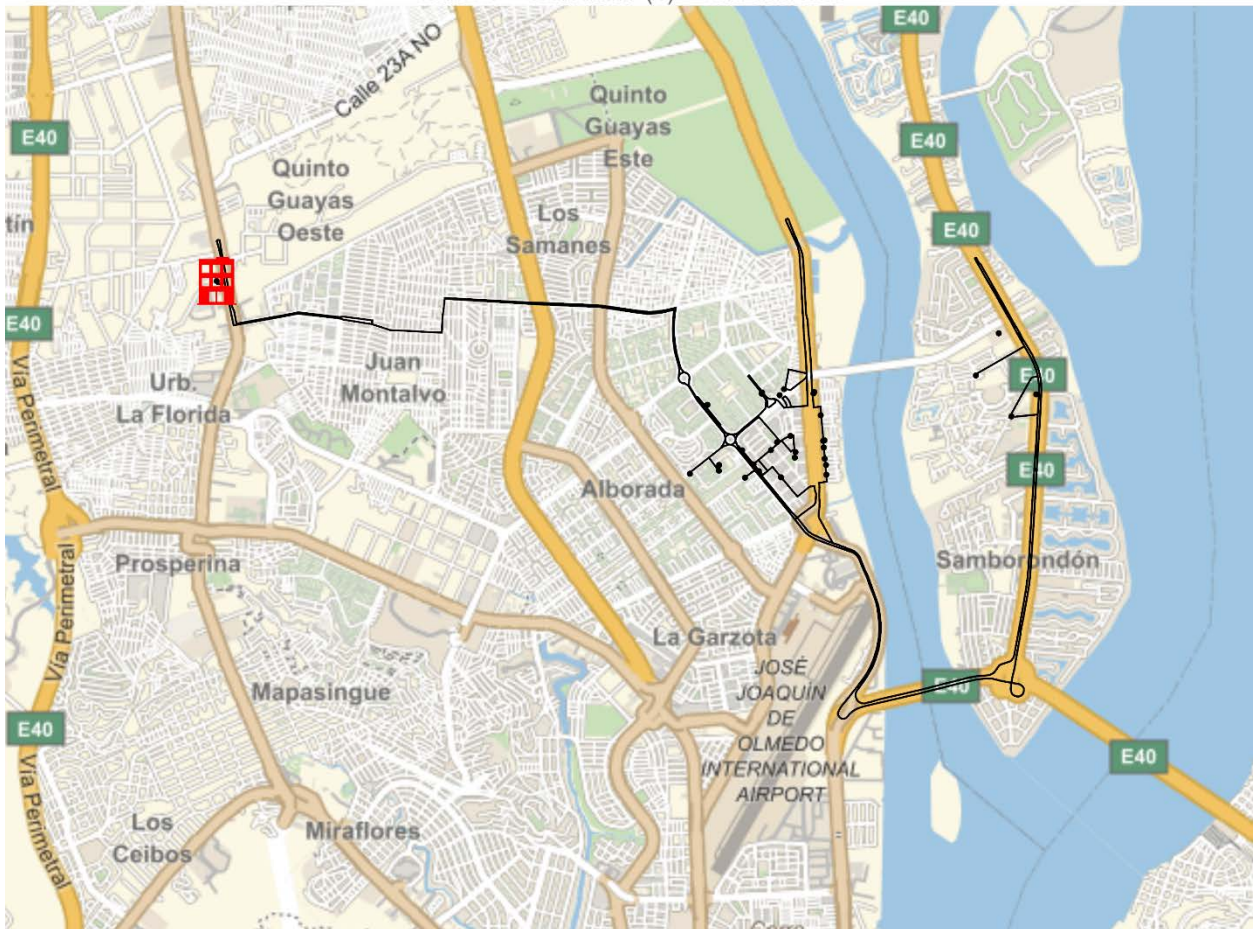
Ruta 27 – Viernes (6) – 15.5954 km.



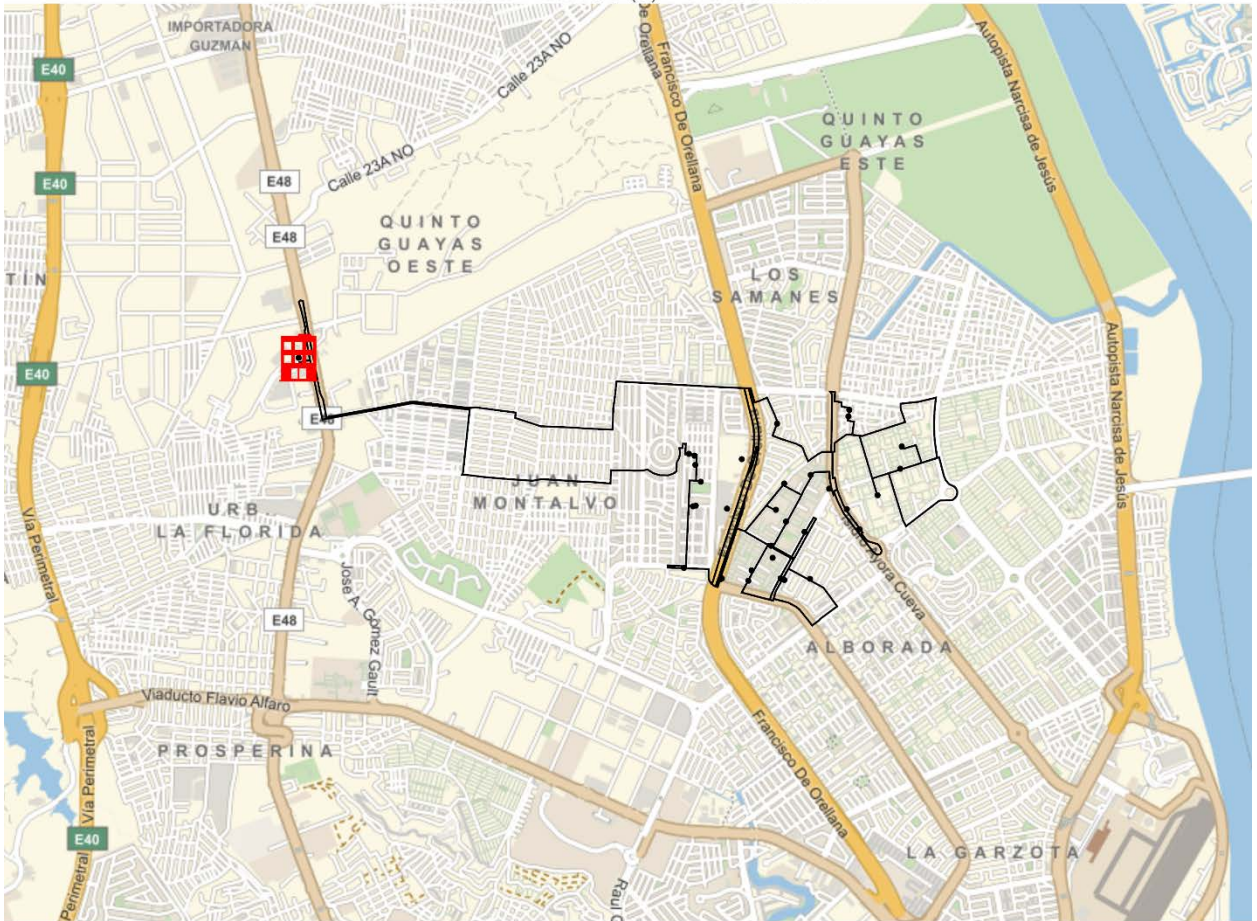
Ruta 27 – Jueves (5) – 50.3509 km.



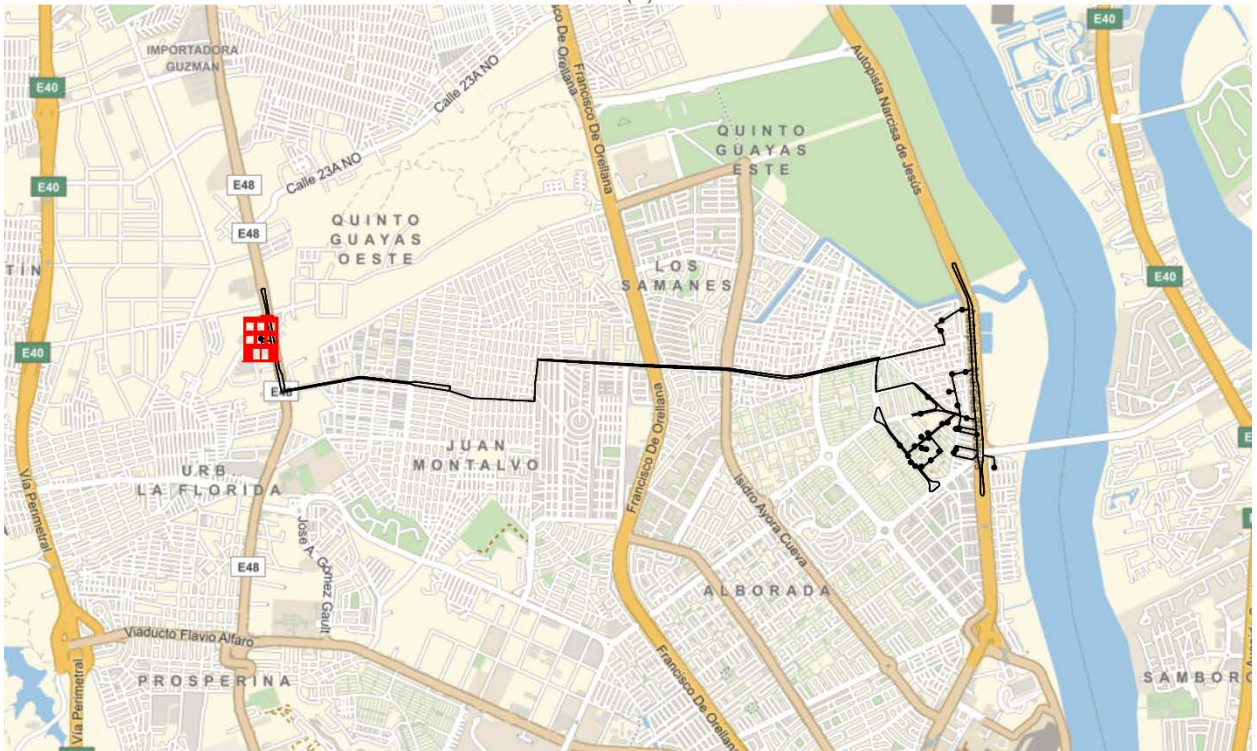
Ruta 27 – Miércoles (4) – 23.7333 km.



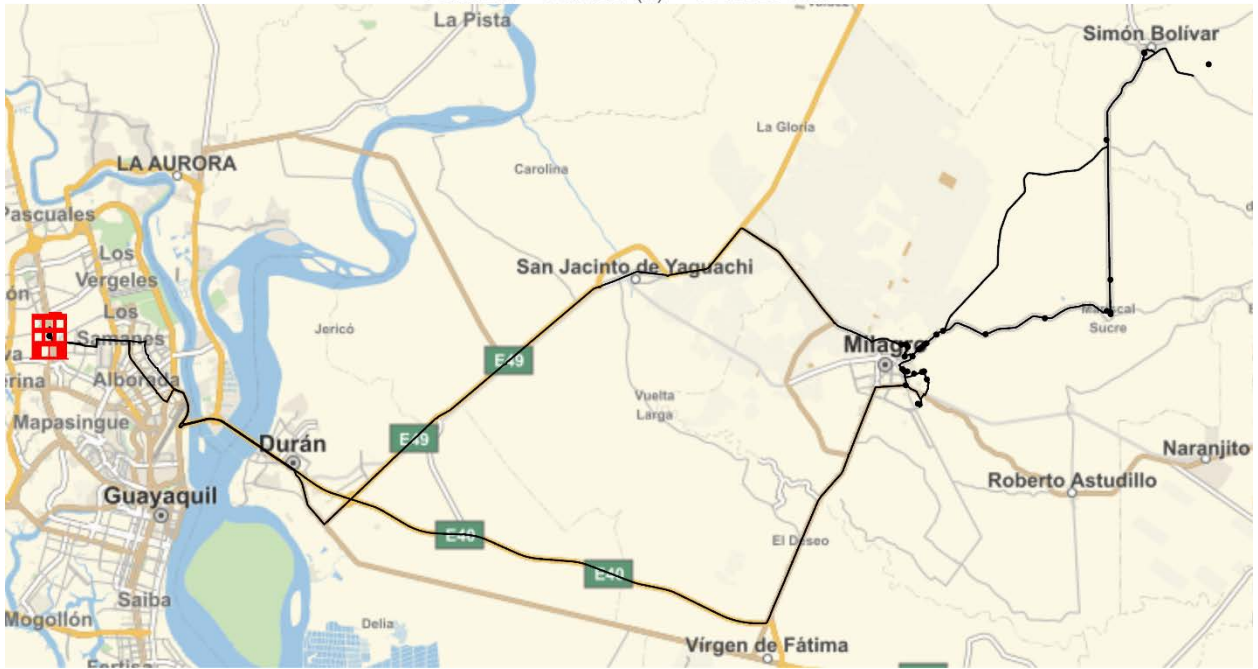
Ruta 27 – Martes (3) – 14.4352 km.



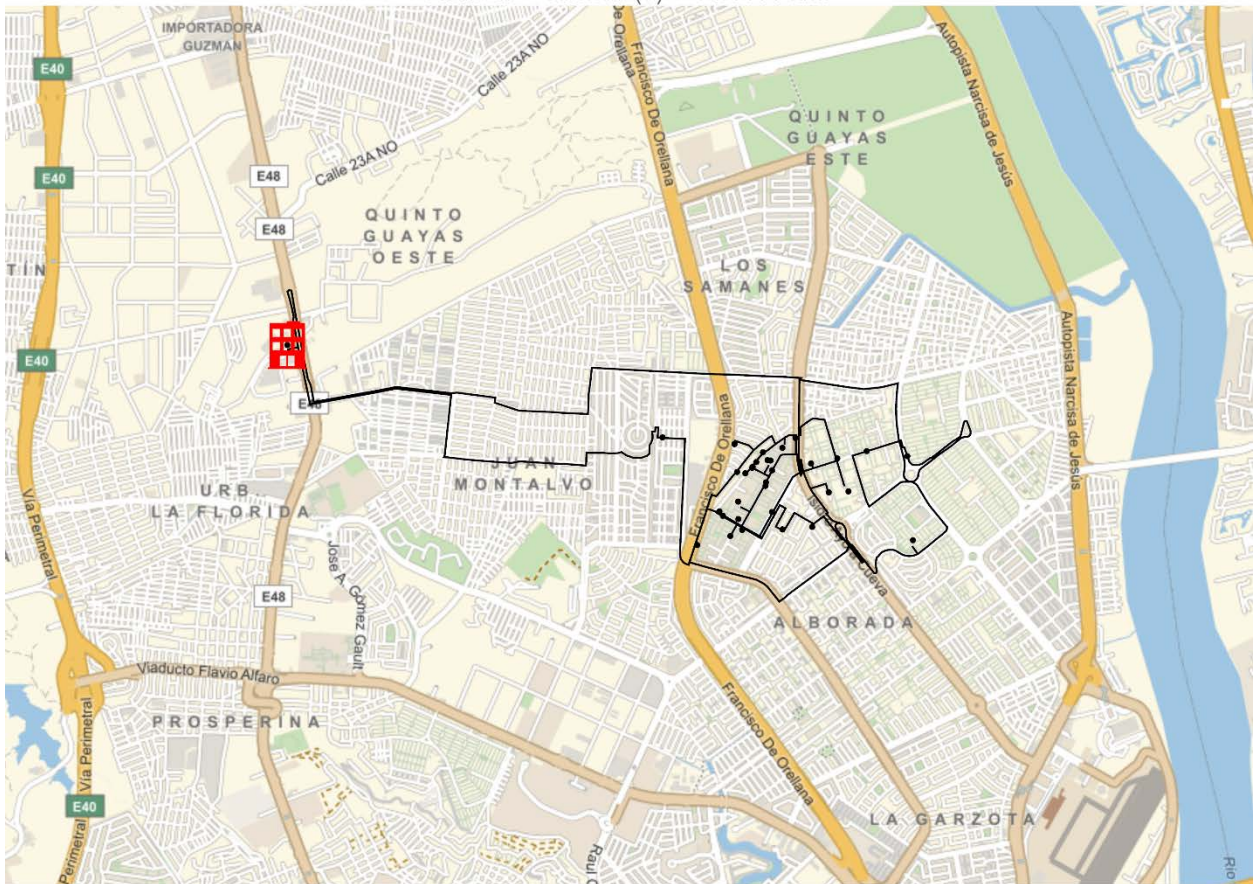
Ruta 27 – Lunes (2) – 15.8198 km.



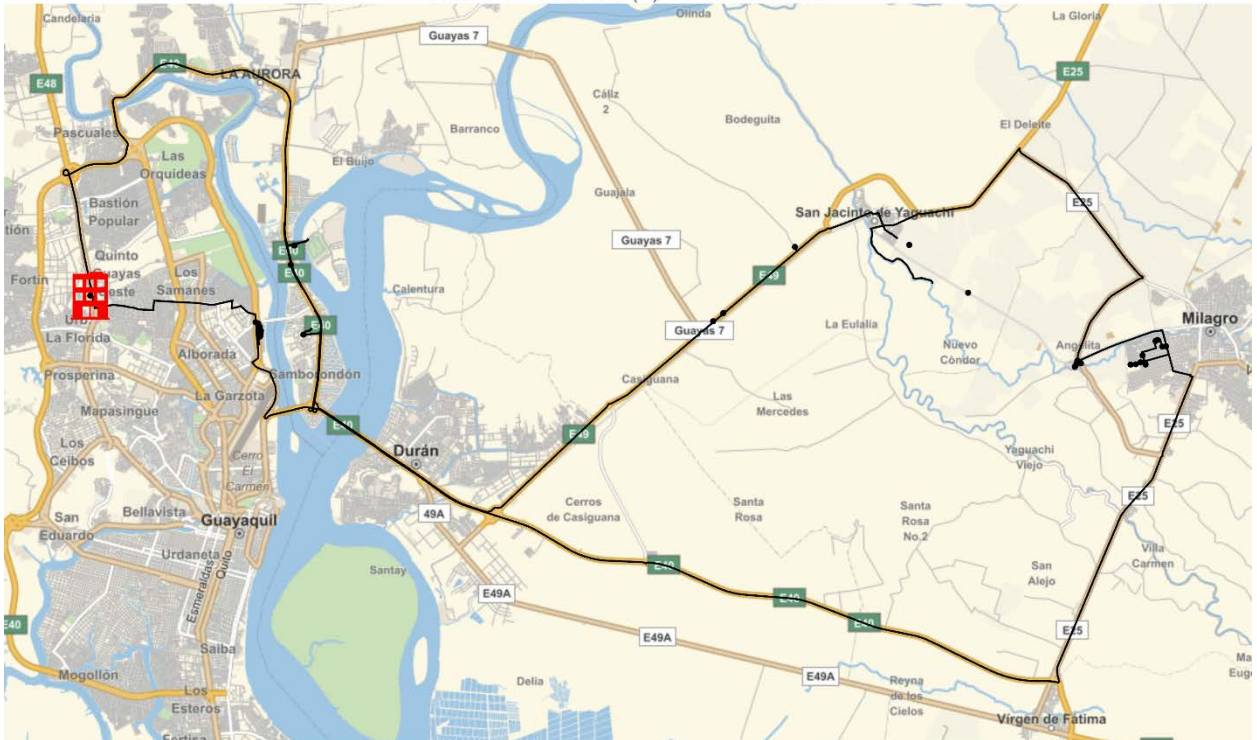
Ruta 26 – Sábado (7) – 141.727 km.



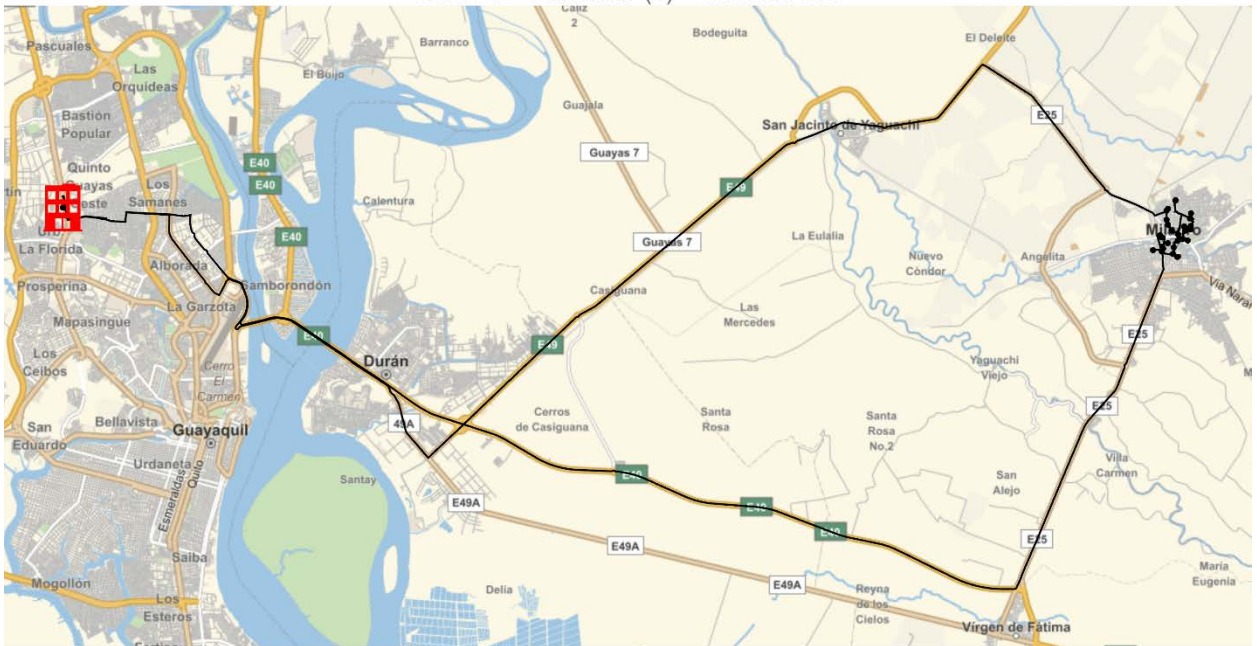
Ruta 26 – Viernes (6) – 15.4499 km.



Ruta 26 – Jueves (5) – 87.2758 km.



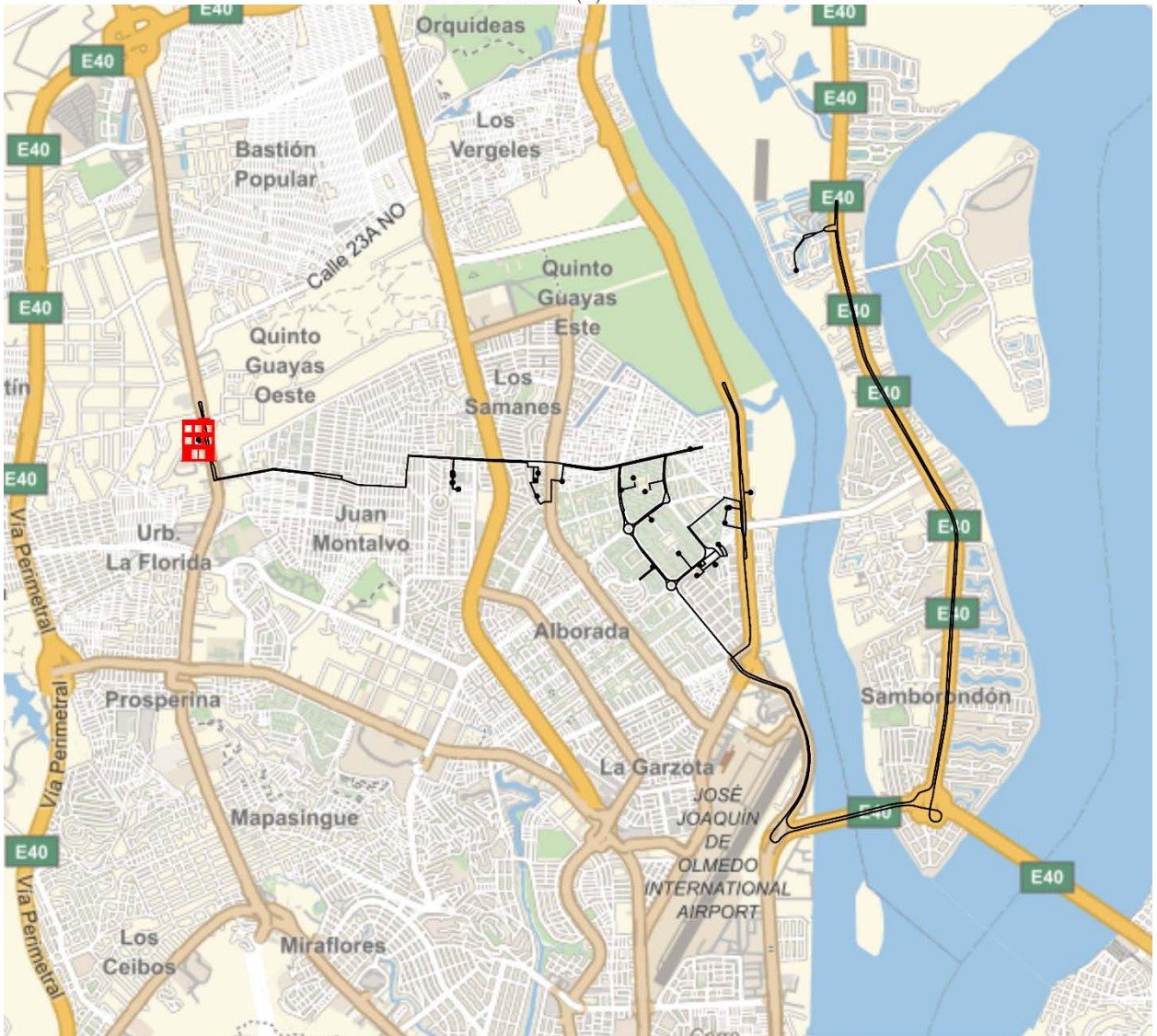
Ruta 26 – Miércoles (4) – 85.0593 km.



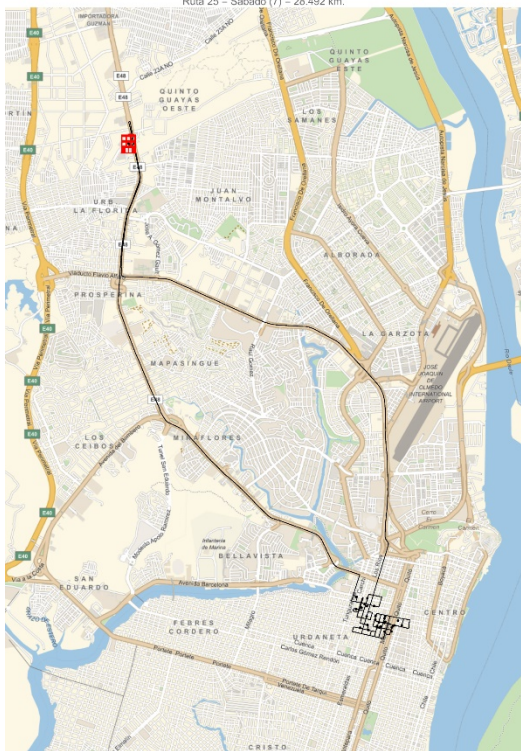
Ruta 26 – Martes (3) – 89.2511 km.



Ruta 26 – Lunes (2) – 19.9362 km.



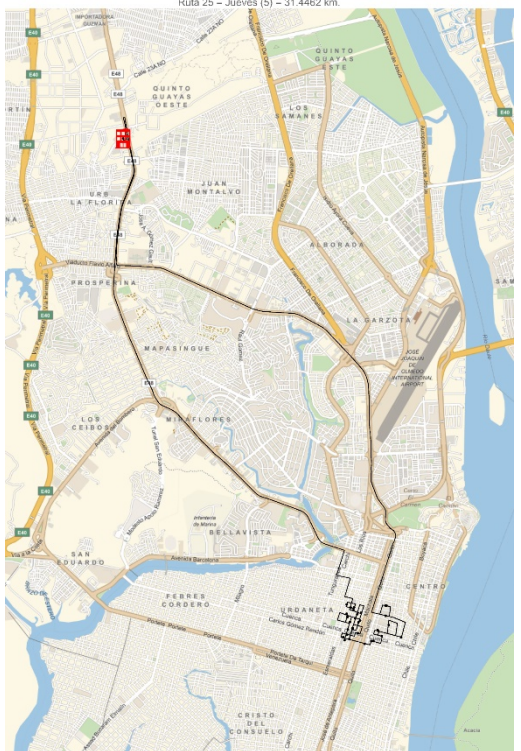
Ruta 25 - Sábado (7) - 28.492 km.



Ruta 25 - Viernes (8) - 29.598 km.



Ruta 25 - Jueves (5) - 31.4482 km.



Ruta 25 - Miércoles (4) - 28.375 km.





Ruta 25 – Martes (3) – 31.836 km.



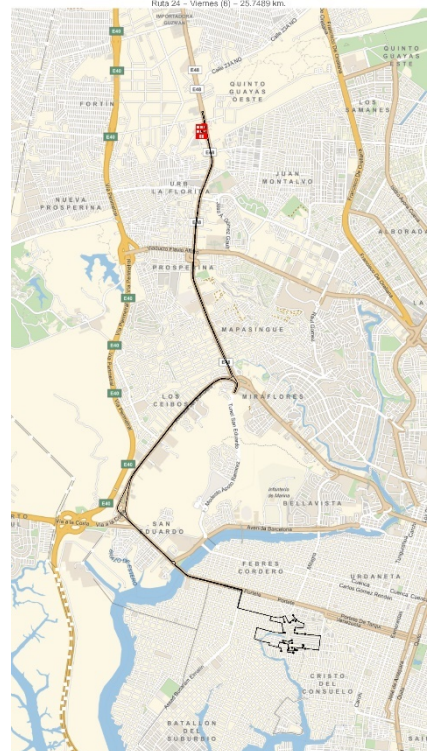
Ruta 25 – Lunes (2) – 33.3002 km.



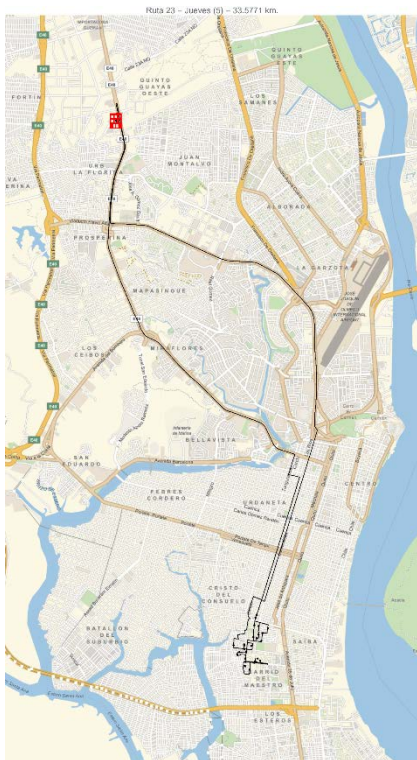
Ruta 24 – Sábado (7) – 26.7217 km.

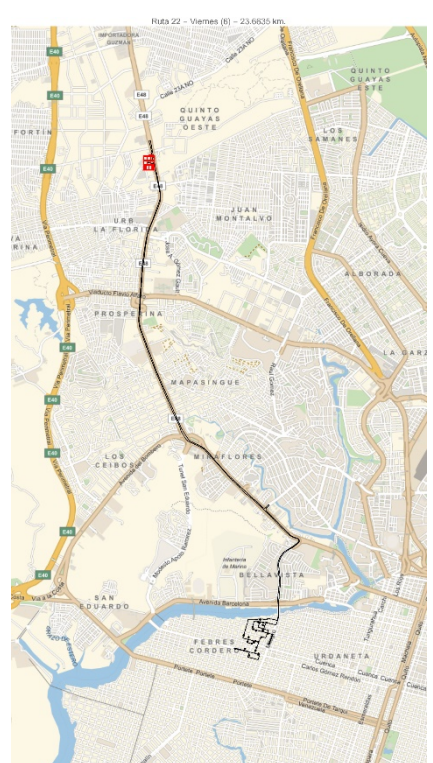
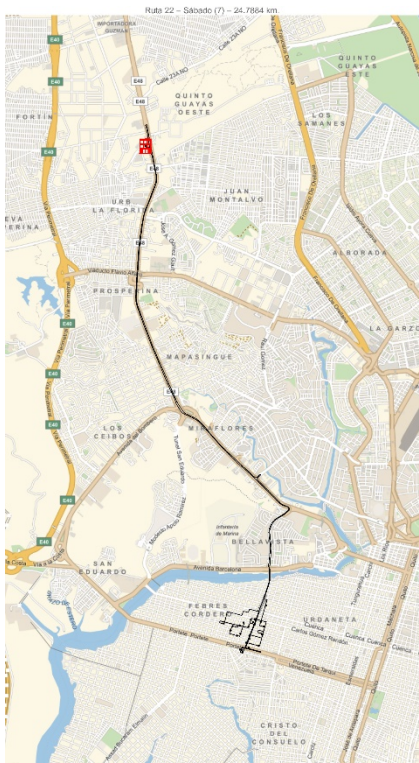


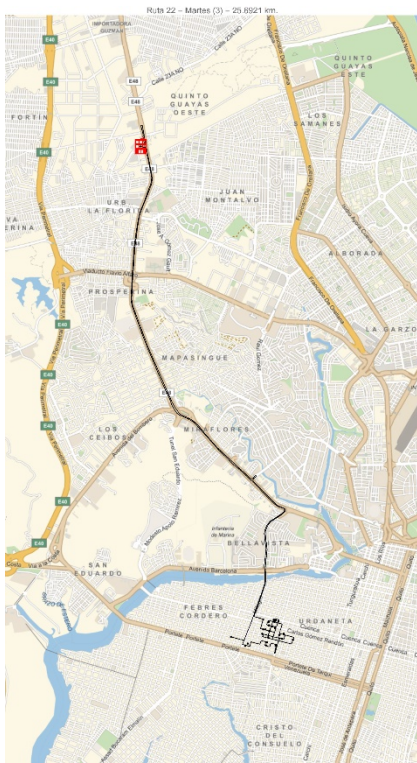
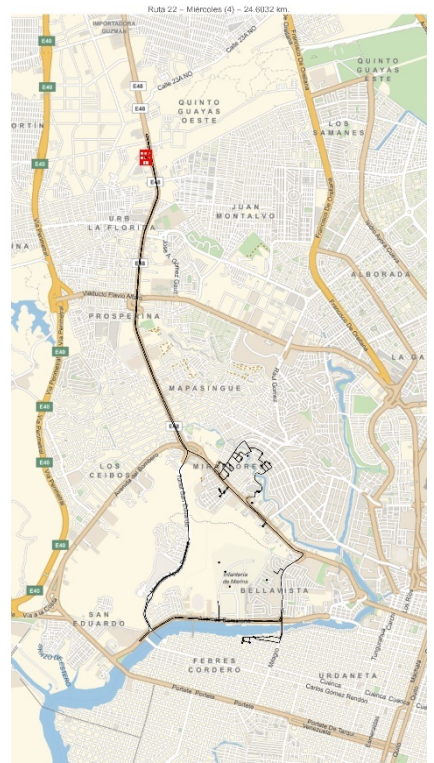
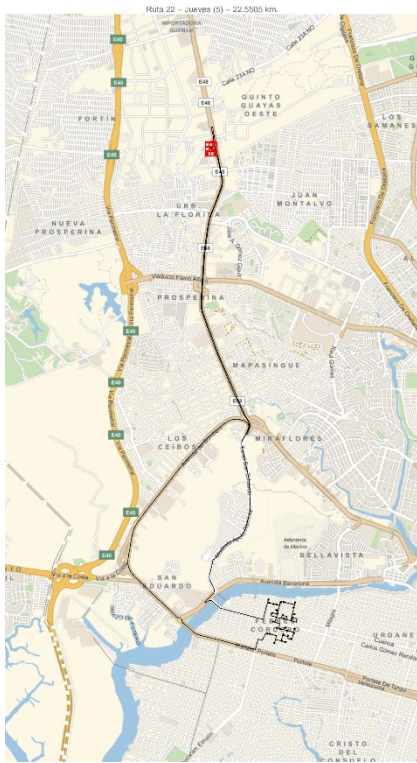
Ruta 24 – Viernes (6) – 25.7489 km.







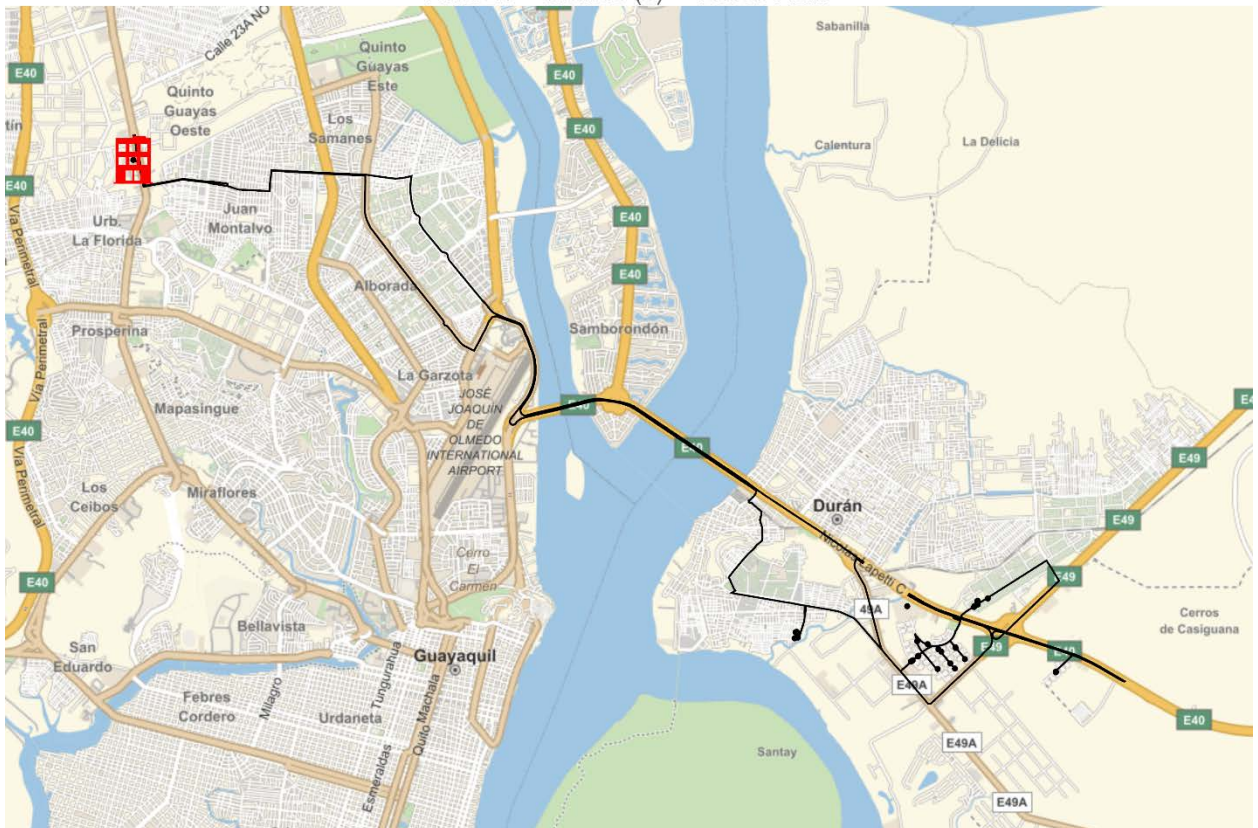




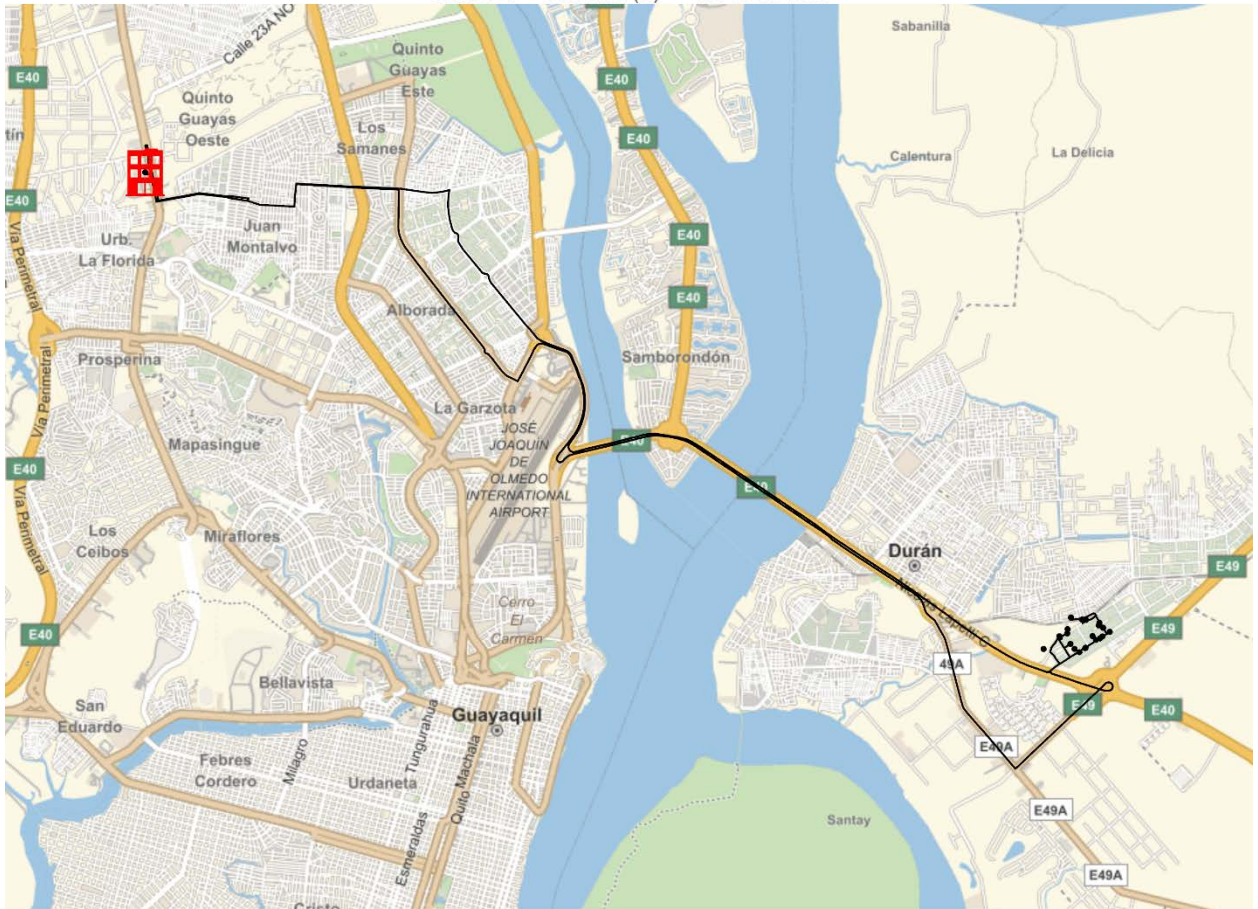
Ruta 21 – Sábado (7) – 38.2897 km.



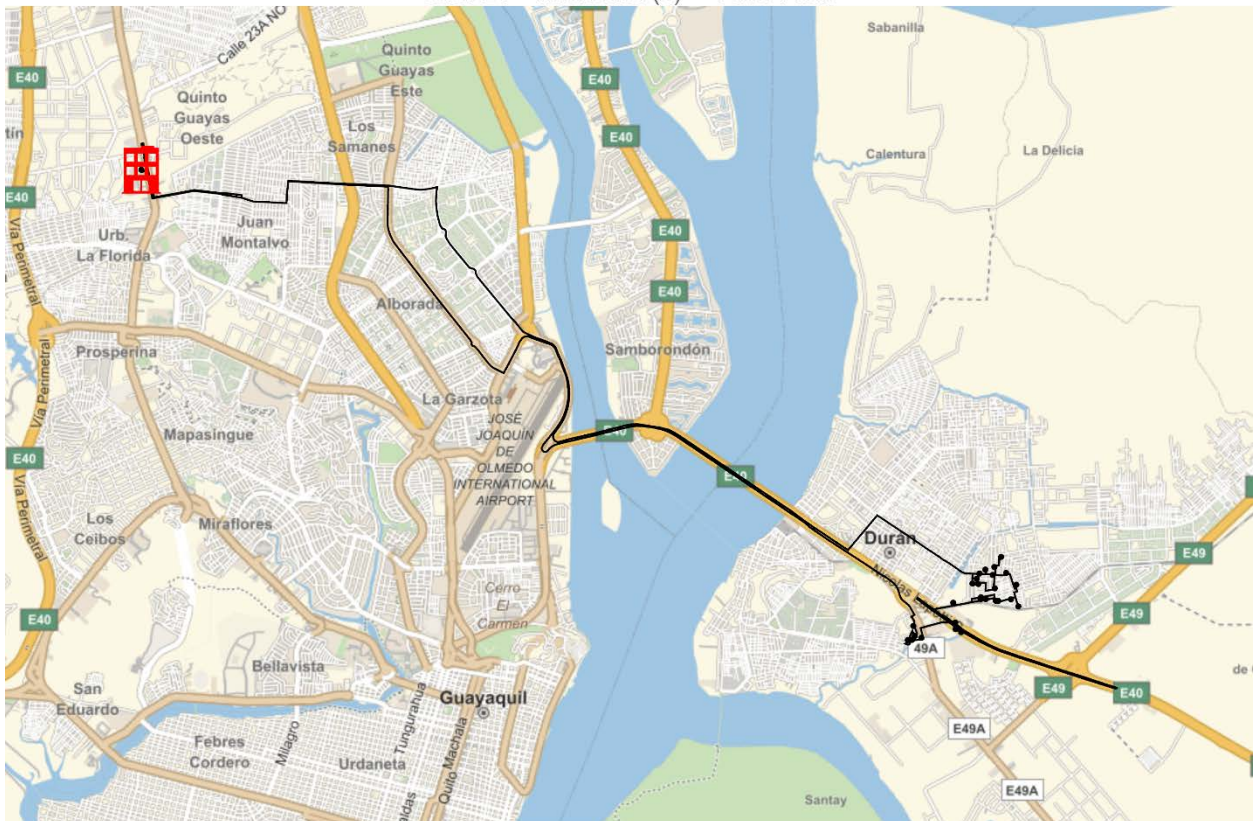
Ruta 21 – Viernes (6) – 48.8371 km.



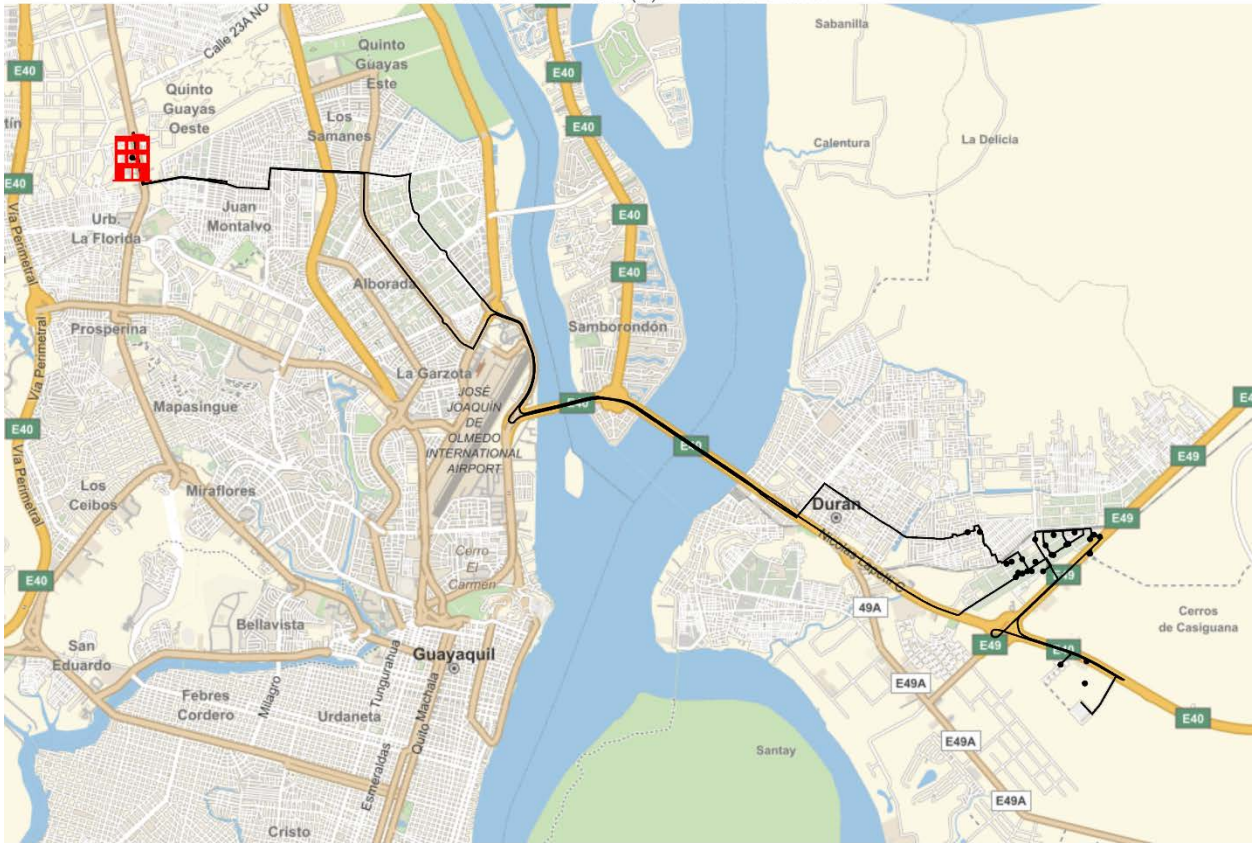
Ruta 21 – Jueves (5) – 43.1523 km.



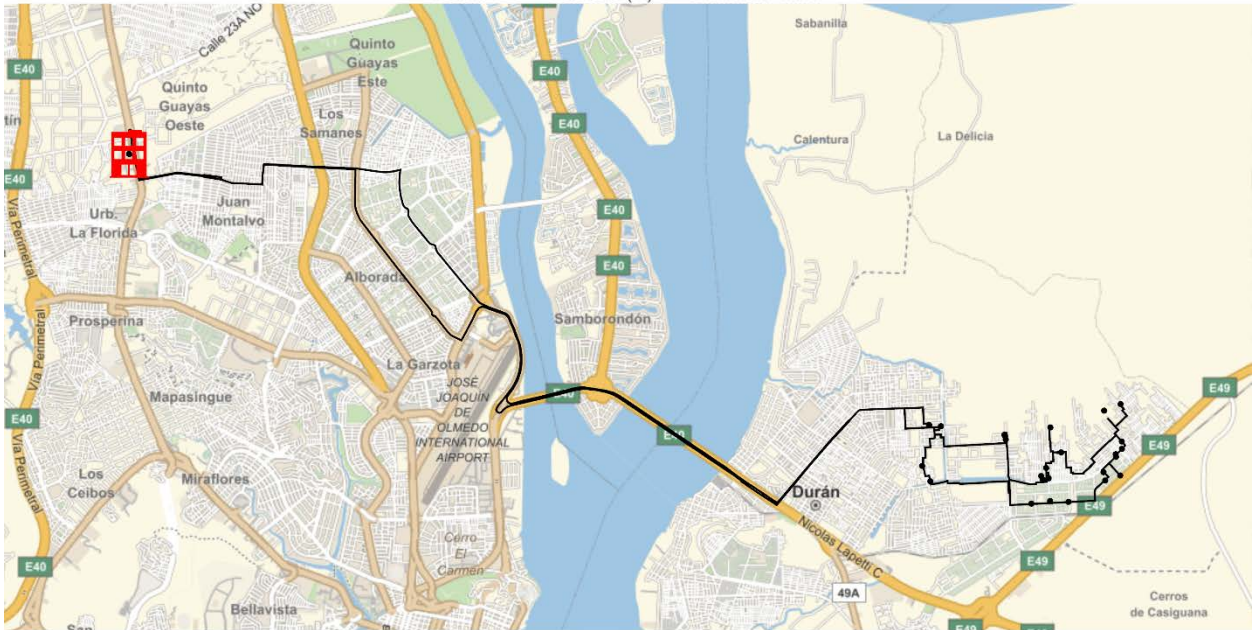
Ruta 21 – Miércoles (4) – 41.864 km.



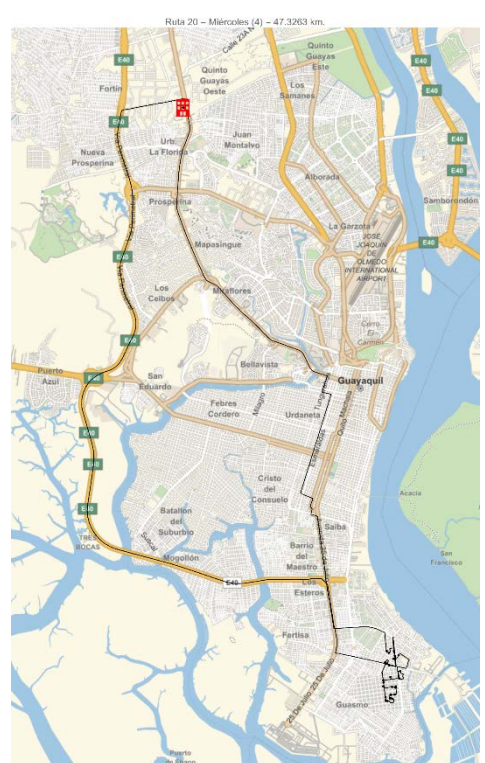
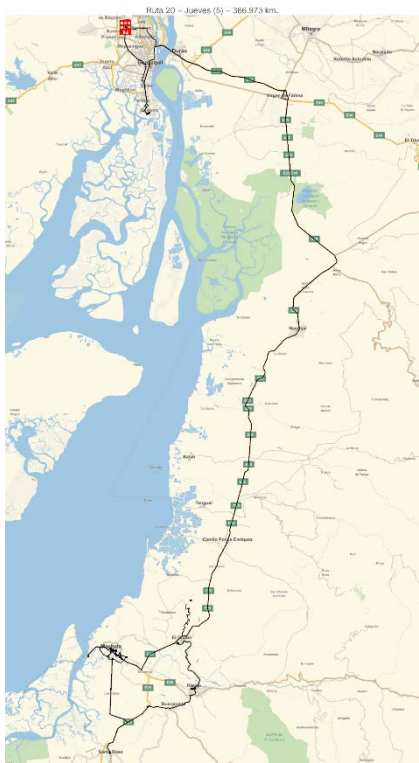
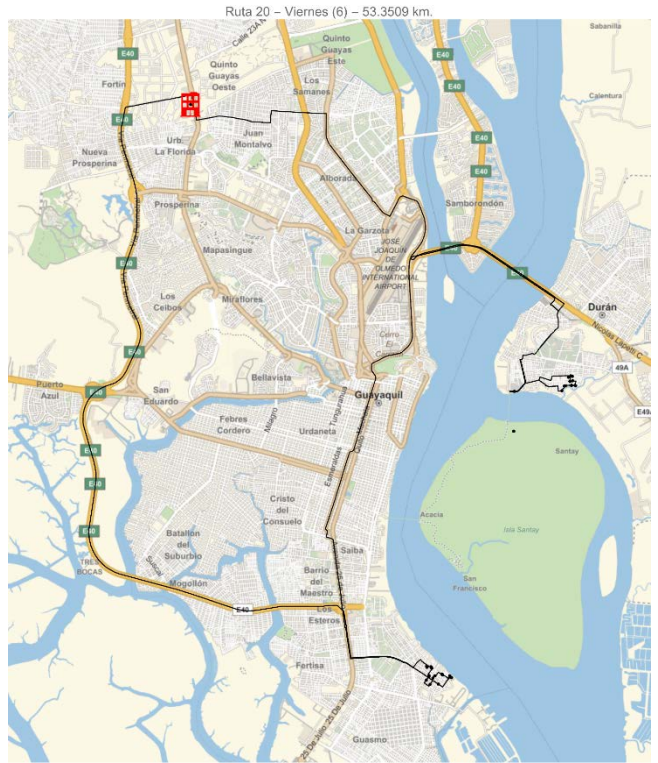
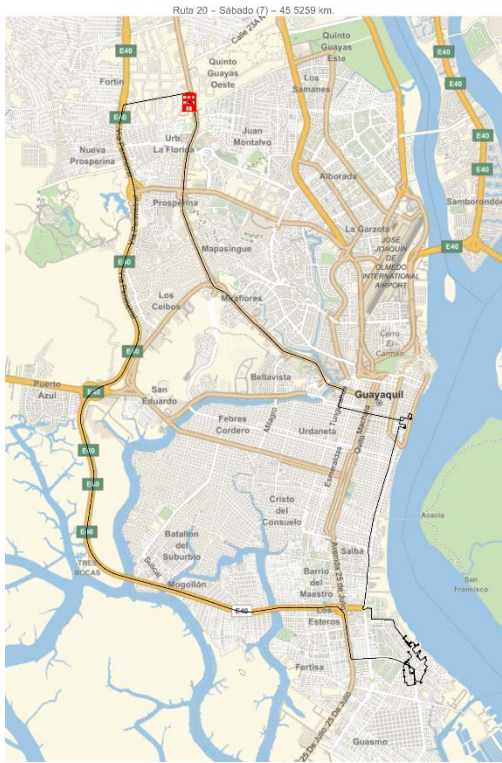
Ruta 21 – Martes (3) – 48.5576 km.

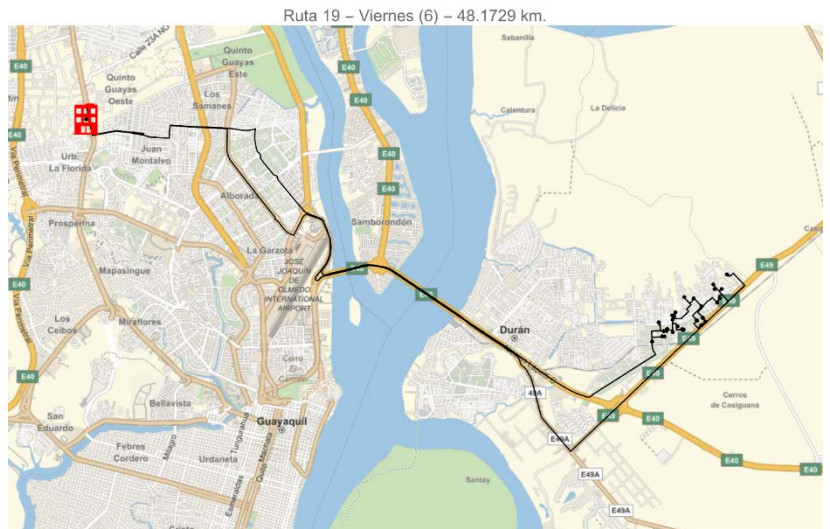
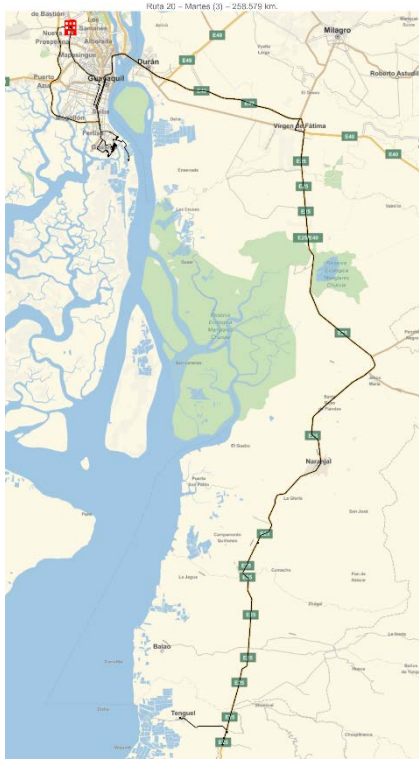


Ruta 21 – Lunes (2) – 46.8058 km.

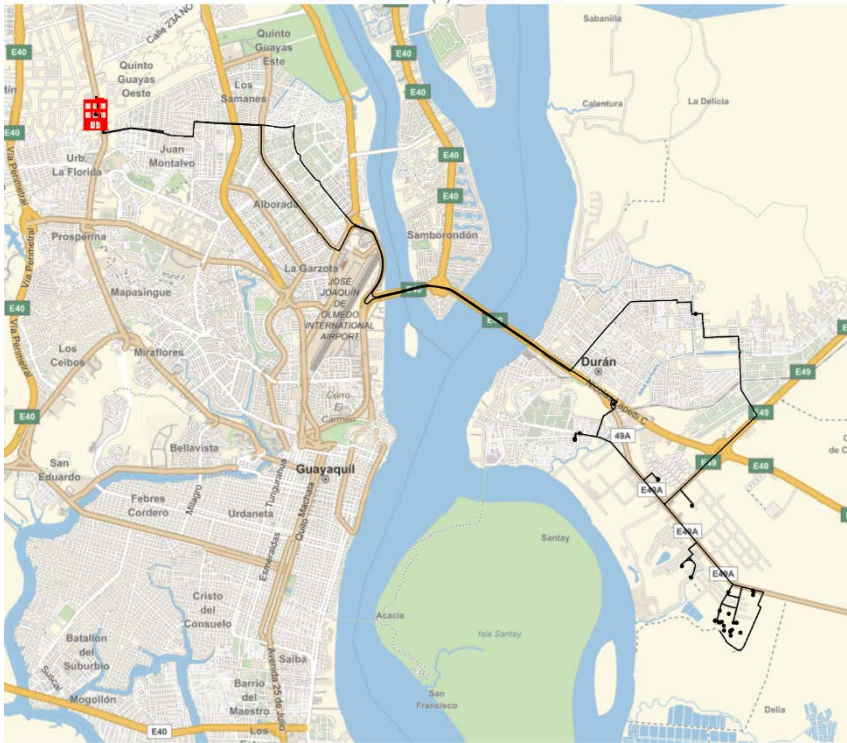




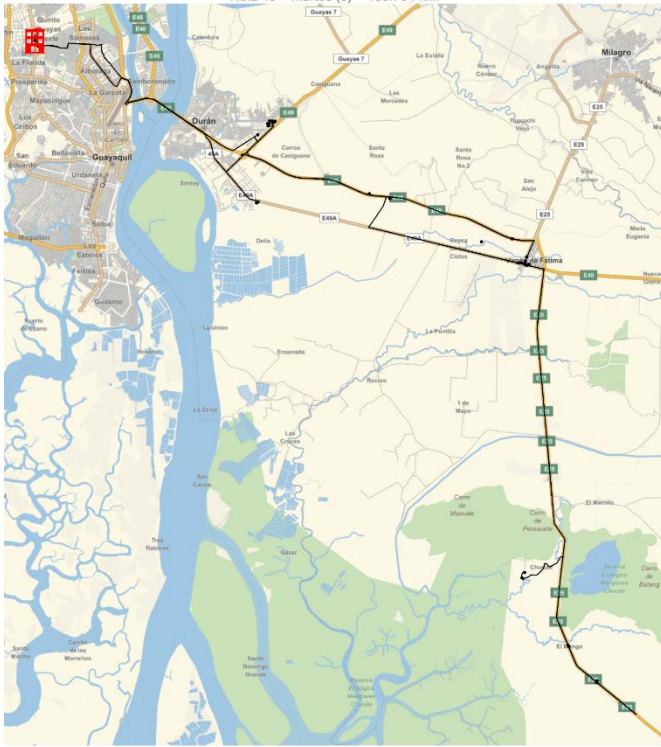




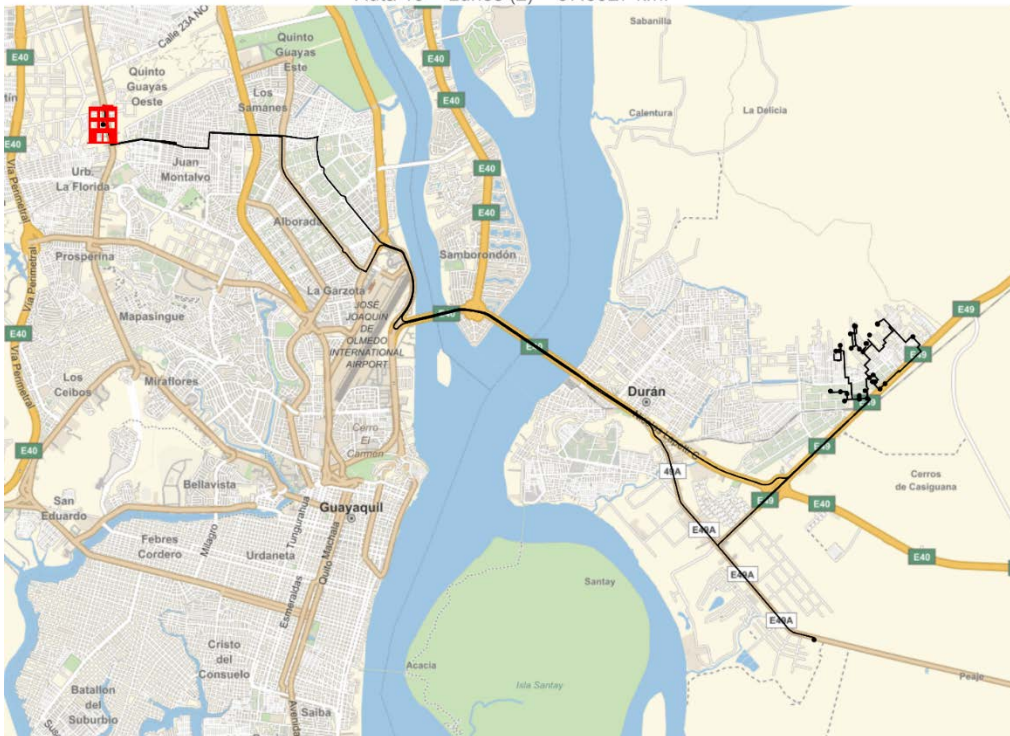
Ruta 19 – Jueves (5) – 55.3622 km.



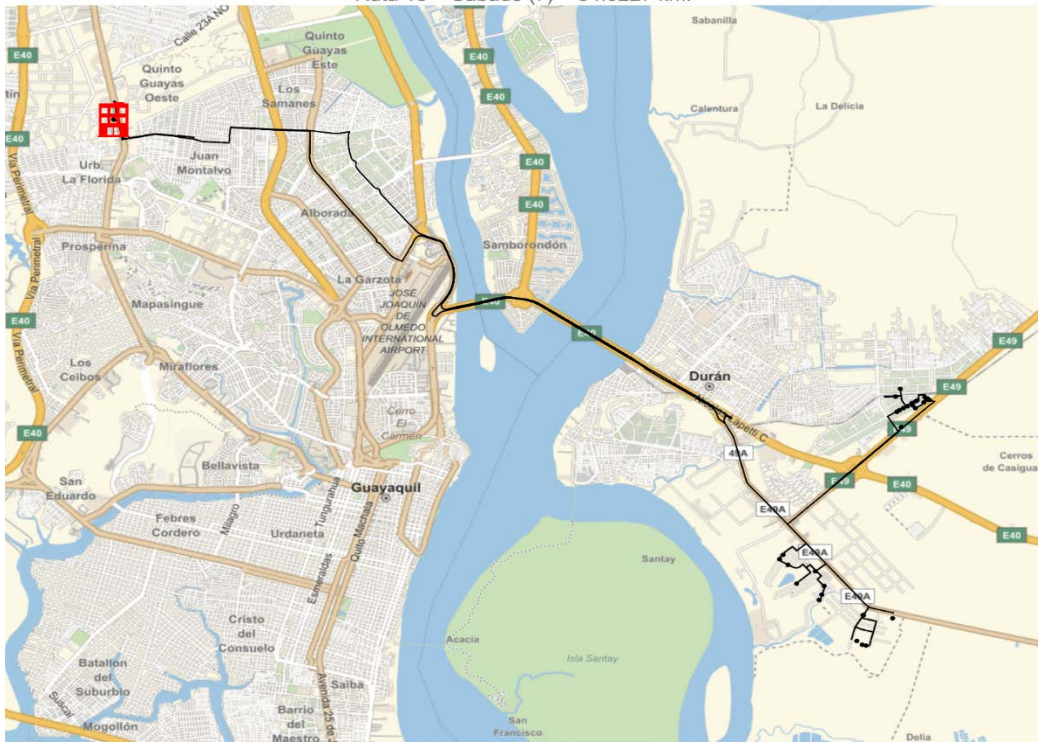
Ruta 19 – Martes (3) – 160.704 km.



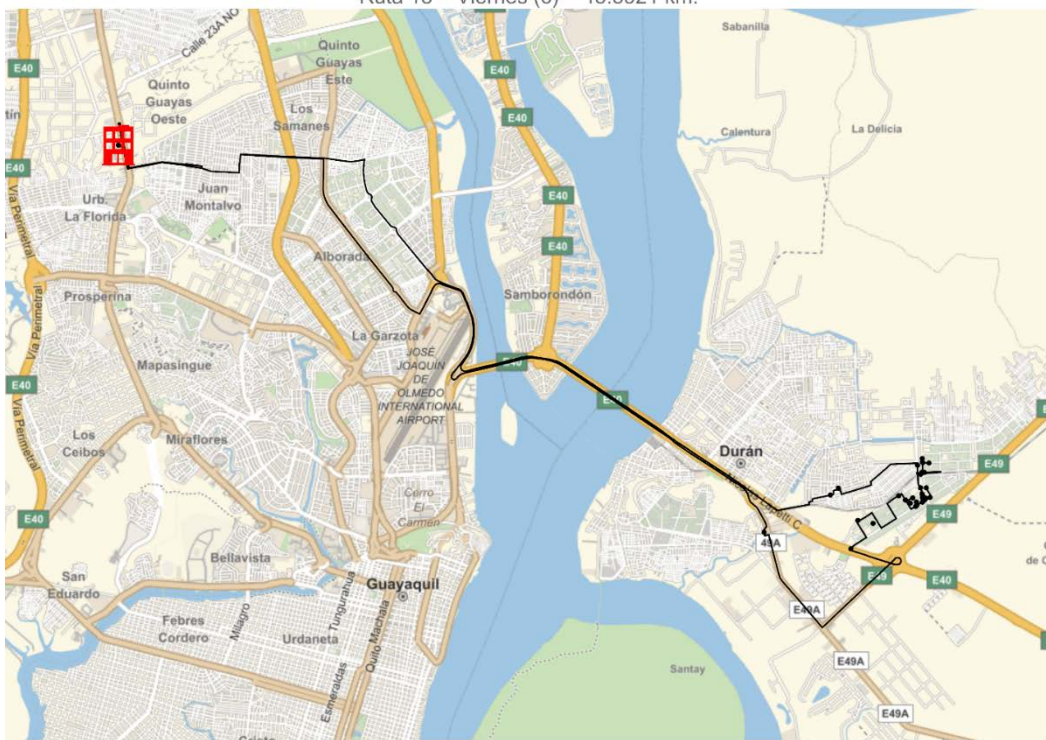
Ruta 19 – Lunes (2) – 57.8927 km.



Ruta 18 – Sábado (7) – 54.0227 km.



Ruta 18 – Viernes (6) – 45.5921 km.



Ruta 18 – Jueves (5) – 46.5097 km.



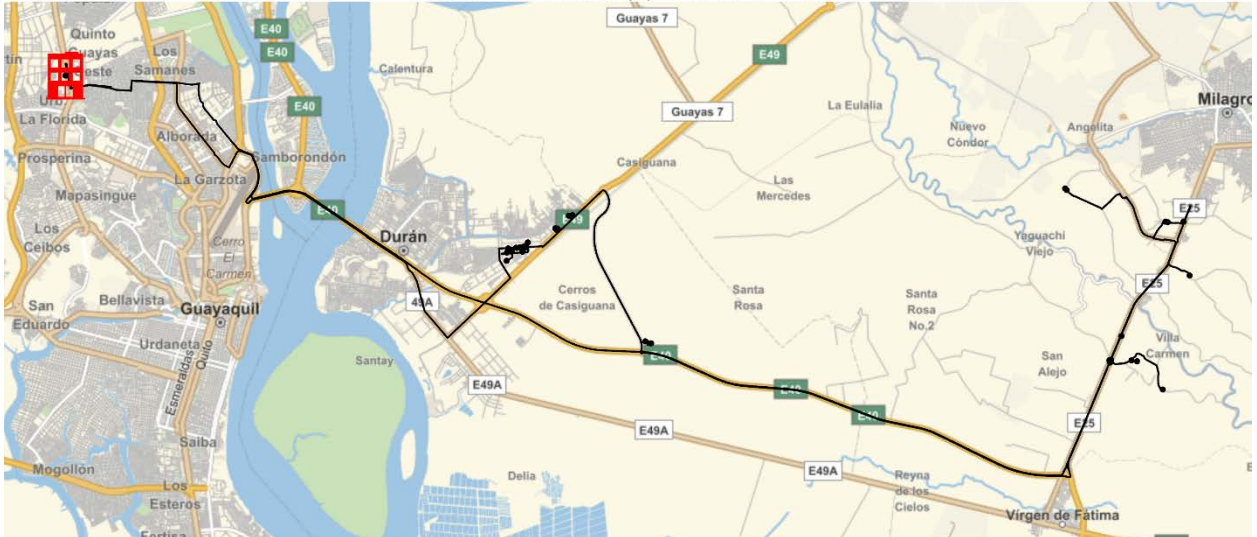
Ruta 18 – Miércoles (4) – 41.2843 km.



Ruta 18 – Martes (3) – 112.403 km.

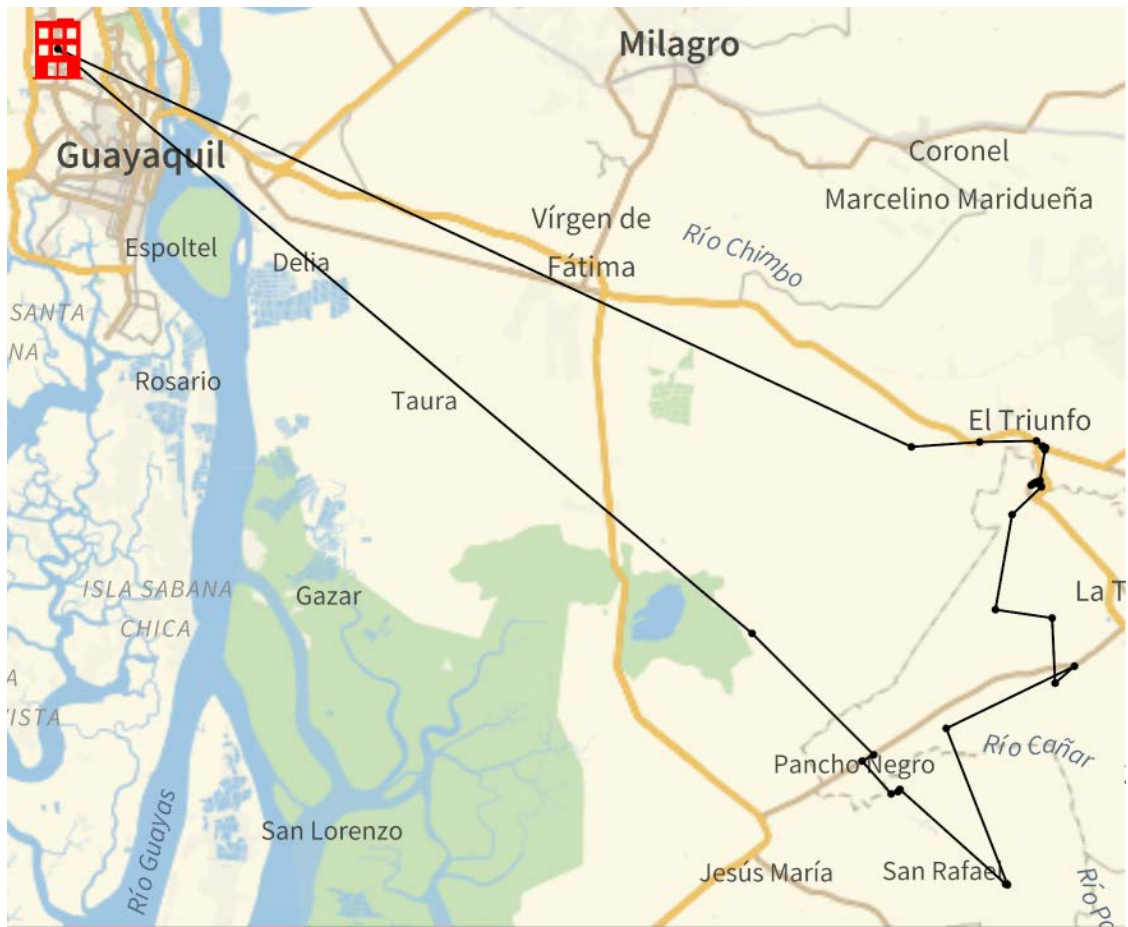


Ruta 18 – Lunes (2) – 100.922 km.

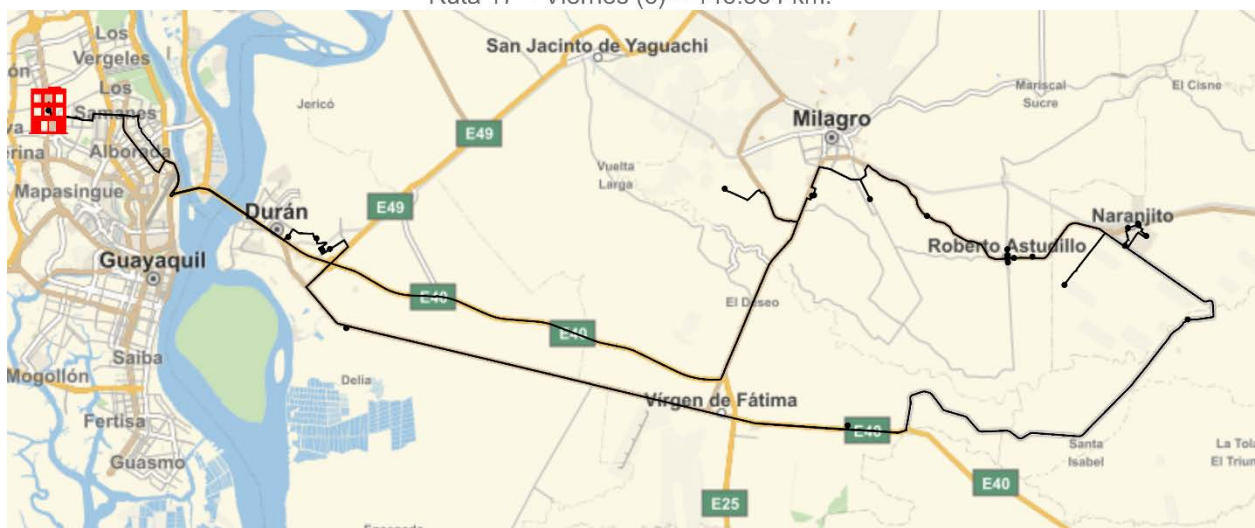


Ruta 17 – Sábado (7) – 273.757 km.



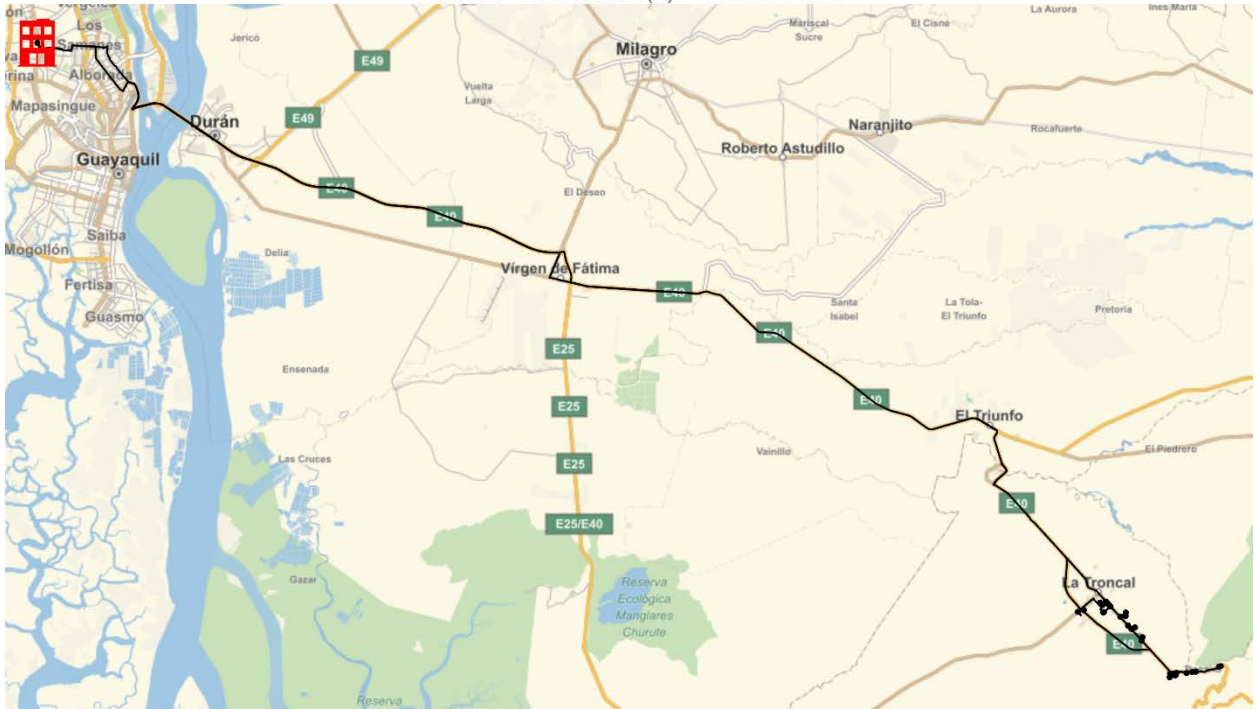


Ruta 17 – Viernes (6) – 146.564 km.

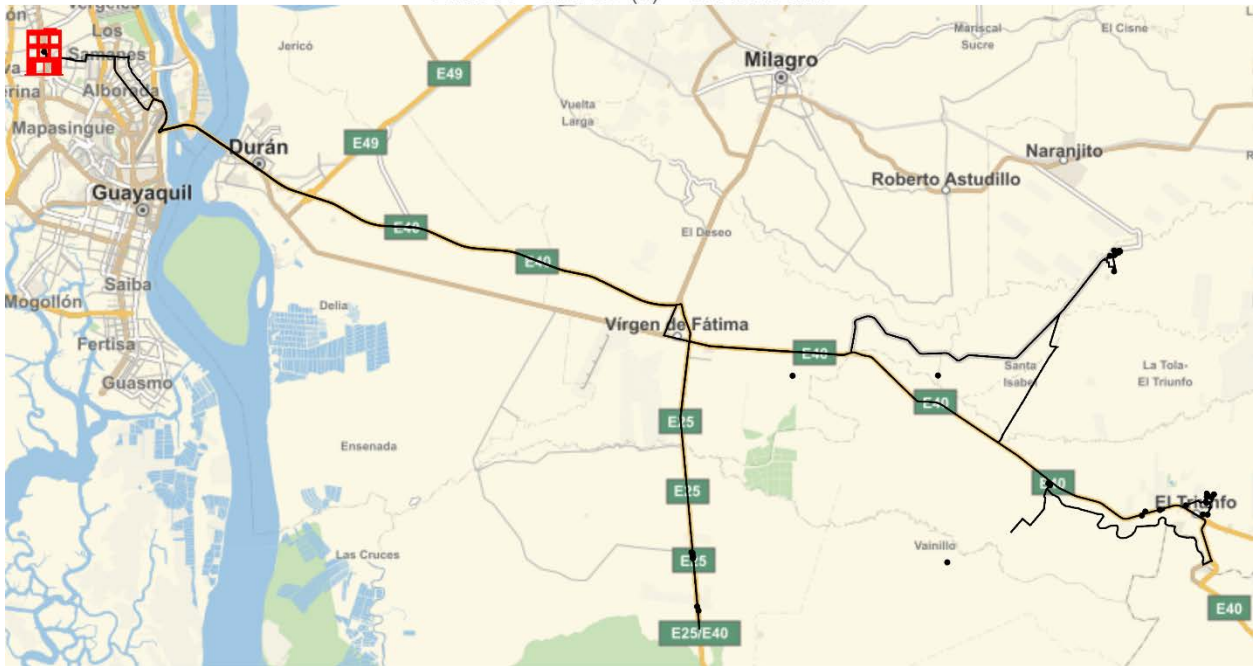




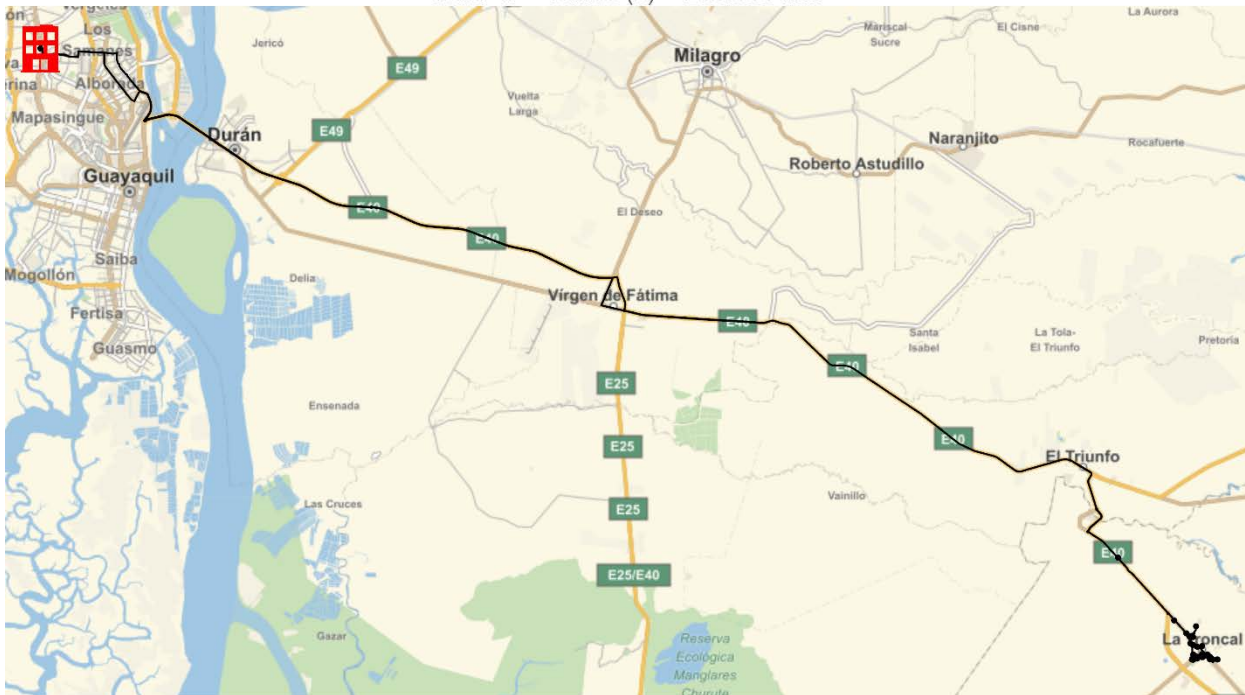
Ruta 17 – Miércoles (4) – 226.585 km.



Ruta 17 – Martes (3) – 197.806 km.



Ruta 17 – Lunes (2) – 203.553 km.



Ruta 16 – Sábado (7) – 256.852 km.



Ruta 16 – Viernes (6) – 244.647 km.



Ruta 16 – Miércoles (4) – 296.392 km.



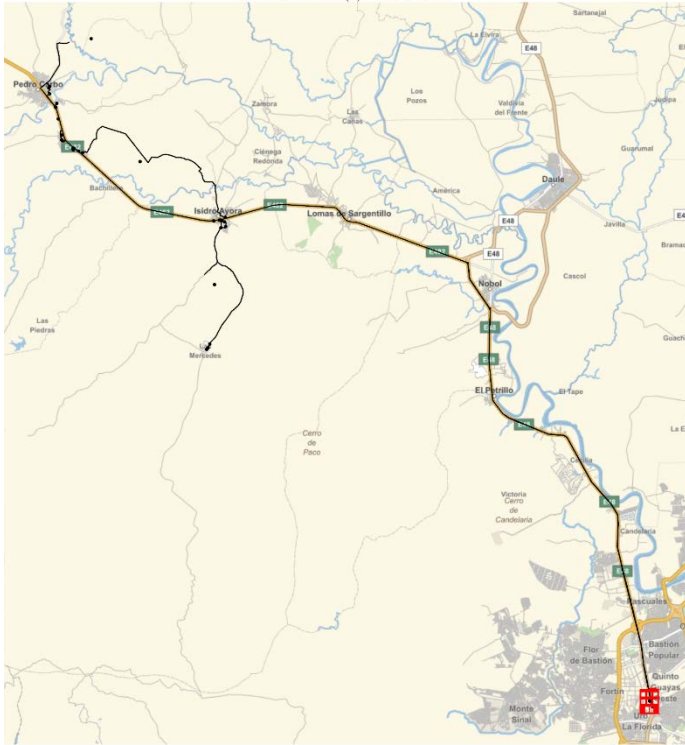
Ruta 16 – Jueves (5) – 240.795 km.



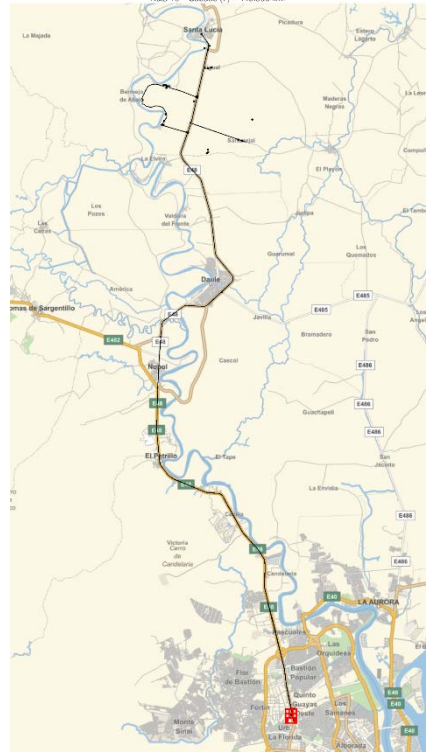
Ruta 16 – Martes (3) – 168.204 km.



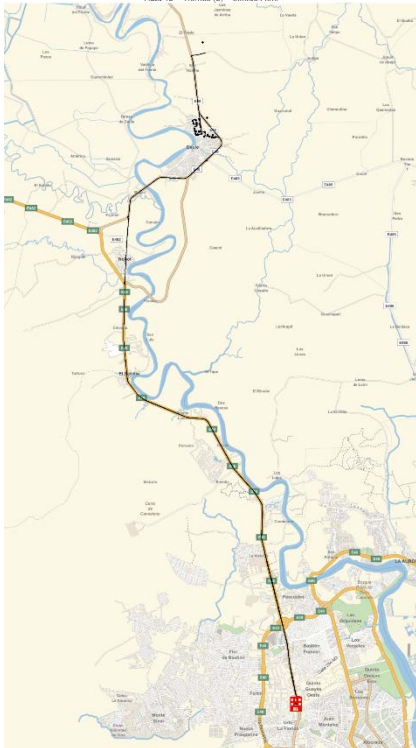
Ruta 16 – Lunes (2) – 140.983 km.



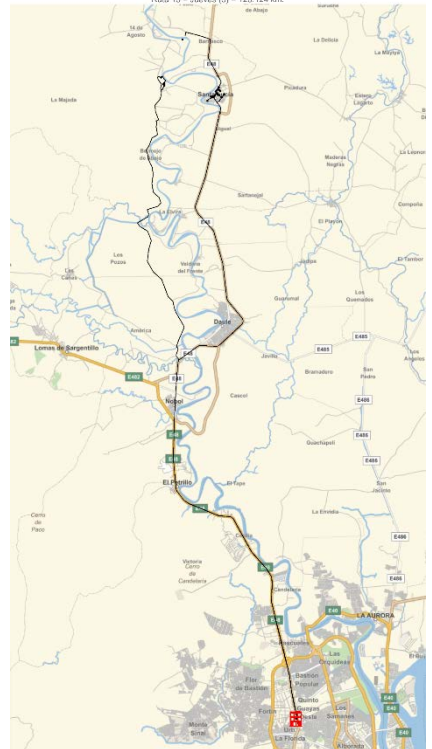
Ruta 15 – Sábado (7) – 113.865 km.

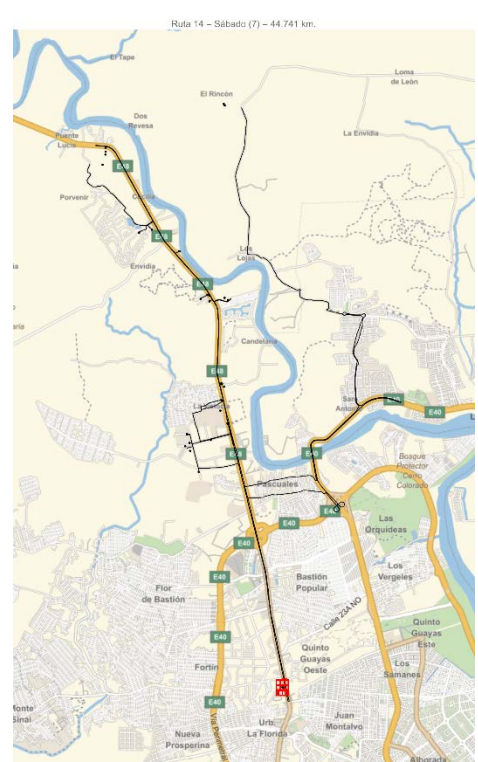
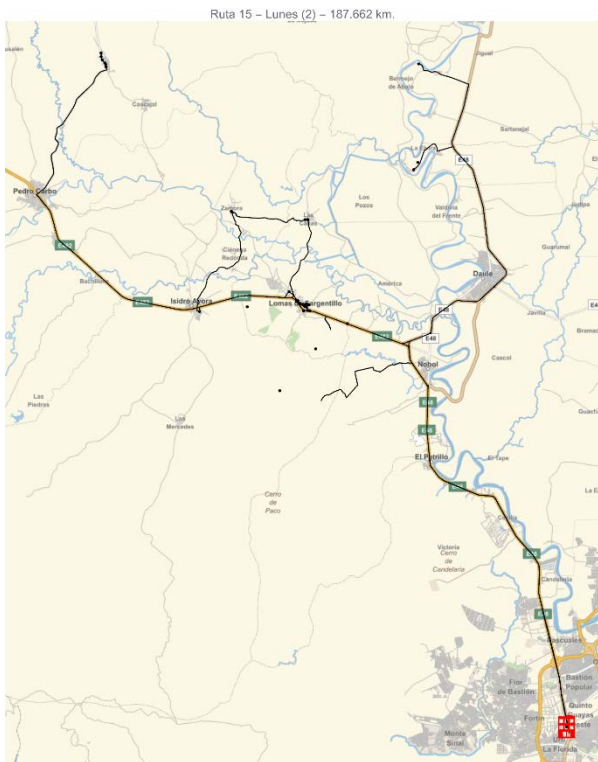
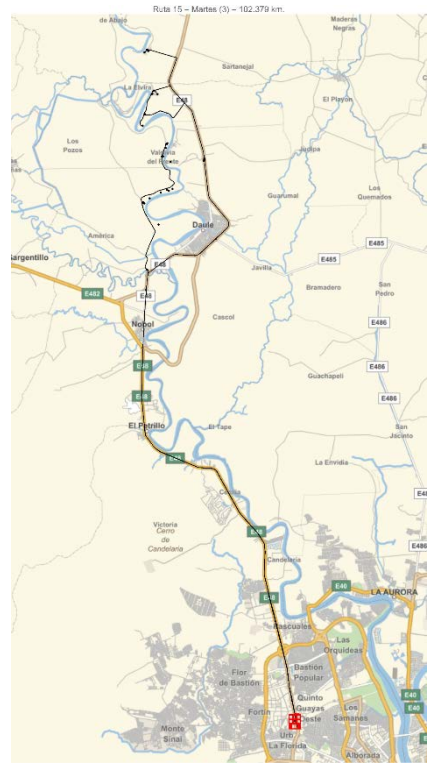
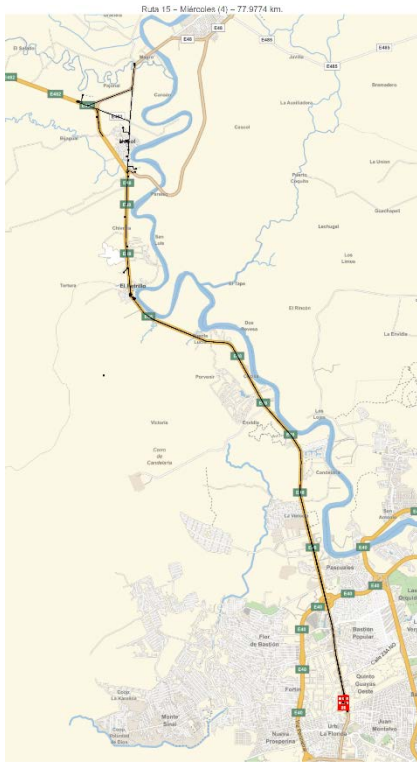


Ruta 15 – Viernes (8) – 80.384 km.

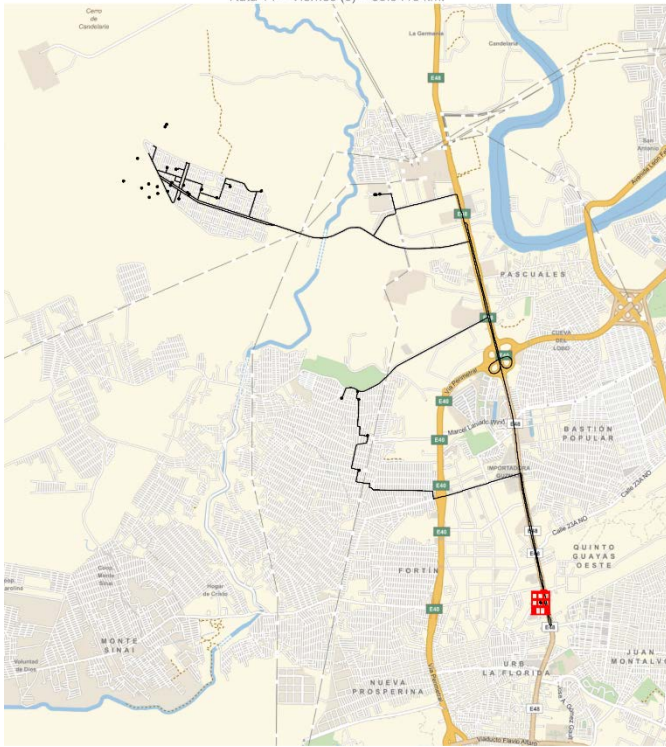


Ruta 15 – Jueves (5) – 120.124 km.

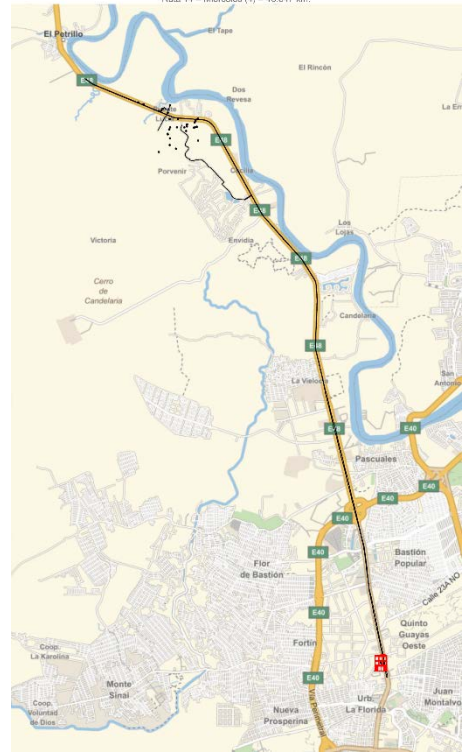




Ruta 14 – Viernes (6) – 33.9418 km.



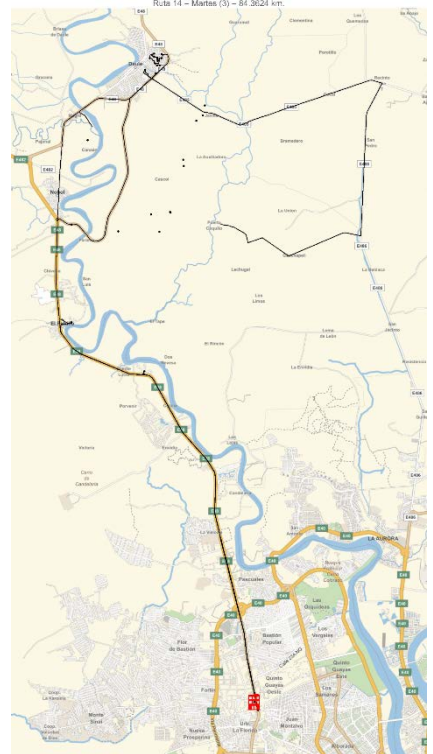
Ruta 14 – Miércoles (4) – 46.847 km.

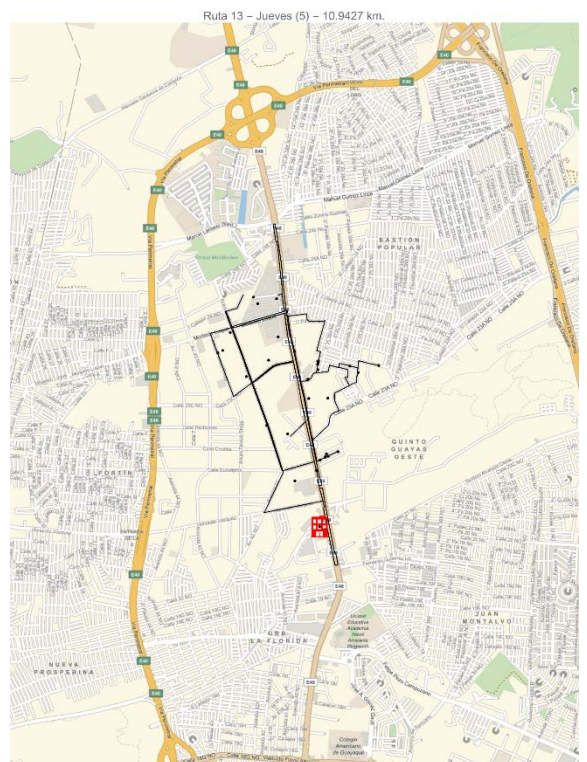
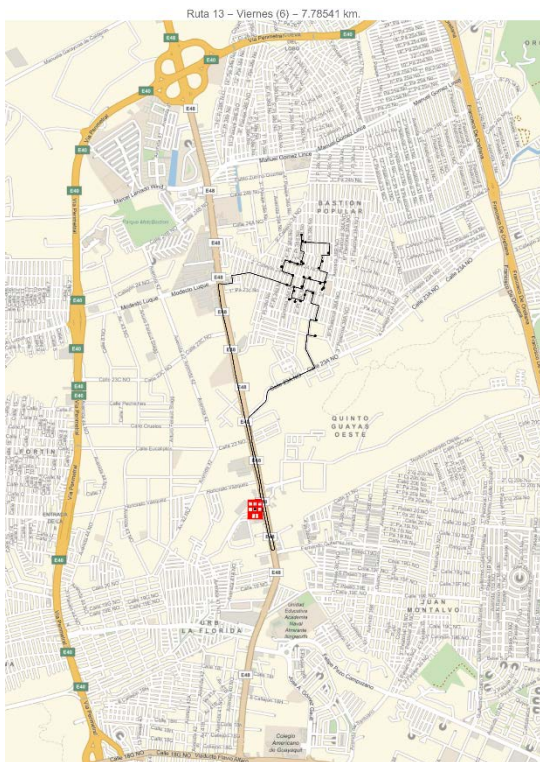
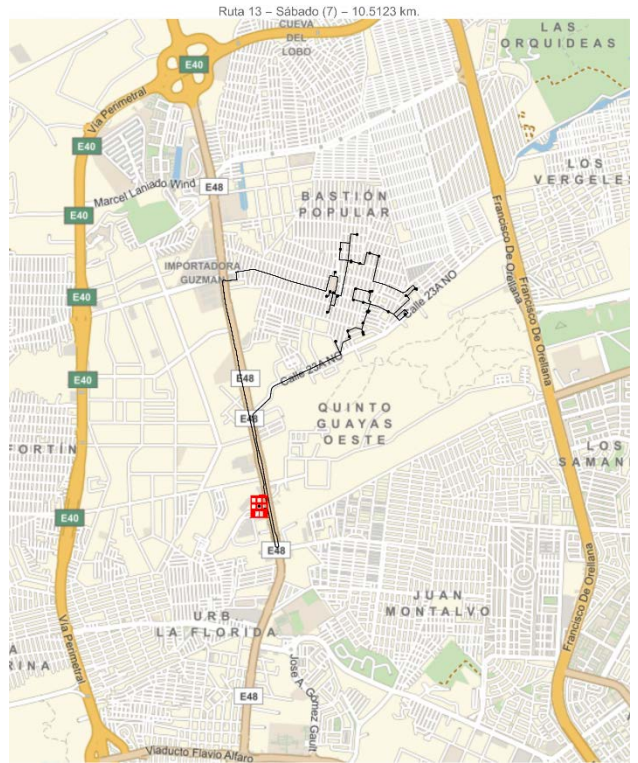
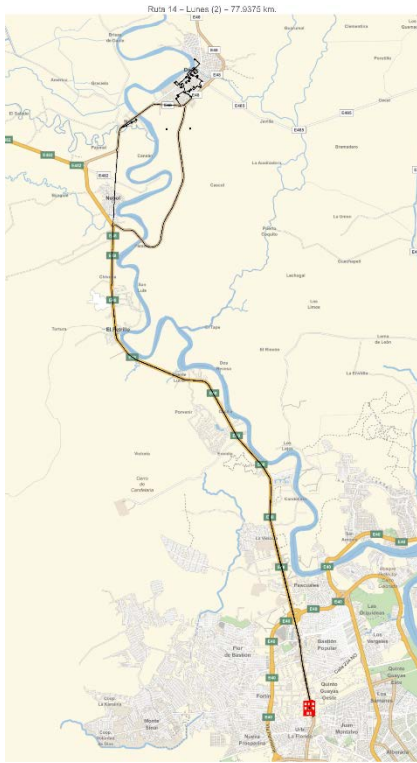


Ruta 14 – Jueves (5) – 29.4321 km.

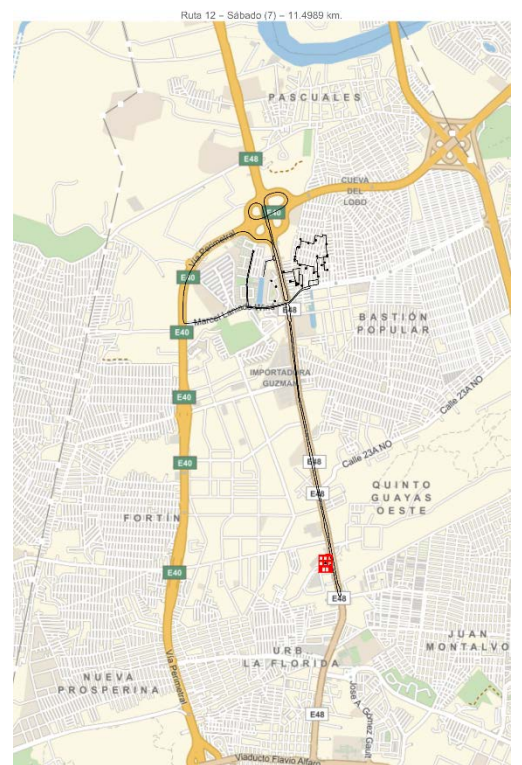
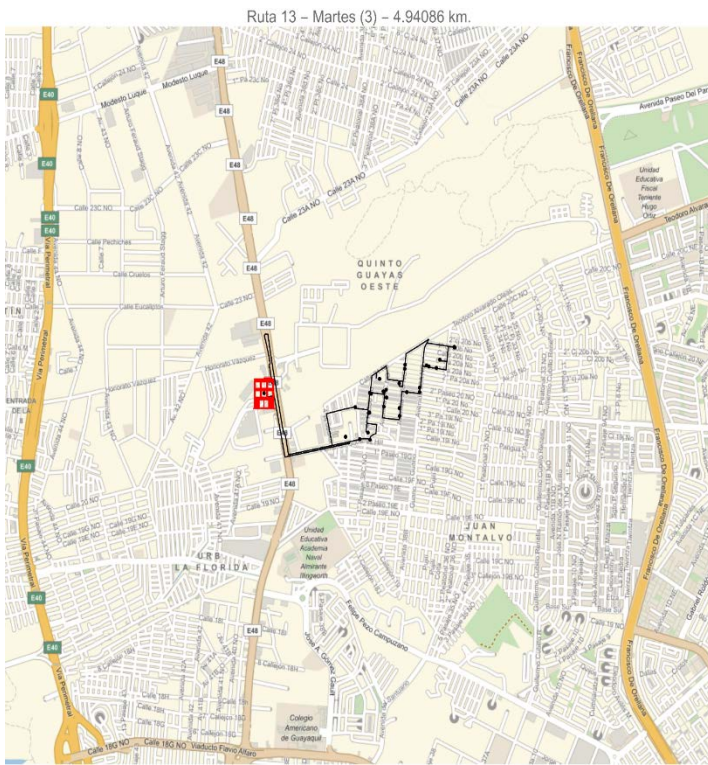
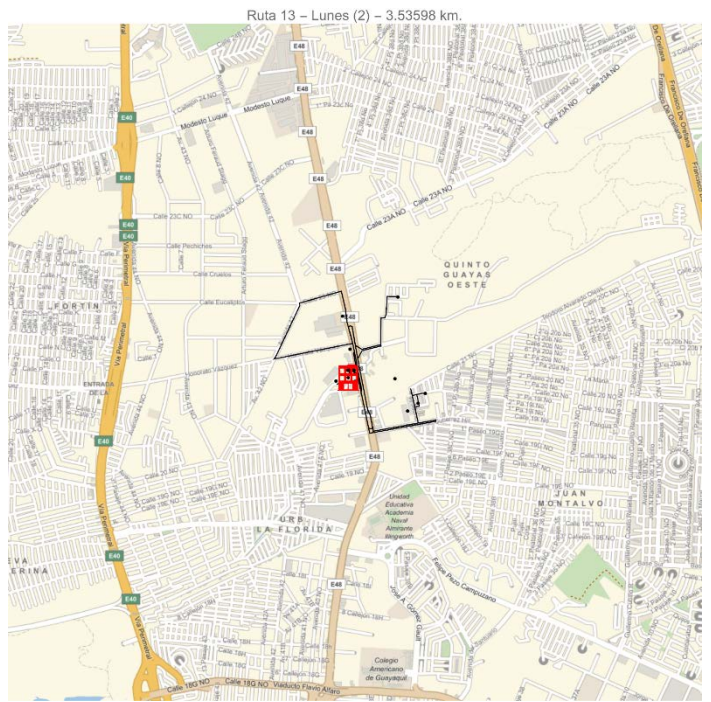
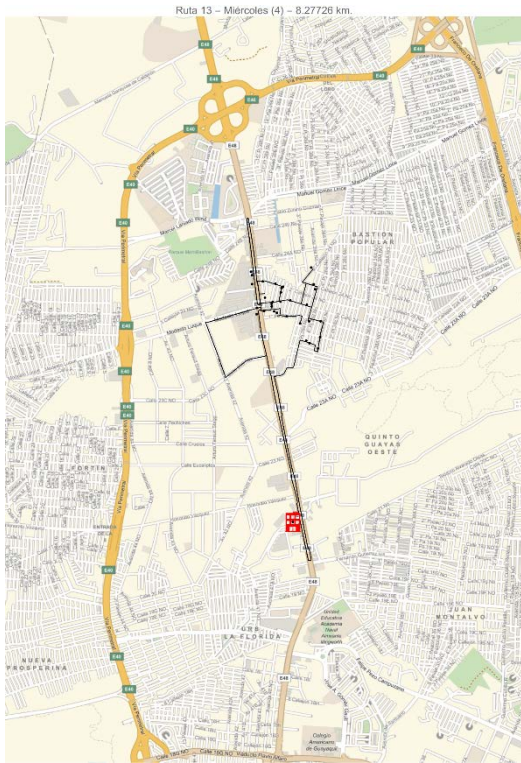


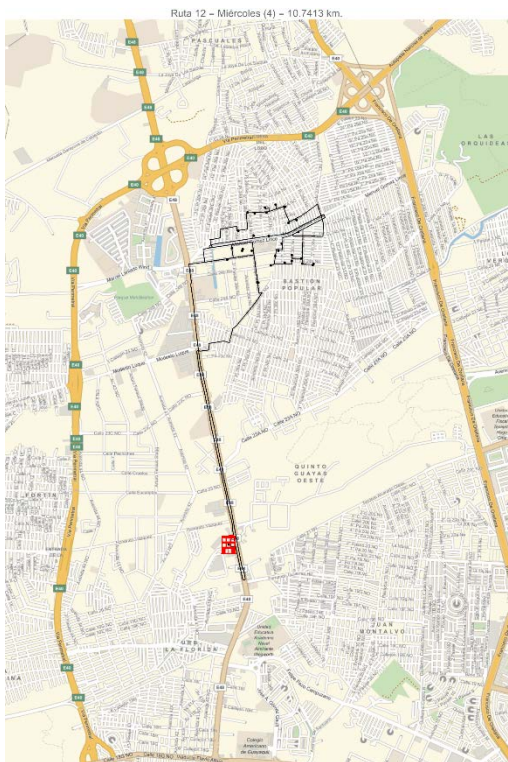
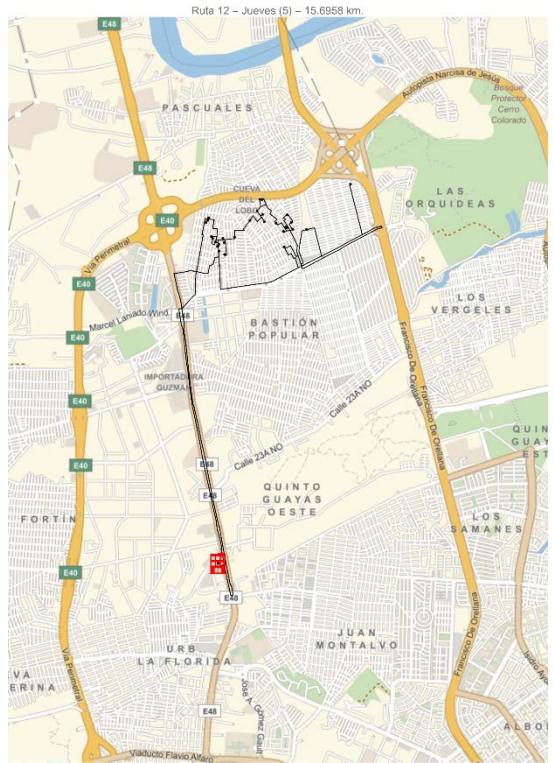
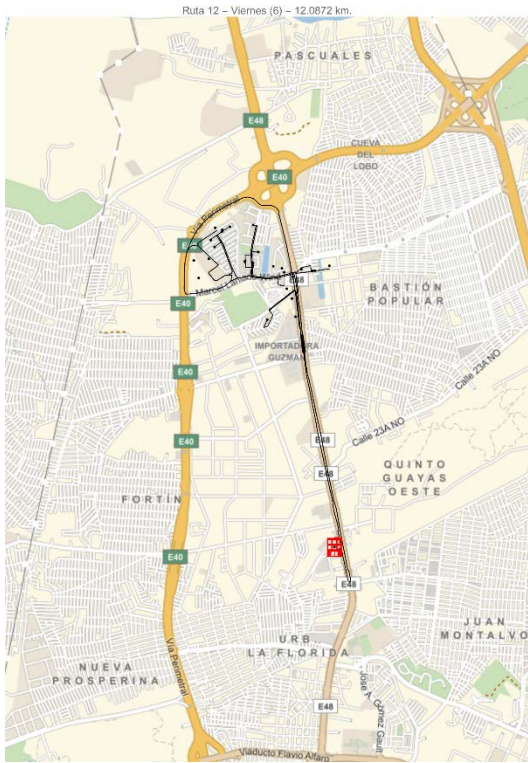
Ruta 14 – Martes (3) – 84.3024 km.











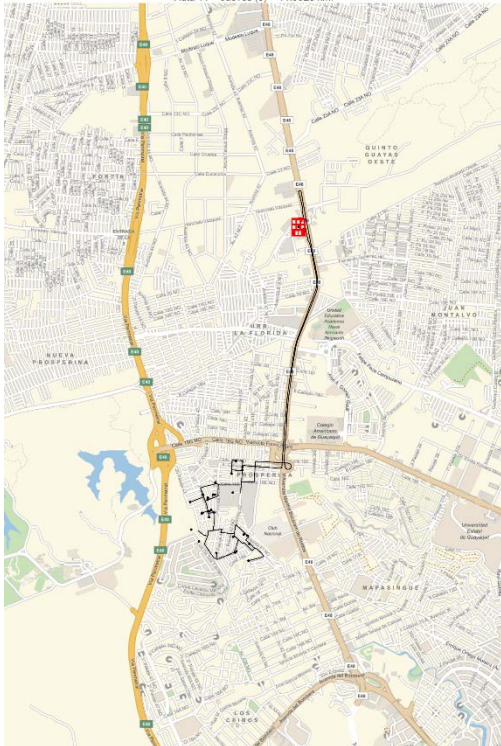
Ruta 12 – Lunes (2) – 15.3598 km.



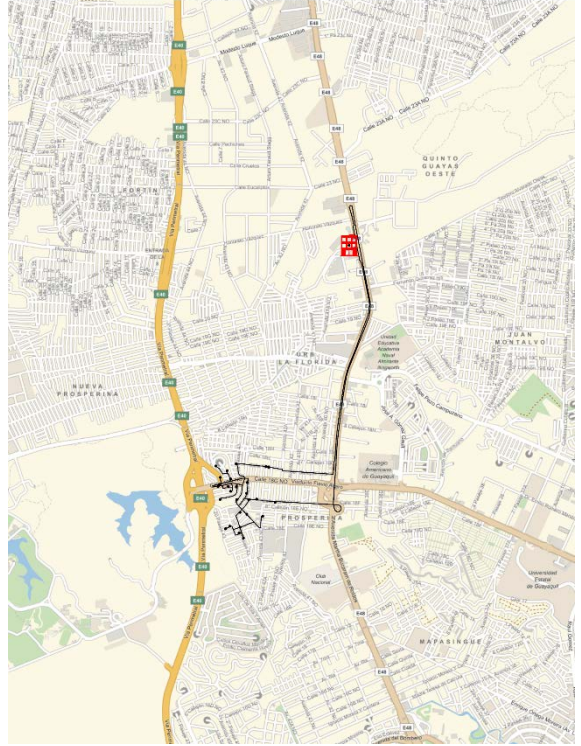
Ruta 11 – Sábado (7) – 13.9537 km.



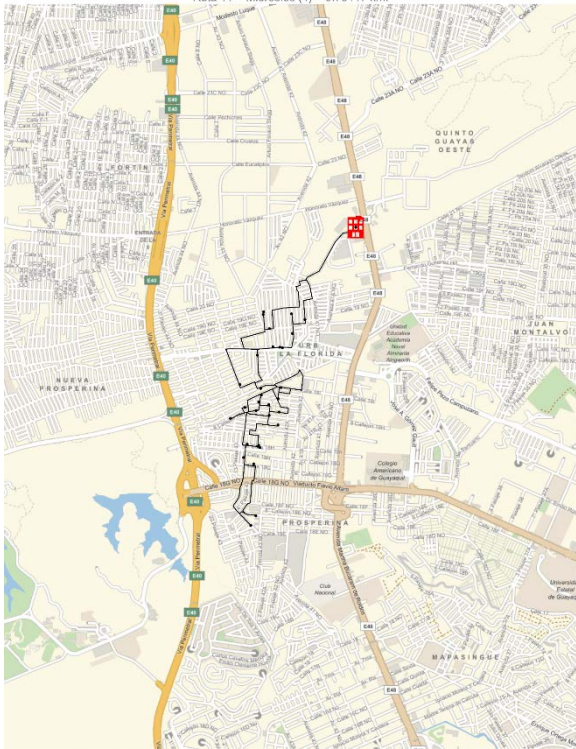
Ruta 11 – Jueves (6) – 11.9926 km.



Ruta 11 – Viernes (6) – 10.7283 km.



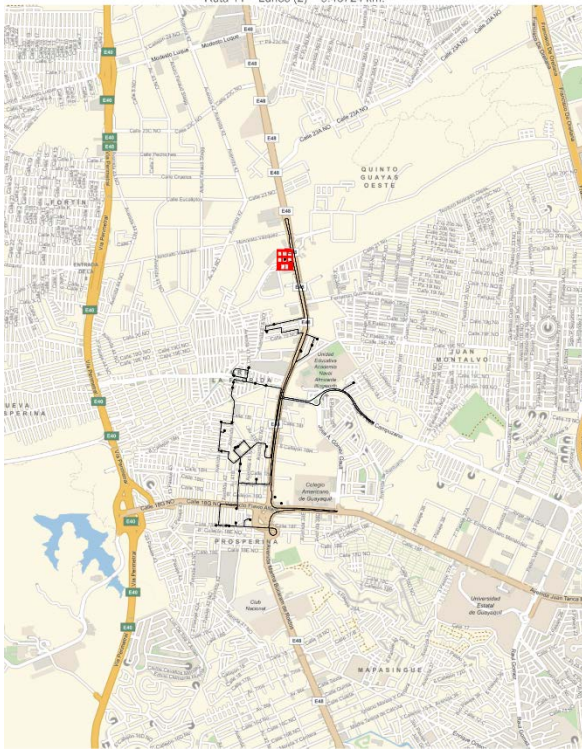
Ruta 11 - Miércoles (4) - 9.79117 km.



Ruta 11 - Martes (3) - 11.6842 km.



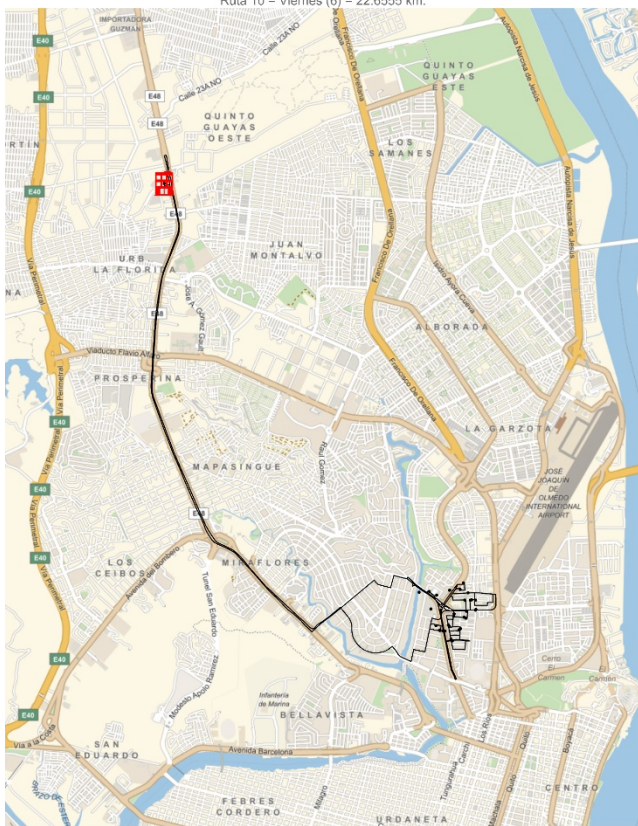
Ruta 11 - Lunes (2) - 9.46724 km.



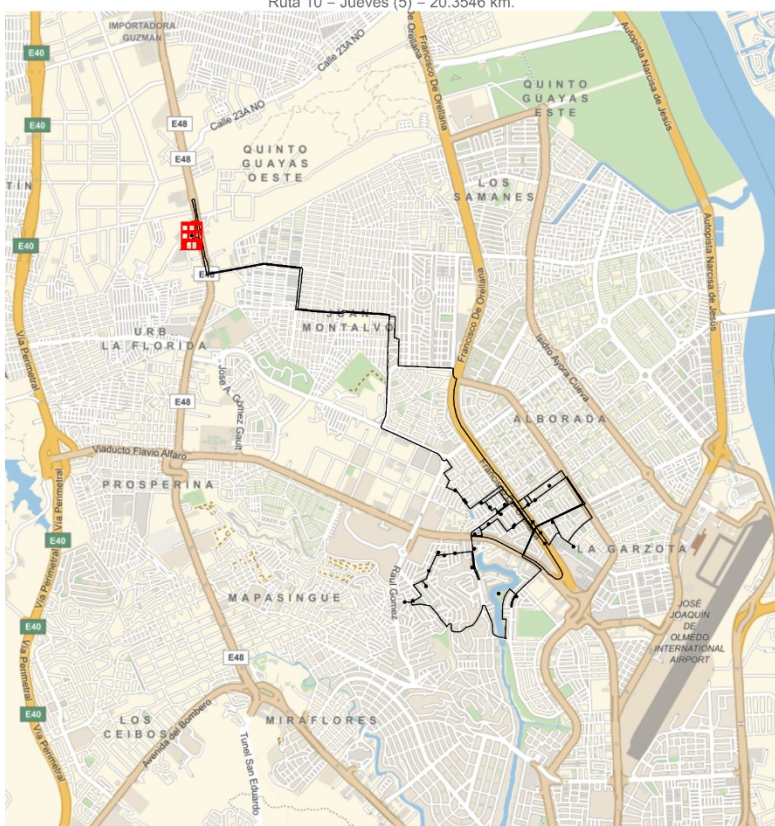
Ruta 10 - Sábado (7) - 21.1935 km.



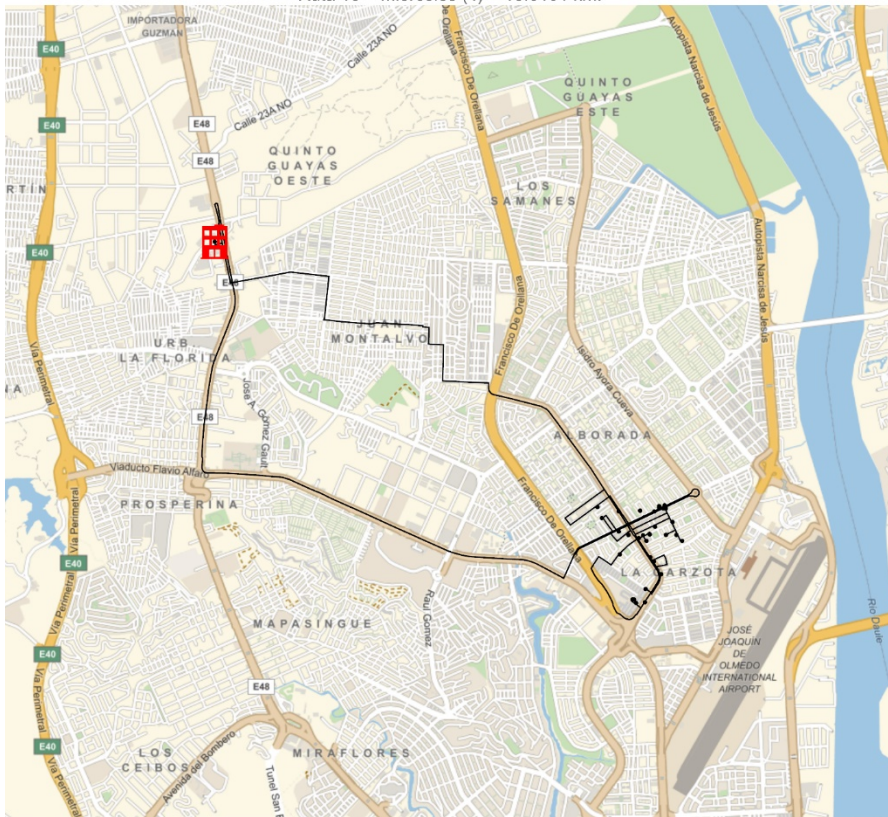
Ruta 10 - Viernes (6) - 22.6555 km.



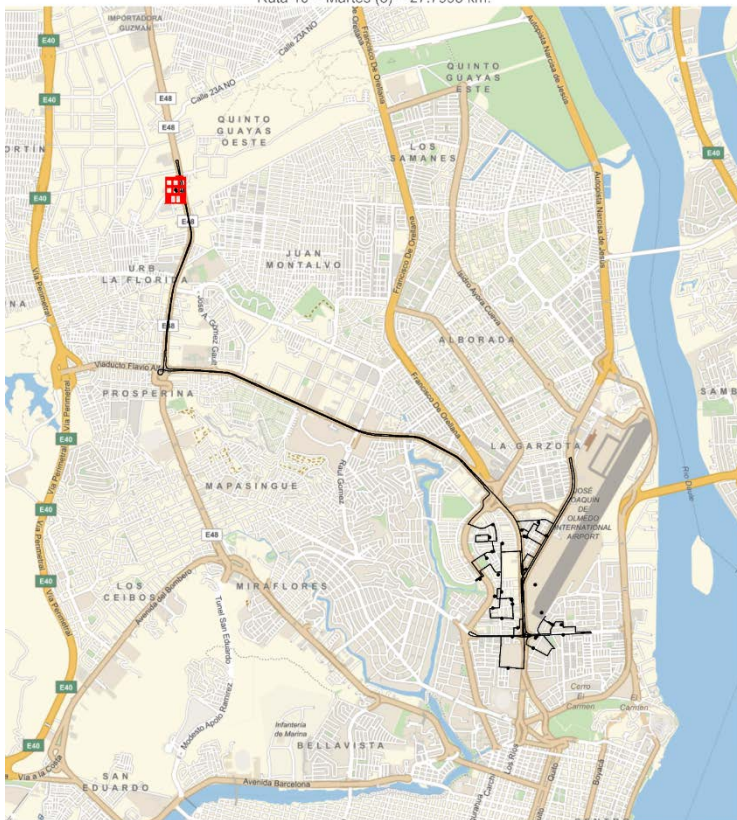
Ruta 10 - Jueves (5) - 20.3546 km.



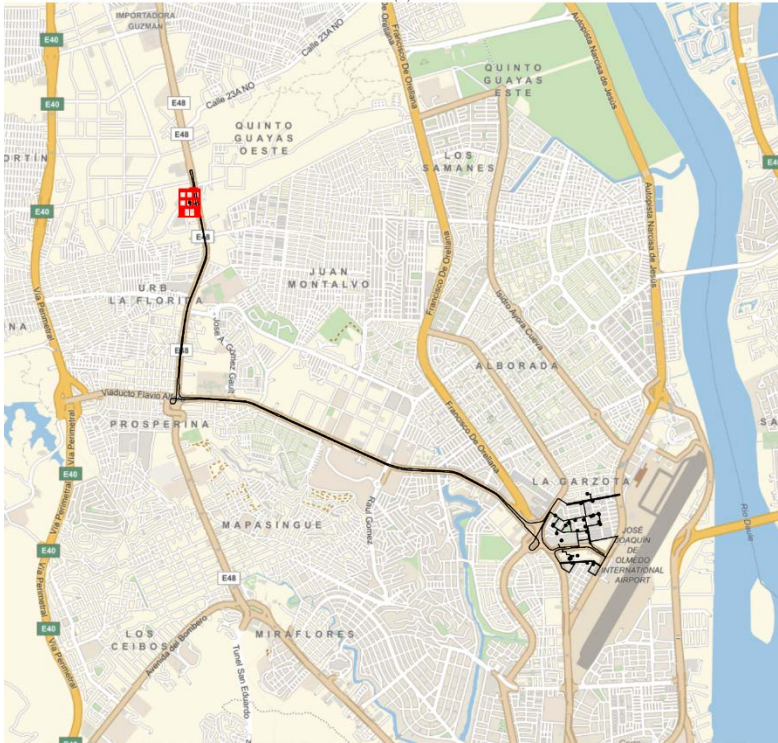
Ruta 10 – Miércoles (4) – 18.6101 km.



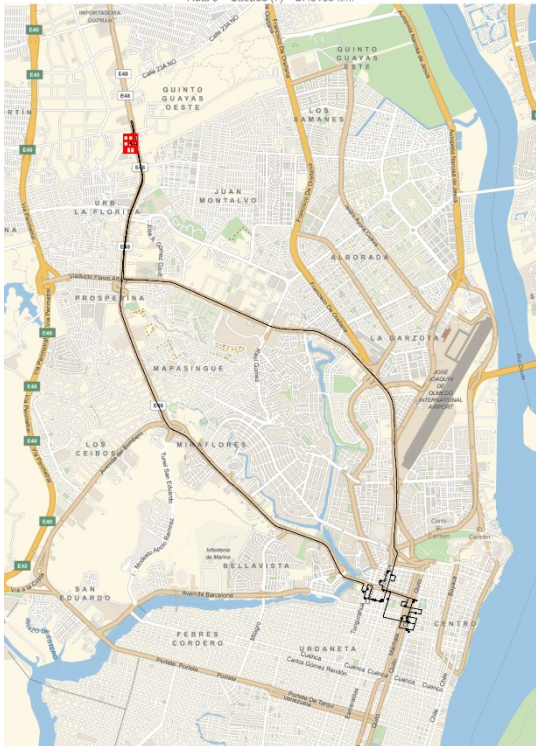
Ruta 10 – Martes (3) – 27.7995 km.



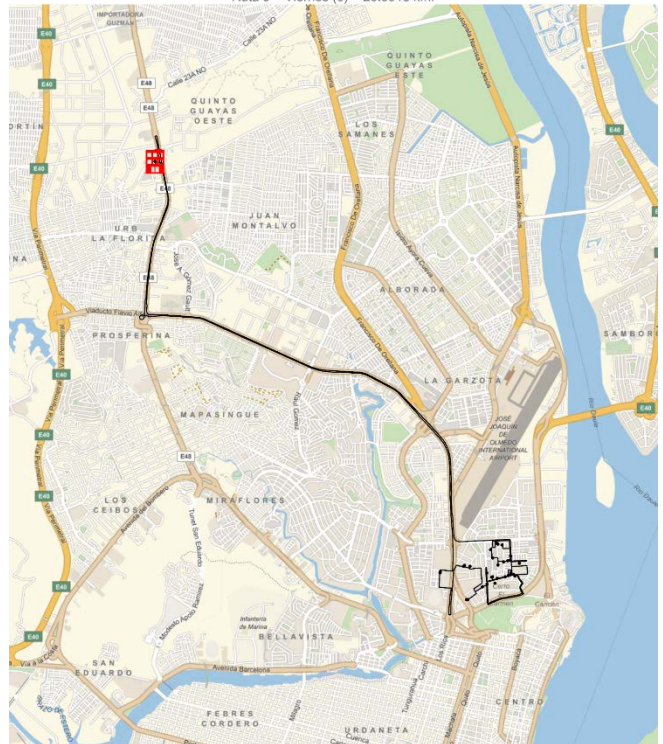
Ruta 10 – Lunes (2) – 19.8442 km.



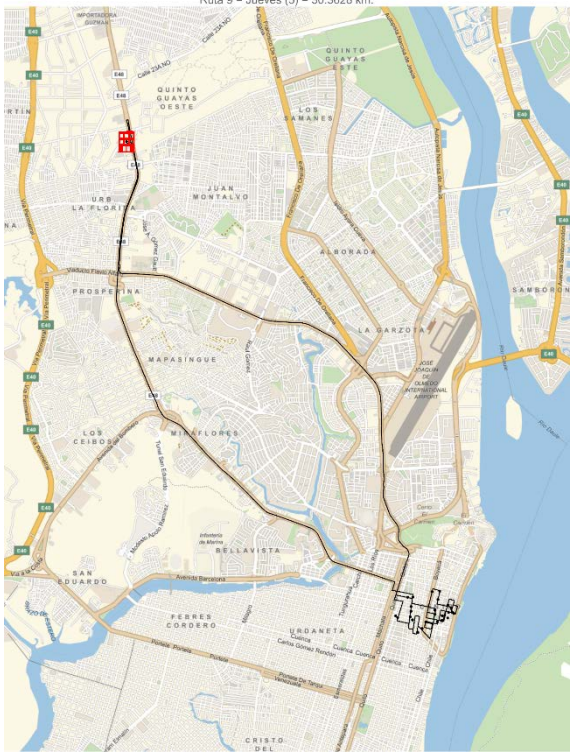
Ruta 9 – Sábado (7) – 27.8139 km.



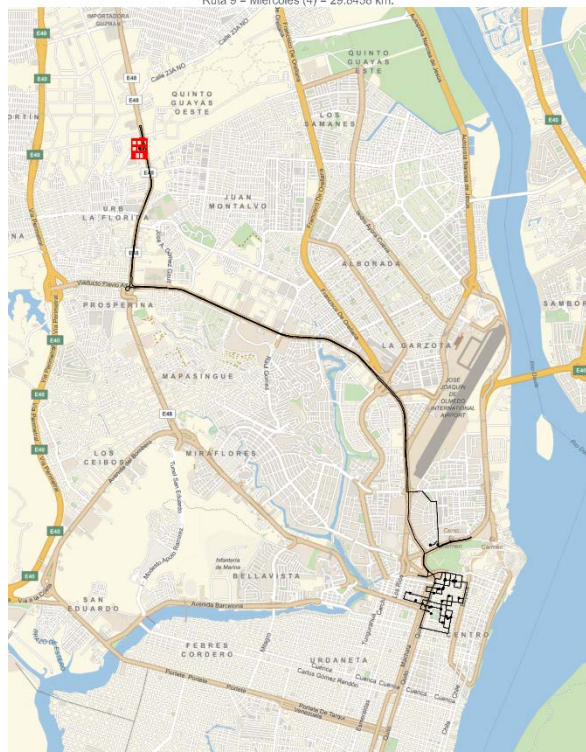
Ruta 9 – Viernes (6) – 26.5913 km.



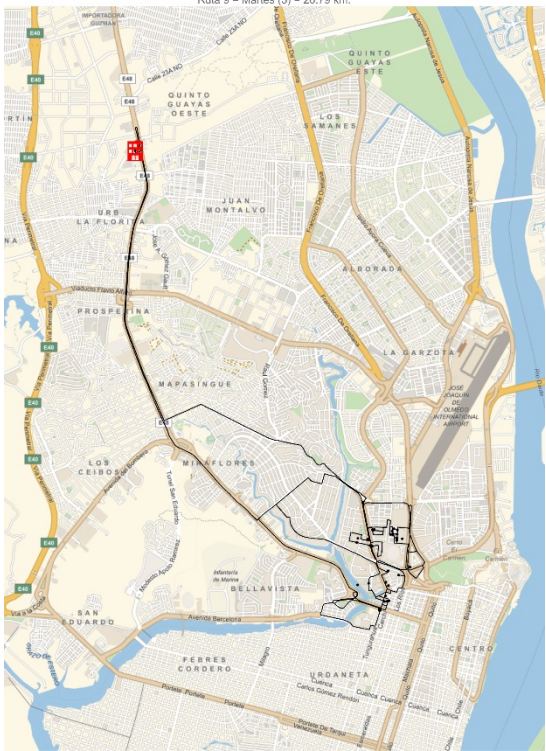
Ruta 9 – Jueves (5) – 30.3628 km.



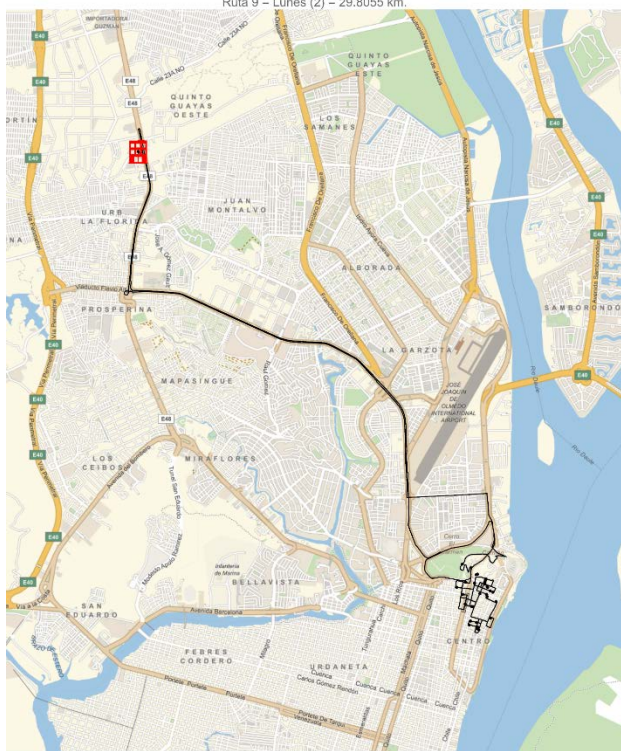
Ruta 9 – Miércoles (4) – 29.8458 km.



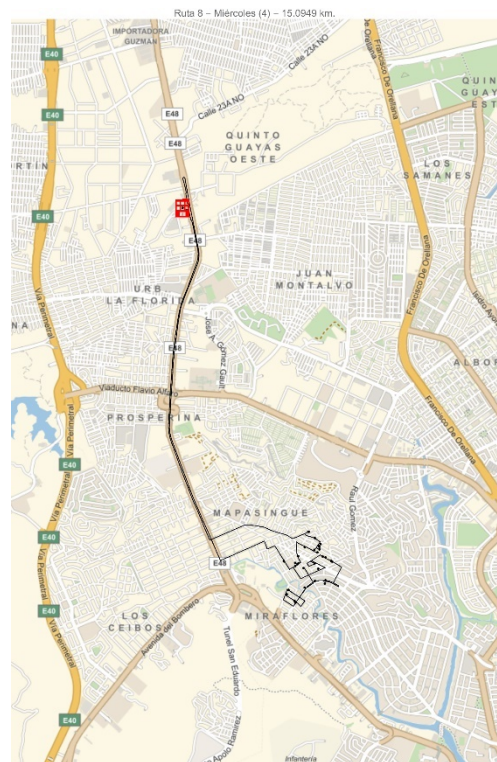
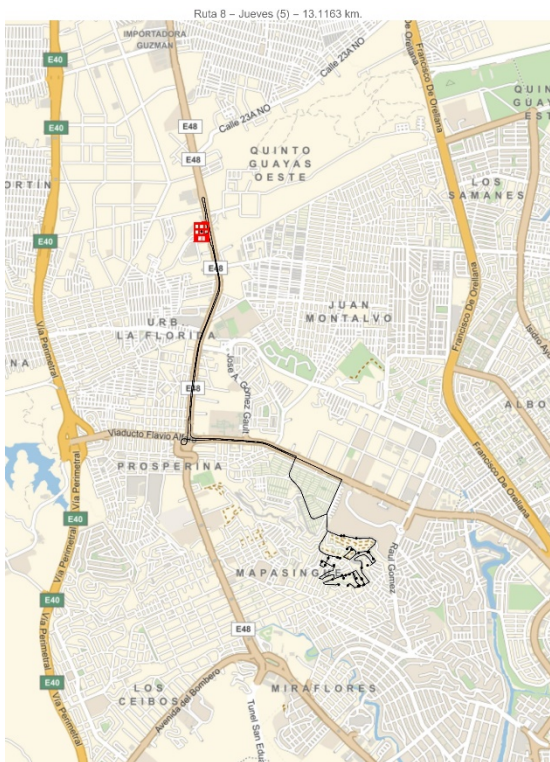
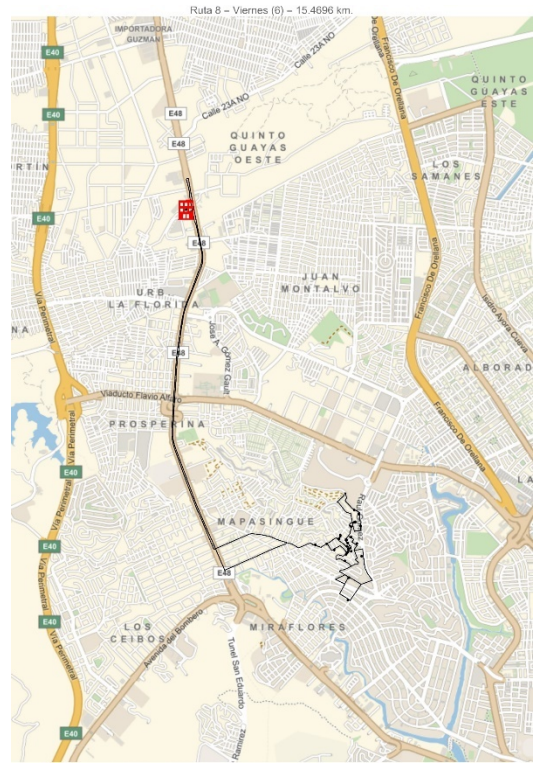
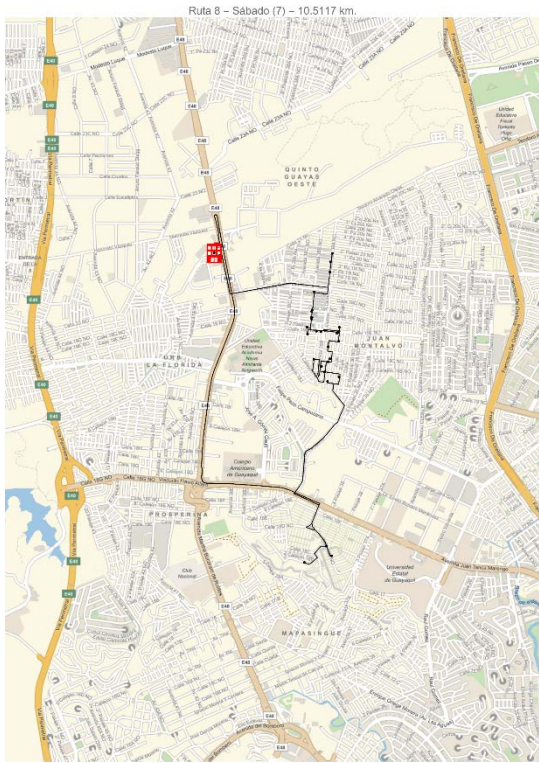
Ruta 9 – Martes (3) – 26.79 km.

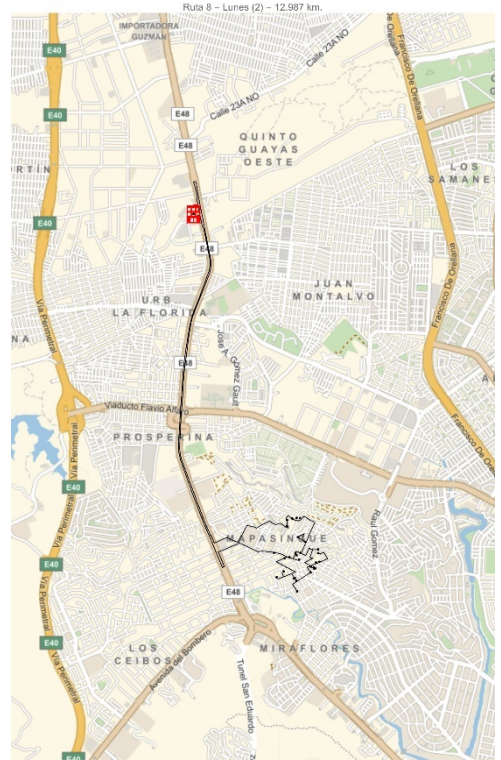
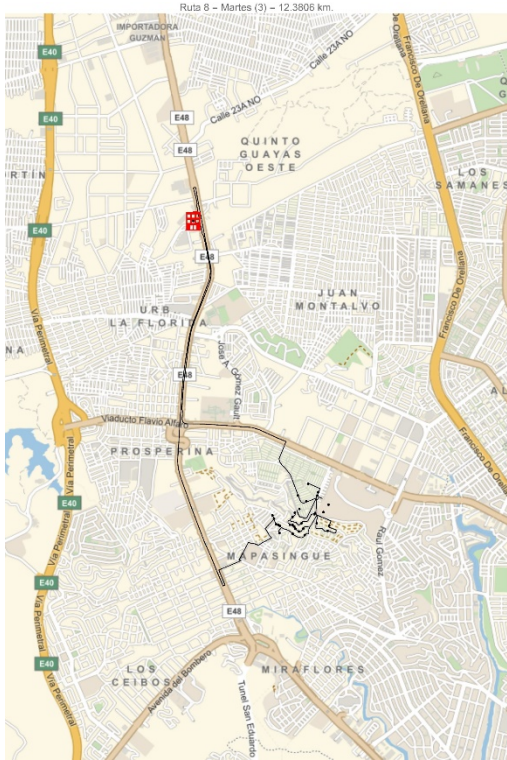


Ruta 9 – Lunes (2) – 29.8055 km.

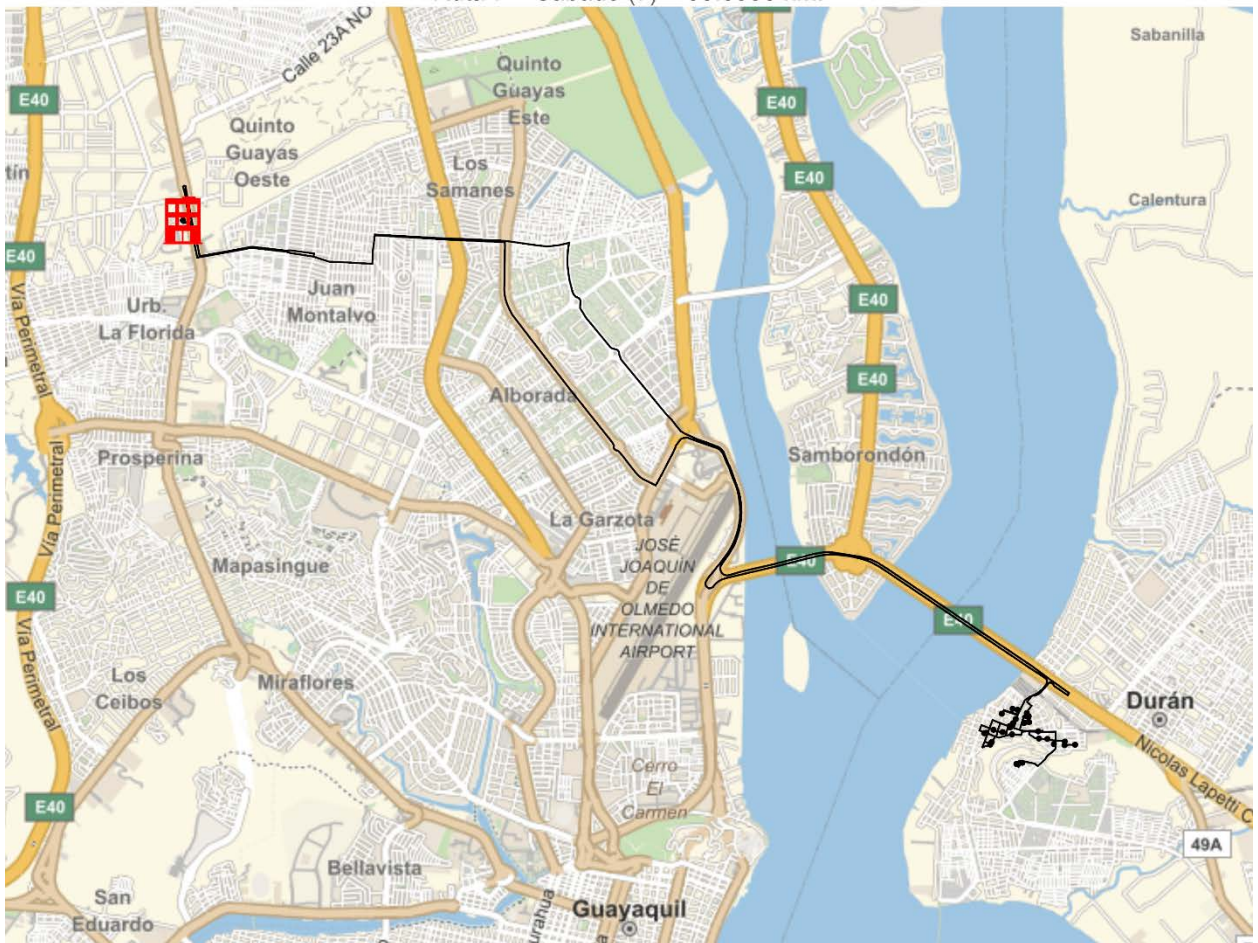




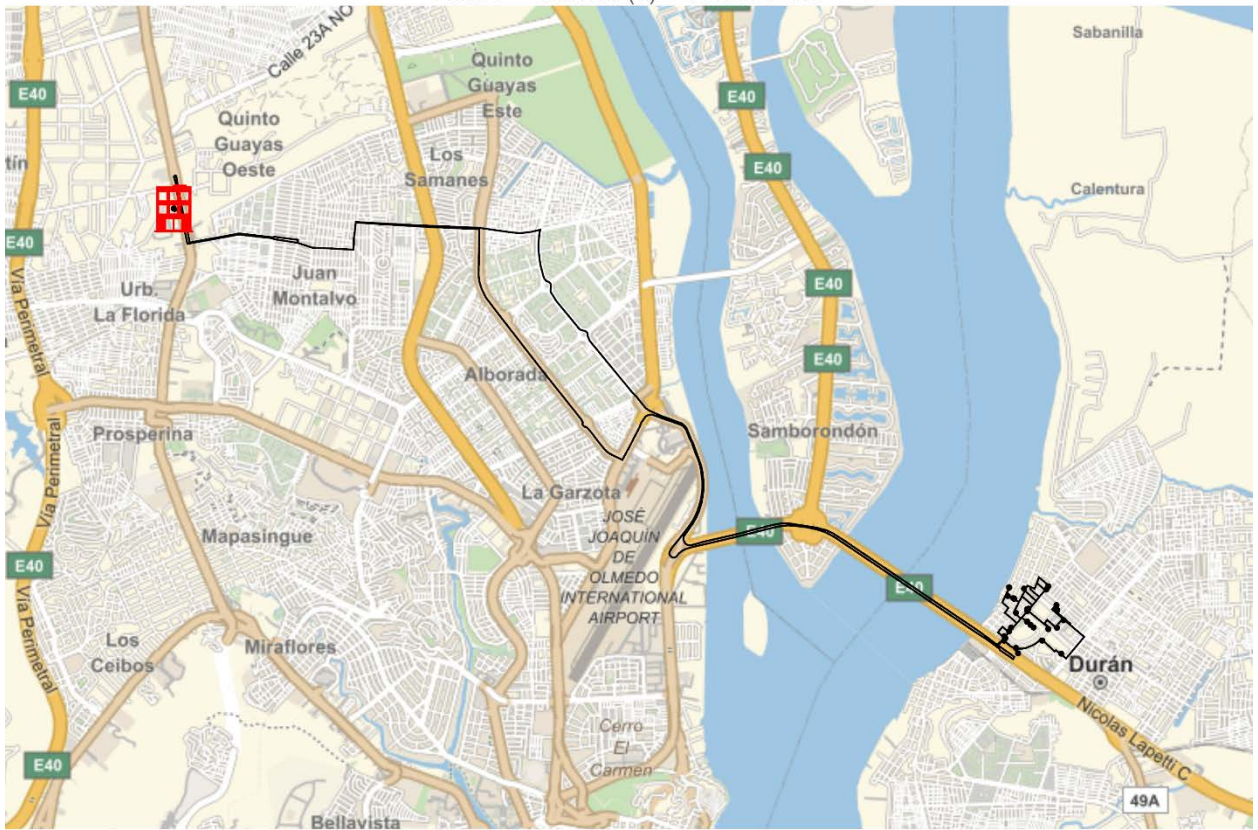




Ruta 7 - Sábado (7) - 33.5998 km.



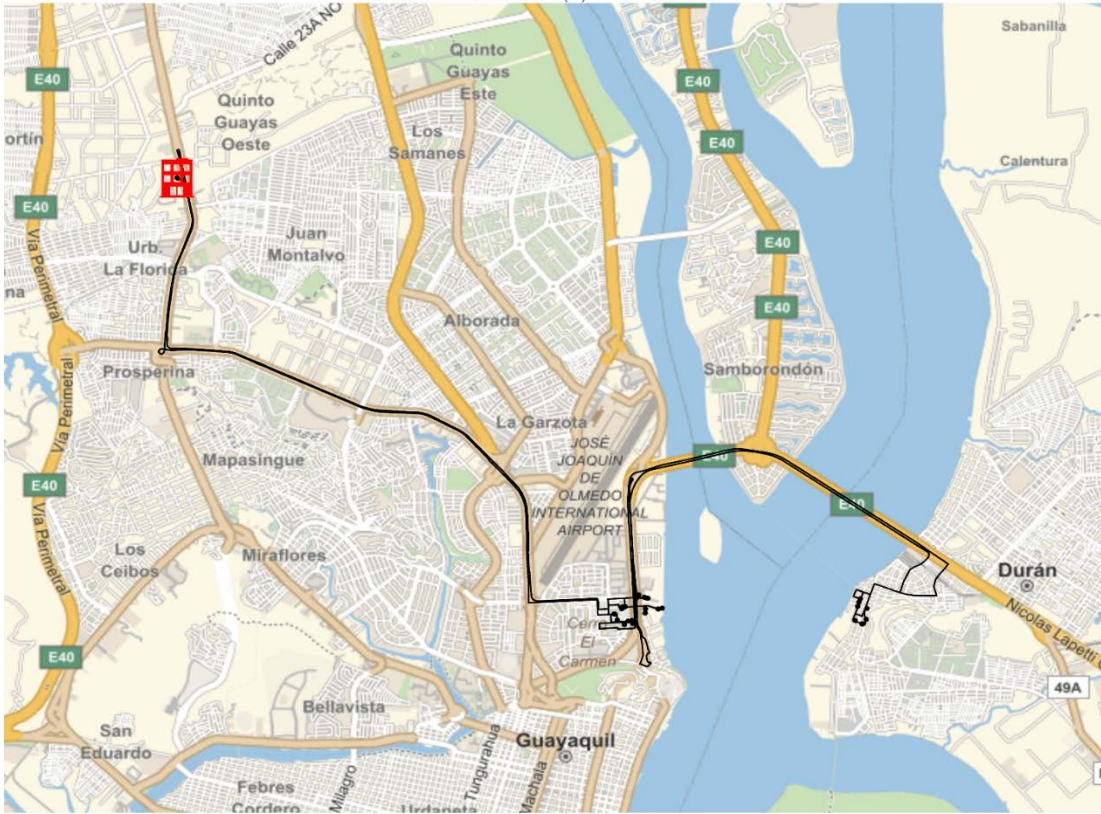
Ruta 7 – Viernes (6) – 33.8675 km.



Ruta 7 – Jueves (5) – 32.972 km.



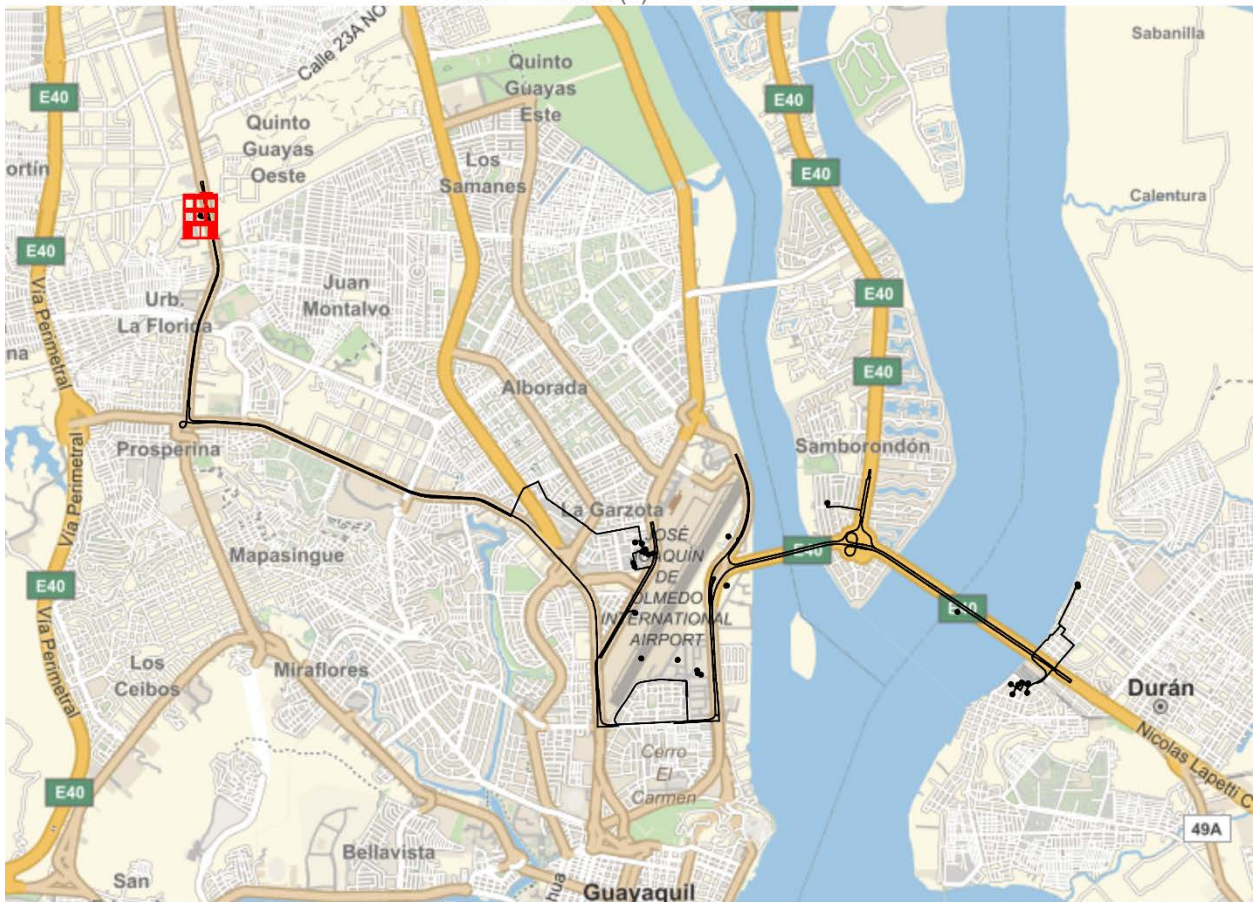
Ruta 7 – Miércoles (4) – 31.1904 km.



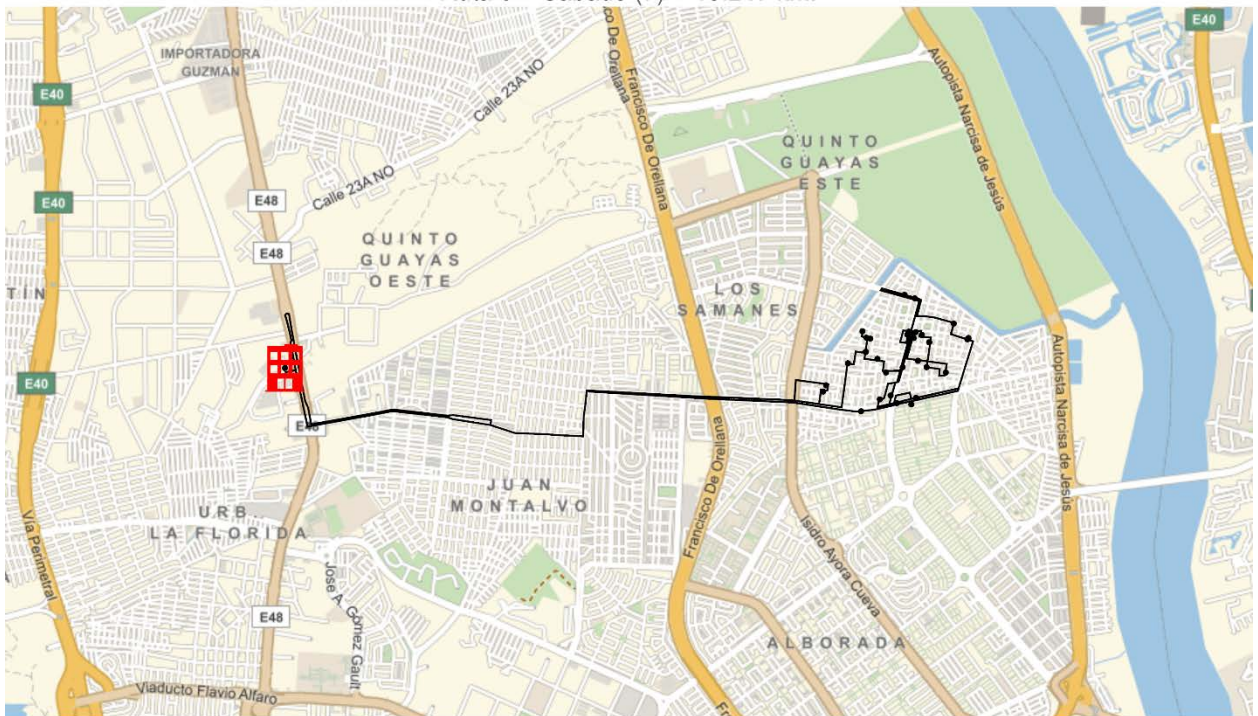
Ruta 7 – Martes (3) – 32.3985 km.



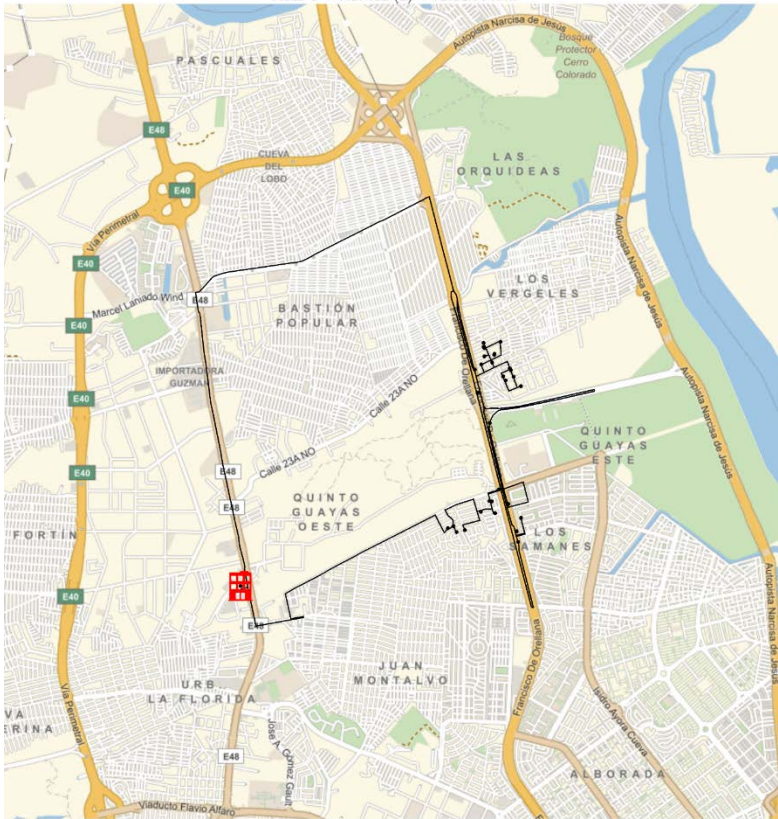
Ruta 7 – Lunes (2) – 34.6898 km.



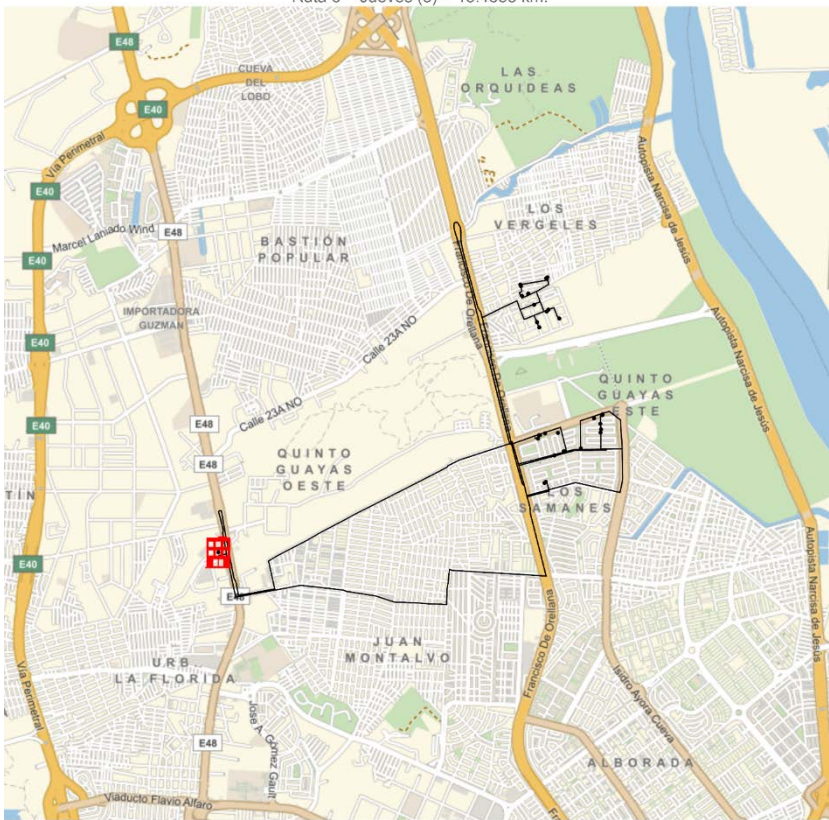
Ruta 6 – Sábado (7) – 13.247 km.



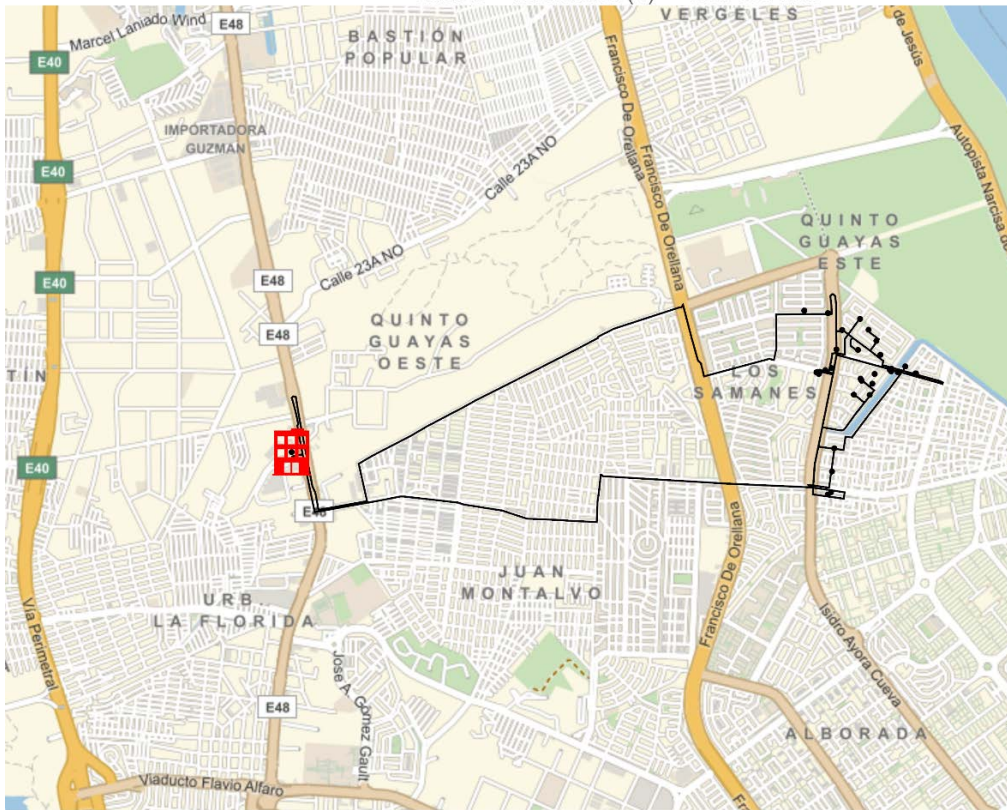
Ruta 6 – Viernes (6) – 13.0941 km.



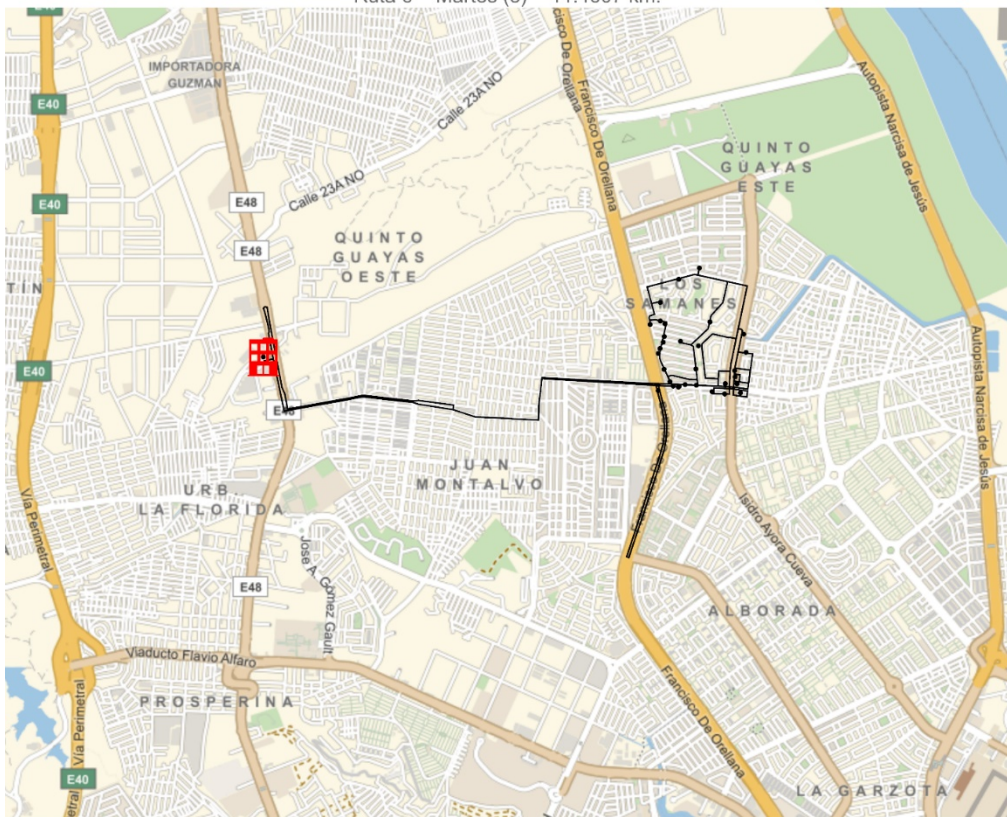
Ruta 6 – Jueves (5) – 13.4355 km.



Ruta 6 – Miércoles (4) – 13.0192 km.



Ruta 6 – Martes (3) – 11.4607 km.



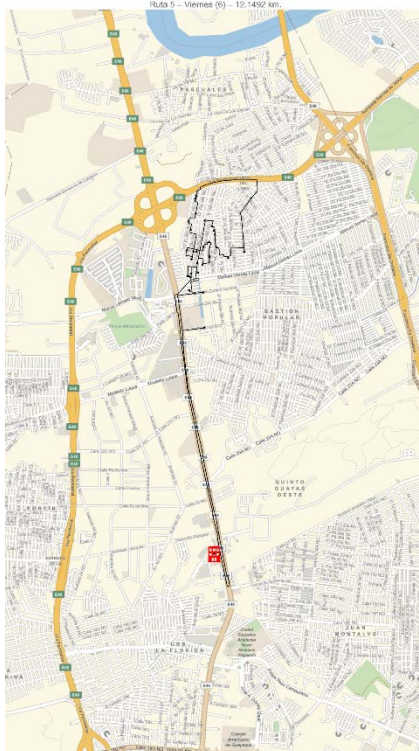
Ruta 6 – Lunes (2) – 8.49452 km.



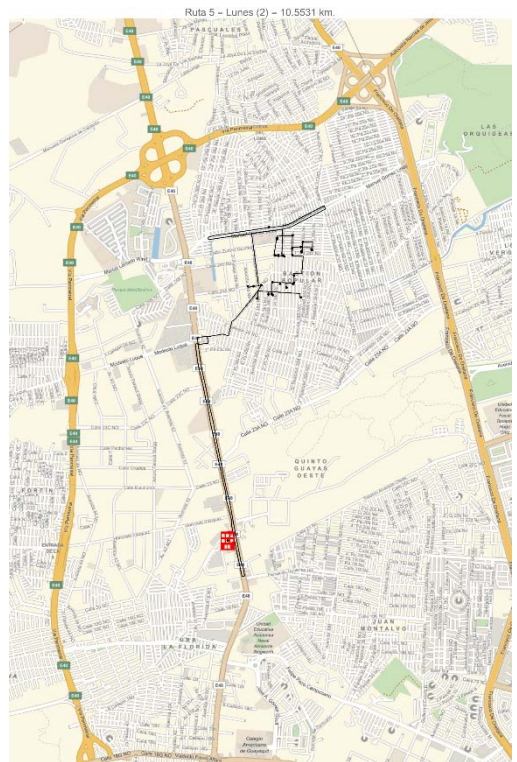
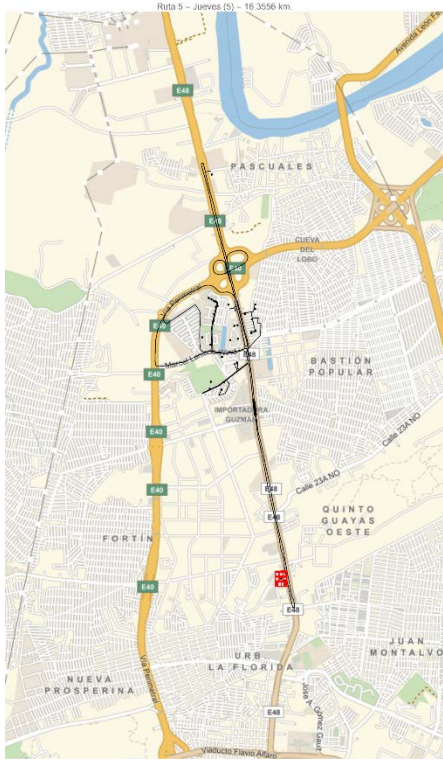
Ruta 5 – Sábado (7) – 27.9619 km.



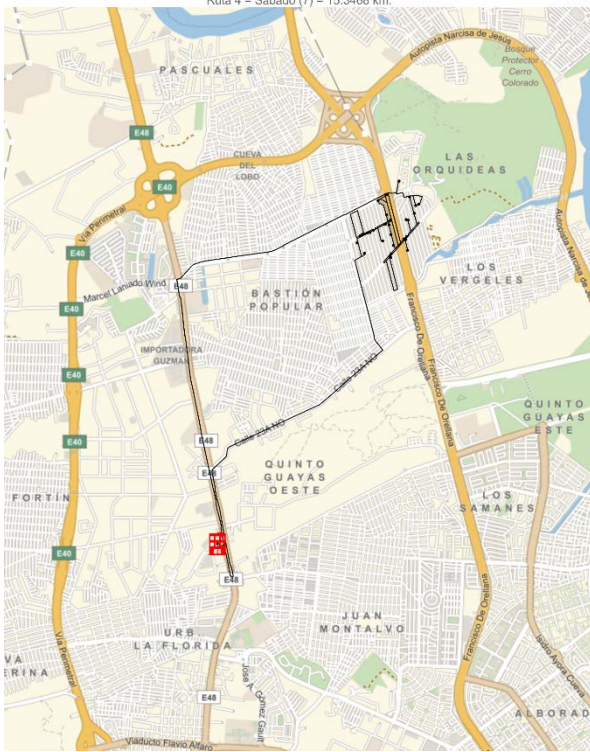
Ruta 5 – Viernes (6) – 12.1492 km.



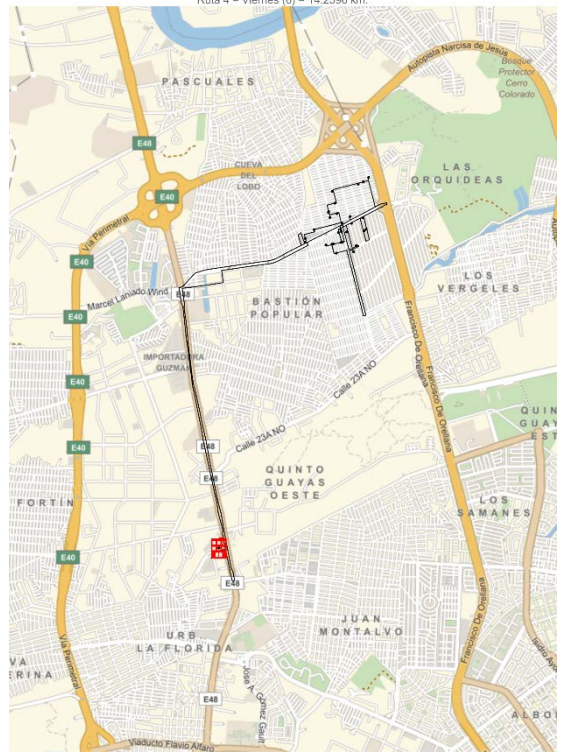




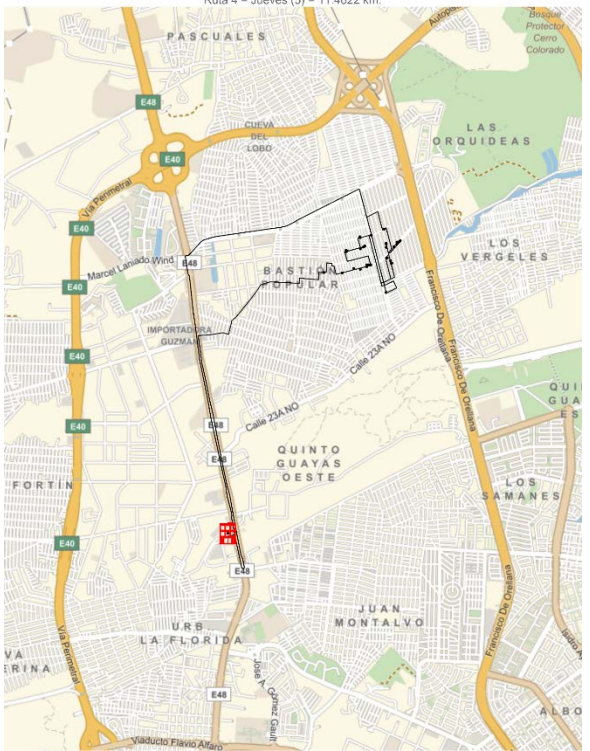
Ruta 4 – Sábado (7) – 15.3468 km.



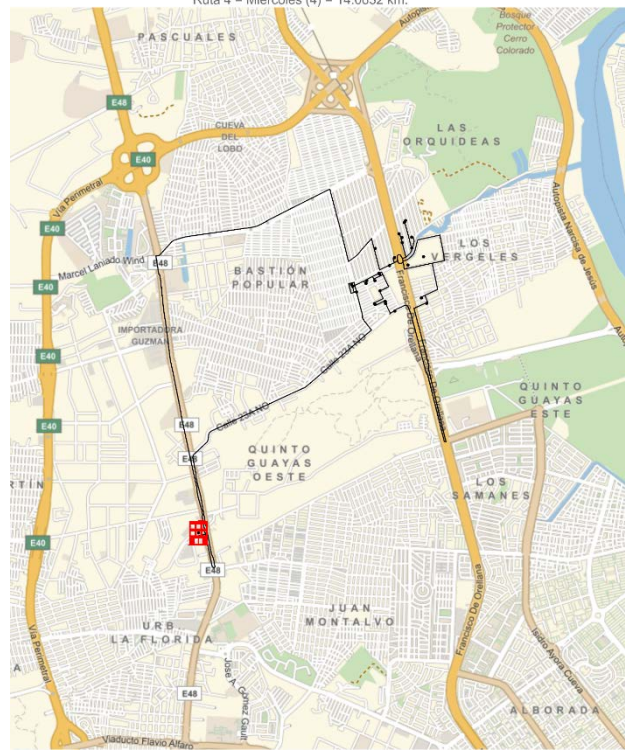
Ruta 4 – Viernes (6) – 14.2596 km.



Ruta 4 – Jueves (5) – 11.4622 km.

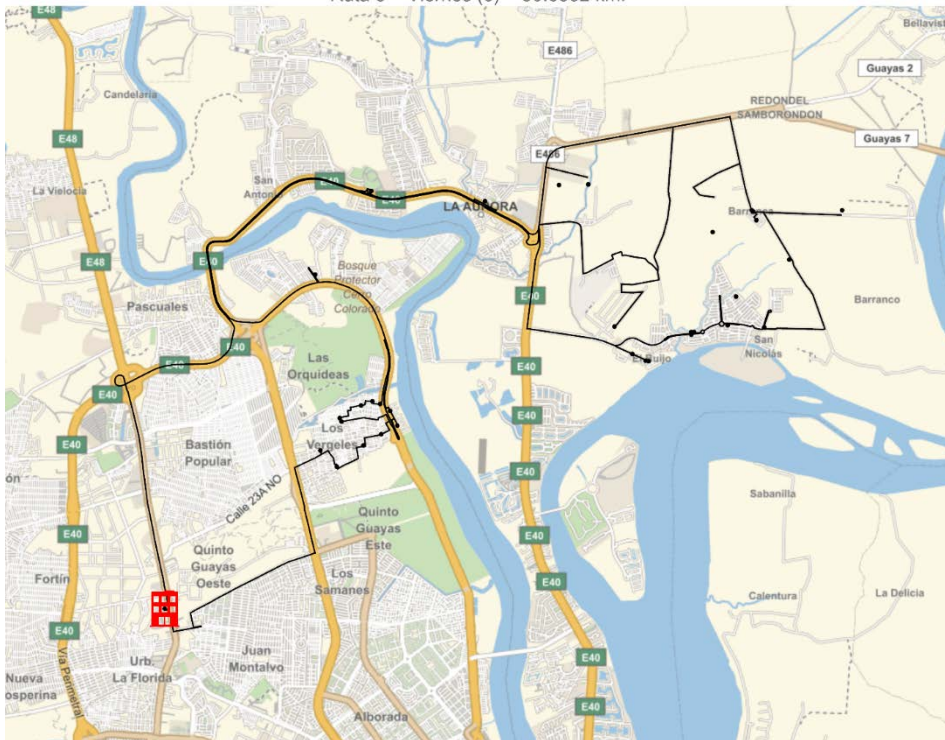


Ruta 4 – Miércoles (4) – 14.0632 km.

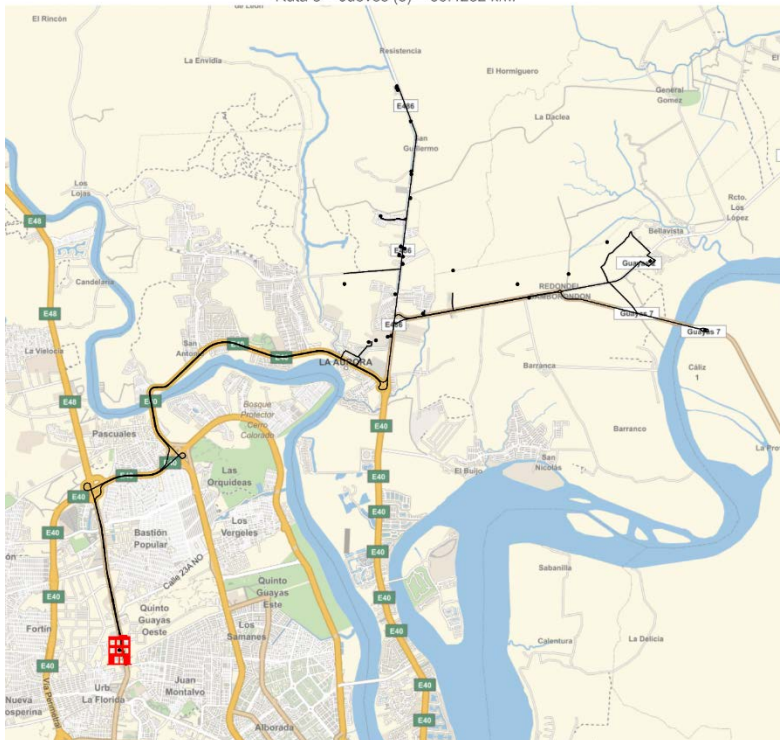




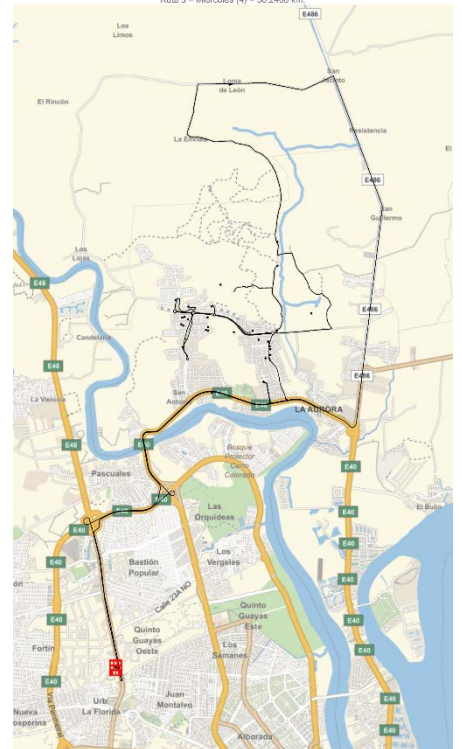
Ruta 3 – Viernes (6) – 50.6332 km.



Ruta 3 – Jueves (5) – 66.4232 km.



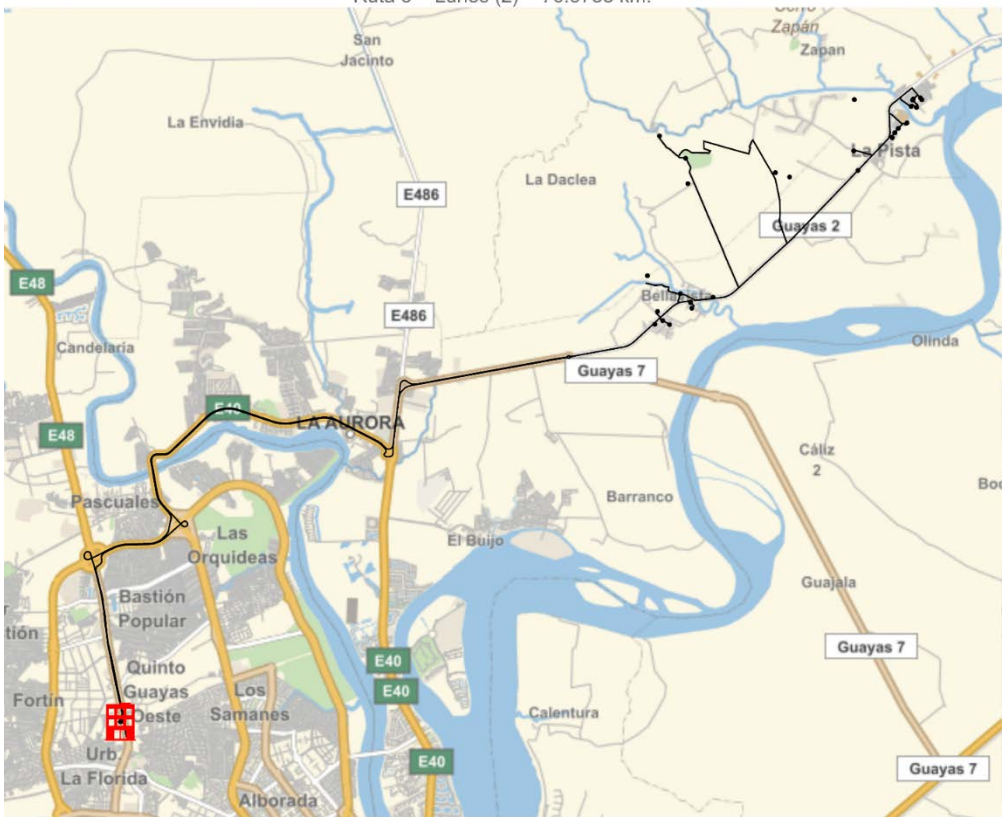
Ruta 3 – Miércoles (4) – 56.2466 km.



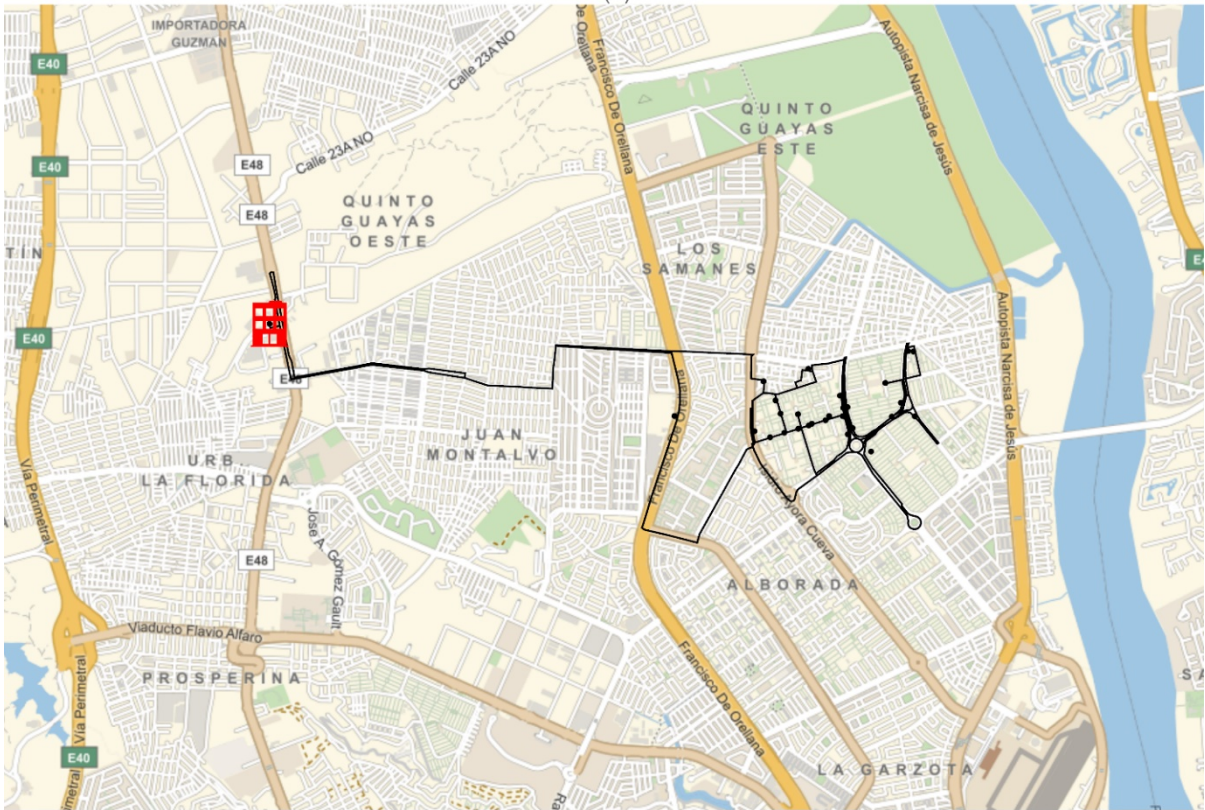
Ruta 3 - Martes (3) - 85.8353 km.



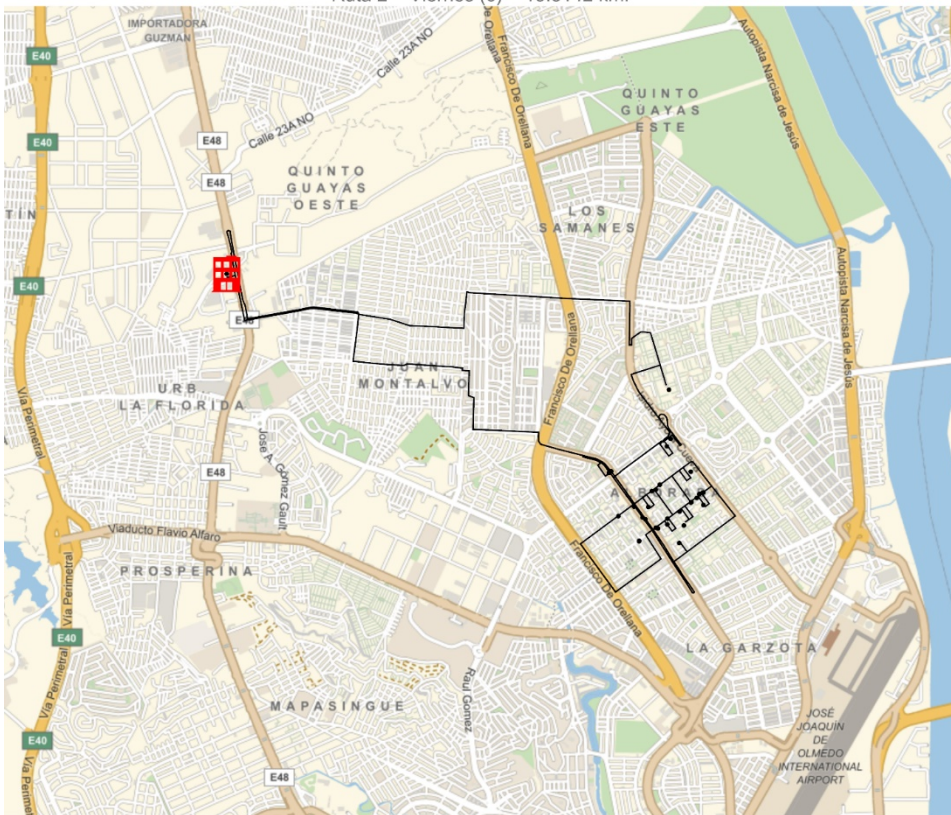
Ruta 3 - Lunes (2) - 76.3733 km.



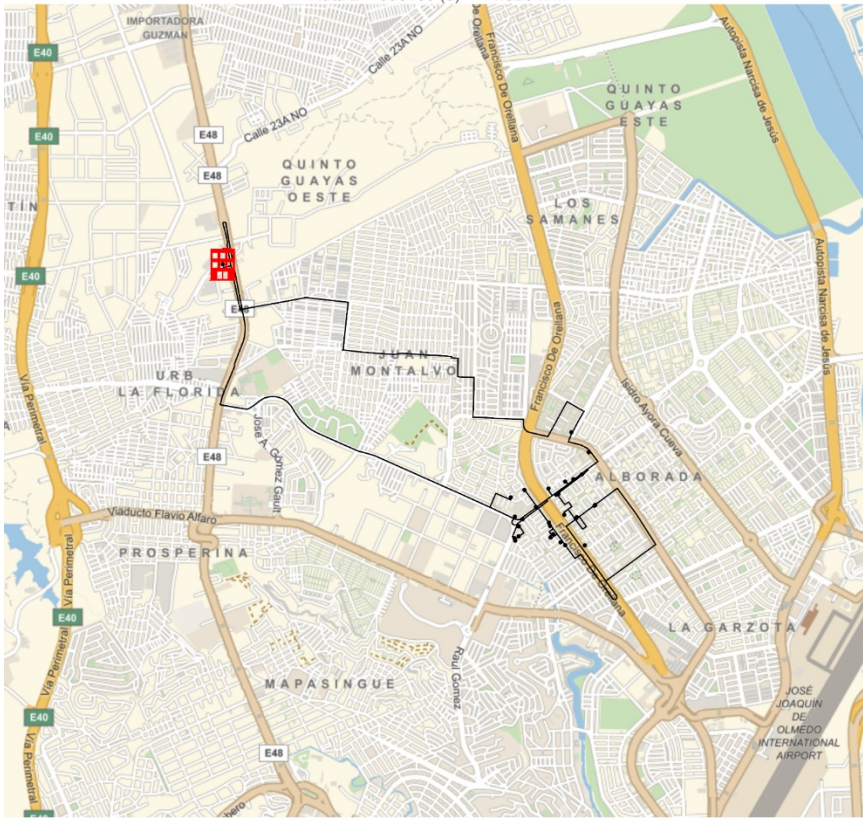
Ruta 2 – Sábado (7) – 13.8167 km.



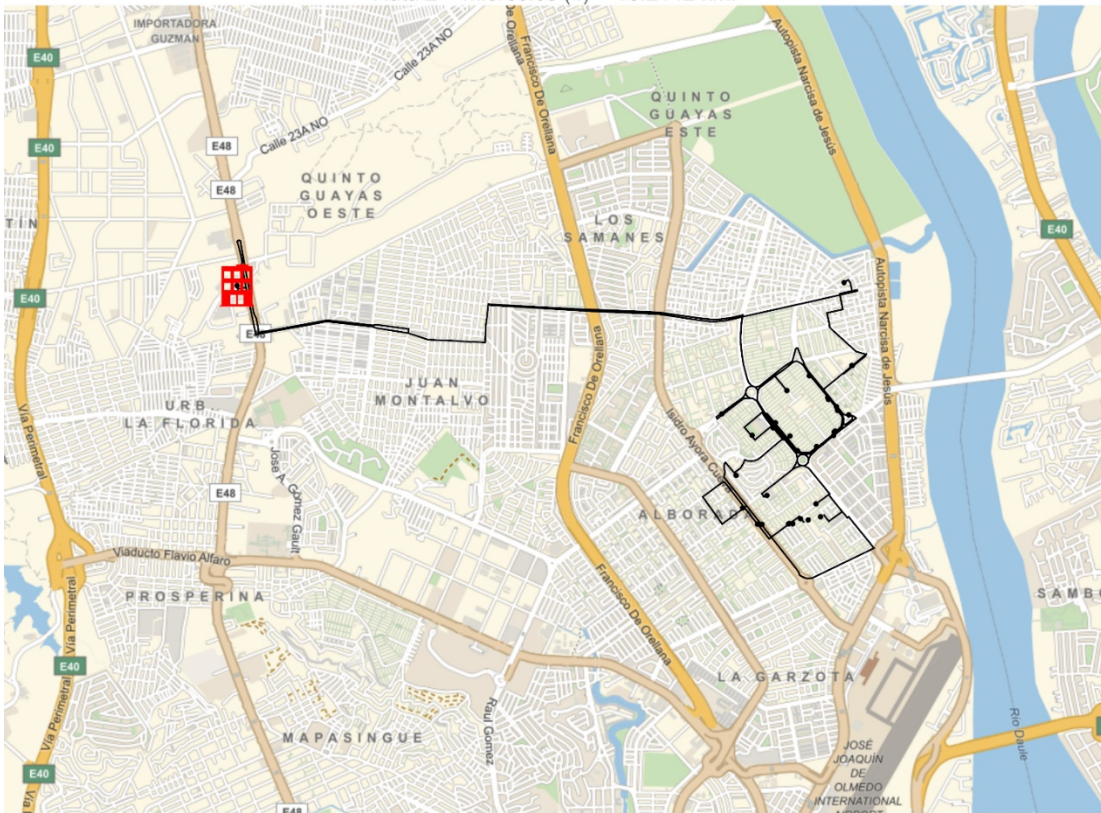
Ruta 2 – Viernes (6) – 15.8142 km.



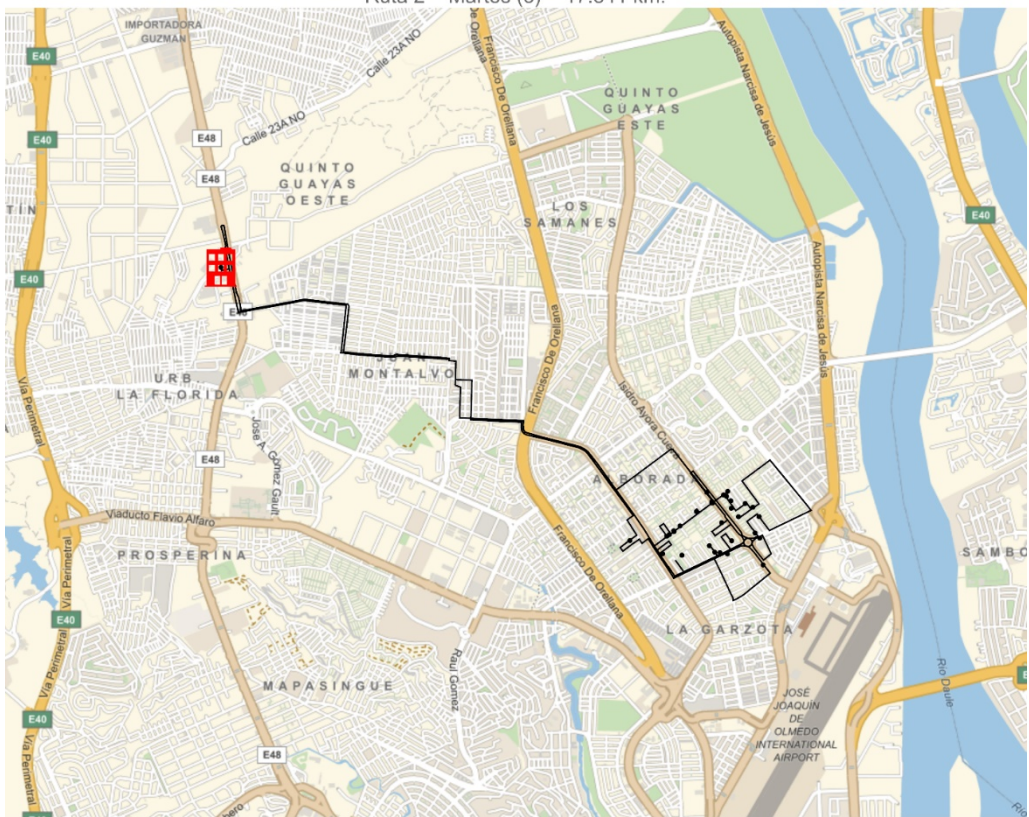
Ruta 2 – Jueves (5) – 14.326 km.



Ruta 2 – Miércoles (4) – 18.2142 km.



Ruta 2 – Martes (3) – 17.841 km.

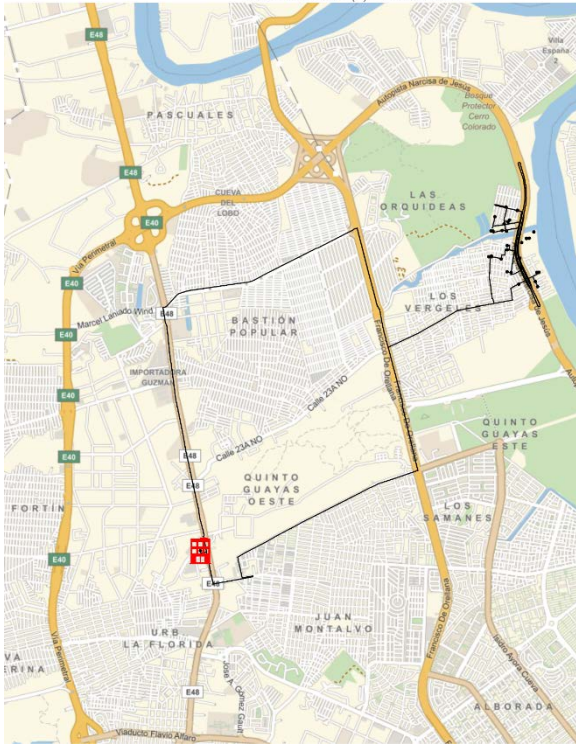


Ruta 2 – Lunes (2) – 20.627 km.





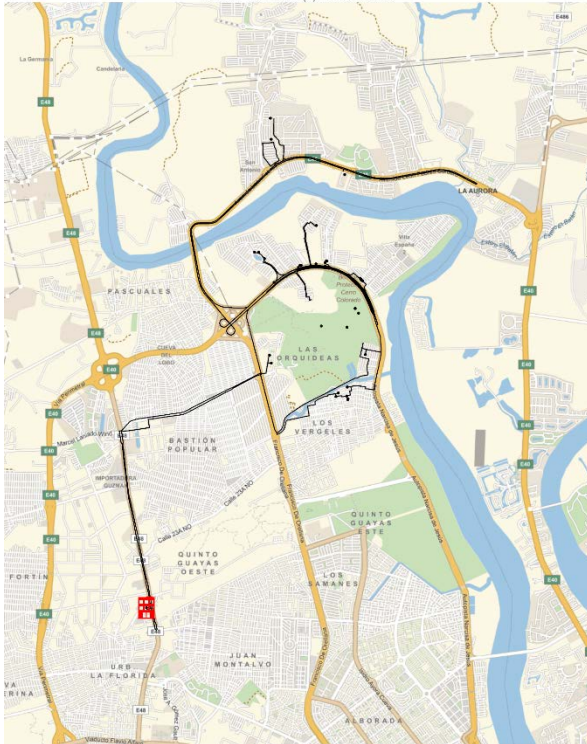
Ruta 1 – Sábado (7) – 19.2484 km.



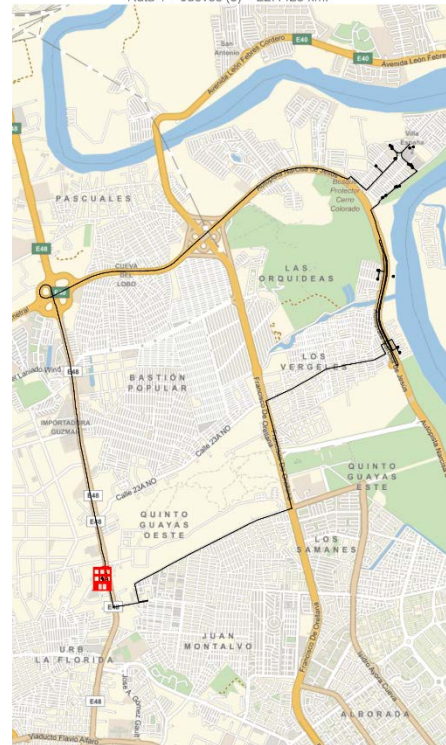
Ruta 1 – Viernes (6) – 17.6526 km.



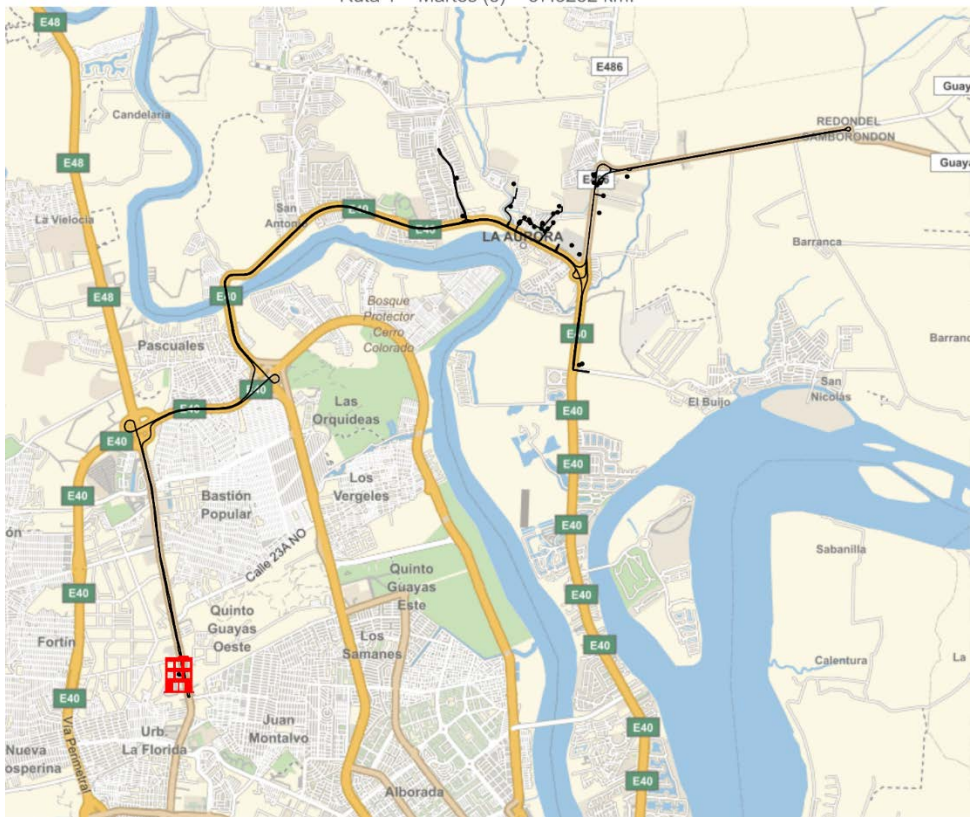
Ruta 1 – Miércoles (4) – 29.0675 km.



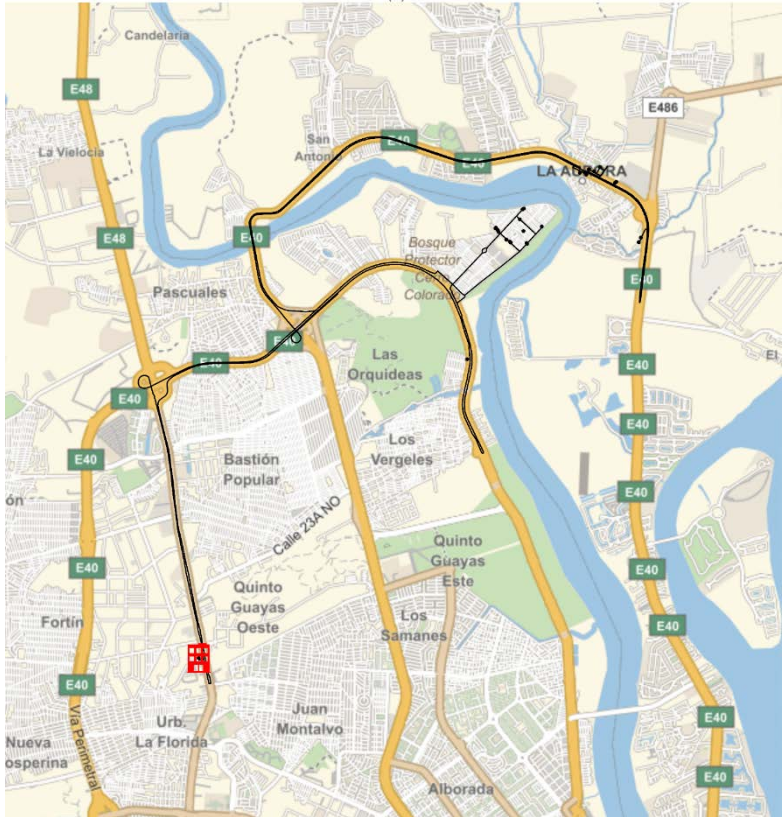
Ruta 1 – Jueves (5) – 22.4423 km.



Ruta 1 – Martes (3) – 37.3252 km.



Ruta 1 – Lunes (2) – 29.2813 km.



# APÉNDICE D

## Código de Wolfram Mathematica

### OPT DE RUTAS

#### LECTURA DE DATOS

```
In[184]:= SetDirectory[NotebookDirectory[]];
(*distancia base*) (*distancia de ciudades*)
DataGeneral = Import["TESIS.xls"];
(*ruta*)

In[196]:= RutasOriginal = DataGeneral[[2]];
DatosOriginal = RutasOriginal[[2 ;; 9540]];
InternalIDOriginal = DatosOriginal[[1 ;; 11]];
CustomerIDOriginal = DatosOriginal[[1 ;; 22]];
RutasOriginal = DatosOriginal[[1 ;; 33]];
DiasOriginal = DatosOriginal[[1 ;; 43]];
CoordenadasOriginal = DatosOriginal[[1 ;; 10, 11]];
SegmentacionOriginal = DatosOriginal[[1 ;; 12]];
nOriginal = Length[DatosOriginal];

FormaClientesOriginal = Table[{CustomerIDOriginal[[i]], RutasOriginal[[i]], DiasOriginal[[i]], CoordenadasOriginal[[i]], SegmentacionOriginal[[i]], {i, nOriginal}}];

In[204]:= CoordenadasBaseNestle = {(-2.118808325846659, -79.9328297543551)};

In[207]:= CoordenadasOriginales = CoordenadasOriginal;

In[208]:= CoordenadasTotales = PrependTo[CoordenadasOriginal, CoordenadasBaseNestle[[1]];
(*añadir al principio*)

In[220]:= NodosDefinitivos = GeoPosition /@ CoordenadasTotales;
(*posición geográfica*)
NodosMarcadosDefinitivos = Table[NodosDefinitivos[[i, 1, nOriginal + 1]];
(*ruta*)
Graphics[{NodosMarcadosDefinitivos, {PointSize[0.009], Blue, Point[NodosDefinitivos]}}, GeoBackground -> "VectorMinimal", GeoRange -> Automatic]
(*mapa geográfico*) (*tamaño de punto*) (*color*) (*punto*) (*fondo geográfico*) (*rango geogr.*) (*automático*)
```

#### VMC ACTUAL

##### Rutas actuales

```
In[212]:= ObtenerCoordenadas[ruta_, dia_] := Module[{coincidencias}, coincidencias = Select[DatosOriginal, #[[3]] == ruta && #[[4]] == dia &];
(*ruta*) (*selección*)
Return[coincidencias[[1]], {10, 11}];
(*ruta*)

In[213]:= rutas = {"EC9201", "EC9202", "EC9203", "EC9204", "EC9205", "EC9206", "EC9207", "EC9208", "EC9209", "EC9210", "EC9211", "EC9212", "EC9213", "EC9214", "EC9215", "EC9216", "EC9217", "EC9218", "EC9219",
"EC9220", "EC9221", "EC9222", "EC9223", "EC9224", "EC9225", "EC9226", "EC9227", "EC9228", "EC9229", "EC9230", "EC9231", "EC9232", "EC9233", "EC9234", "EC9235", "EC9236", "EC9237", "EC9238", "EC9239", "EC9240", "EC9241", "EC9242",
"EC9243", "EC9244", "EC9245", "EC9246", "EC9247", "EC9248", "EC9249", "EC9250", "EC9251", "EC9252", "EC9253", "EC9254", "EC9255", "EC9256", "EC9257", "EC9258", "EC9259", "EC9260", "EC9261", "EC9262"};
dias = {0, 2, 3, 4, 5, 6, 7};

In[215]:= ObtenerTodasCoordenadas[rutas_List, dias_List] := Module[{resultados}, resultados = Table[{ruta, dia} + If[Length[coor = ObtenerCoordenadas[ruta, dia]] > 0, coor, {}], {ruta, rutas}, {dia, dias}];
(*ruta*) (*dia*) (*ruta*) (*dia*)
Flatten[resultados, 1];
(*aplana*)

In[216]:= coordenadas = ObtenerTodasCoordenadas[rutas, dias];
```

##### MD actuales

```
In[217]:= MatrizDist[coor_] := Module[{coor0 = coor},
(*ruta*)
Table[ManhattanDistance[coor0[[i]], coor0[[j]], {i, Length[coor0]}, {j, Length[coor0]}];
(*ruta*) (*distancia Manhattan*) (*longitud*) (*longitud*)
```

##### coordenadas de cada ruta x día

```
In[218]:= rutaDia[coordenadas_] := Table[r = coordenadas[[1]][2],
(*ruta*)
{1, Length[coordenadas]}];
(*longitud*)

(*revisar si se usa sino borrar*)
In[219]:= GenerarCodigo[coordenadas_List] := Module[{codigo},
(*ruta*) (*resultado*)
codigo = Table[coordenadas[[1]][2], {1, Length[coordenadas]}];
(*ruta*) (*longitud*)

In[220]:= x = GenerarCodigo[coordenadas];

In[221]:= xConBase = Prepend[x, CoordenadasBaseNestle[[1]]] & /@ x;
(*añadir al principio*)
```

### matriz distancia de cada ruta x día

```
h[22]= MDistancia[coordenadasactuales_] := Module[{geocartesianas, geocartesianas_n, nombresArchivos},  
  (*Inicio*)  
  n = Table[If[Length[coordenadasactuales[i]] > 0,  
    Table[{  
      geoactual = GeoPosition[# coordenadasactuales[i]];  
      geocartesianas = GeoGridPosition[geoactual, "UTMZone17South"];  
      MatrizDist[geocartesianas[i], {0}], {i, Length[coordenadasactuales]}]]],  
    {Length[coordenadasactuales]}];  
  (*Fin*)
```

```
h[22]= MDx = MDistancia[xConBase];
```

### Función VMCM[nodosTotales\_, matrizDist\_]

```
h[22]= VMCM[nodosTotales2_Integer, matrizDistancia2_List] := And[  
  (*Inicio*)  
  nodos2 = Range[nodosTotales2];  
  mejorDist2 = 99999999; mejorTour2 = {};  
  Do[  
    (*Inicio*)  
    base2 = 1;  
    tour2 = {base2};  
    norut2 = DeleteCases[nodos2, base2];  
    (*Fin de caso*)  
    dist2 = 0;  
    Do[  
      (*Inicio*)  
      deln2 = Min[matrizDistancia2[[tour2[-1], norut2]]; k2 = Position[matrizDistancia2[[tour2[-1], norut2], deln2][[1, 1];  
      (*Fin*)  
      AppendTo[tour2, norut2[[k2]];  
      (*Inicio*)  
      norut2 = DeleteCases[norut2, norut2[[k2]];  
      (*Fin de caso*)  
      dist2 += deln2, {i, nodosTotales2 - 1};  
      dist2 = matrizDistancia2[[tour2[-1], base2];  
      AppendTo[tour2, base2];  
      (*Inicio*)  
      If[dist2 < mejorDist2, mejorDist2 = dist2; mejorTour2 = tour2, {}, 1, nodosTotales2];  
      (*Fin*)  
    ];  
  ];  
  mejorDist2, mejorTour2  
  ]
```

### Tour de rutas actuales

```
h[22]= ruta1d0 = VMCM[Length[xConBase [1]], MDx [1]];  
ruta1d2 = VMCM[Length[xConBase [2]], MDx [2]];  
ruta1d3 = VMCM[Length[xConBase [3]], MDx [3]];  
ruta1d4 = VMCM[Length[xConBase [4]], MDx [4]];  
ruta1d5 = VMCM[Length[xConBase [5]], MDx [5]];  
ruta1d6 = VMCM[Length[xConBase [6]], MDx [6]];  
ruta1d7 = VMCM[Length[xConBase [7]], MDx [7]];  
  
ruta2d0 = VMCM[Length[xConBase [8]], MDx [8]];  
ruta2d2 = VMCM[Length[xConBase [9]], MDx [9]];  
ruta2d3 = VMCM[Length[xConBase [10]], MDx [10]];  
ruta2d4 = VMCM[Length[xConBase [11]], MDx [11]];  
ruta2d5 = VMCM[Length[xConBase [12]], MDx [12]];  
ruta2d6 = VMCM[Length[xConBase [13]], MDx [13]];  
ruta2d7 = VMCM[Length[xConBase [14]], MDx [14]];  
  
ruta3d0 = VMCM[Length[xConBase [15]], MDx [15]];  
ruta3d2 = VMCM[Length[xConBase [16]], MDx [16]];  
ruta3d3 = VMCM[Length[xConBase [17]], MDx [17]];  
ruta3d4 = VMCM[Length[xConBase [18]], MDx [18]];  
ruta3d5 = VMCM[Length[xConBase [19]], MDx [19]];  
ruta3d6 = VMCM[Length[xConBase [20]], MDx [20]];  
ruta3d7 = VMCM[Length[xConBase [21]], MDx [21]];  
  
ruta4d0 = VMCM[Length[xConBase [22]], MDx [22]];  
ruta4d2 = VMCM[Length[xConBase [23]], MDx [23]];  
ruta4d3 = VMCM[Length[xConBase [24]], MDx [24]];  
ruta4d4 = VMCM[Length[xConBase [25]], MDx [25]];  
ruta4d5 = VMCM[Length[xConBase [26]], MDx [26]];  
ruta4d6 = VMCM[Length[xConBase [27]], MDx [27]];  
ruta4d7 = VMCM[Length[xConBase [28]], MDx [28]];  
  
ruta5d0 = VMCM[Length[xConBase [29]], MDx [29]];  
ruta5d2 = VMCM[Length[xConBase [30]], MDx [30]];
```

```

ruta5d2 = VMCN[Length[xConBase [30]], MDx[30]];
ruta5d3 = VMCN[Length[xConBase [31]], MDx[31]];
ruta5d4 = VMCN[Length[xConBase [32]], MDx[32]];
ruta5d5 = VMCN[Length[xConBase [33]], MDx[33]];
ruta5d6 = VMCN[Length[xConBase [34]], MDx[34]];
ruta5d7 = VMCN[Length[xConBase [35]], MDx[35]];

ruta6d0 = VMCN[Length[xConBase [36]], MDx[36]];
ruta6d2 = VMCN[Length[xConBase [37]], MDx[37]];
ruta6d3 = VMCN[Length[xConBase [38]], MDx[38]];
ruta6d4 = VMCN[Length[xConBase [39]], MDx[39]];
ruta6d5 = VMCN[Length[xConBase [40]], MDx[40]];
ruta6d6 = VMCN[Length[xConBase [41]], MDx[41]];
ruta6d7 = VMCN[Length[xConBase [42]], MDx[42]];

ruta7d0 = VMCN[Length[xConBase [43]], MDx[43]];
ruta7d2 = VMCN[Length[xConBase [44]], MDx[44]];
ruta7d3 = VMCN[Length[xConBase [45]], MDx[45]];
ruta7d4 = VMCN[Length[xConBase [46]], MDx[46]];
ruta7d5 = VMCN[Length[xConBase [47]], MDx[47]];
ruta7d6 = VMCN[Length[xConBase [48]], MDx[48]];
ruta7d7 = VMCN[Length[xConBase [49]], MDx[49]];

ruta8d0 = VMCN[Length[xConBase [50]], MDx[50]];
ruta8d2 = VMCN[Length[xConBase [51]], MDx[51]];
ruta8d3 = VMCN[Length[xConBase [52]], MDx[52]];
ruta8d4 = VMCN[Length[xConBase [53]], MDx[53]];
ruta8d5 = VMCN[Length[xConBase [54]], MDx[54]];
ruta8d6 = VMCN[Length[xConBase [55]], MDx[55]];
ruta8d7 = VMCN[Length[xConBase [56]], MDx[56]];

ruta9d0 = VMCN[Length[xConBase [57]], MDx[57]];
ruta9d2 = VMCN[Length[xConBase [58]], MDx[58]];
ruta9d3 = VMCN[Length[xConBase [59]], MDx[59]];
ruta9d4 = VMCN[Length[xConBase [60]], MDx[60]];

In[100] = distRutasActuales = {ruta1d0[1], ruta1d2[1], ruta1d3[1], ruta1d4[1], ruta1d5[1], ruta1d6[1], ruta1d7[1], ruta2d0[1], ruta2d2[1], ruta2d3[1], ruta2d4[1], ruta2d5[1], ruta2d6[1], ruta2d7[1],
ruta3d0[1], ruta3d2[1], ruta3d3[1], ruta3d4[1], ruta3d5[1], ruta3d6[1], ruta3d7[1], ruta4d0[1], ruta4d2[1], ruta4d3[1], ruta4d4[1], ruta4d5[1], ruta4d6[1], ruta4d7[1], ruta5d0[1], ruta5d2[1],
ruta5d3[1], ruta5d4[1], ruta5d5[1], ruta5d6[1], ruta5d7[1], ruta6d0[1], ruta6d2[1], ruta6d3[1], ruta6d4[1], ruta6d5[1], ruta6d6[1], ruta6d7[1], ruta7d0[1], ruta7d2[1], ruta7d3[1], ruta7d4[1],
ruta7d5[1], ruta7d6[1], ruta7d7[1], ruta8d0[1], ruta8d2[1], ruta8d3[1], ruta8d4[1], ruta8d5[1], ruta8d6[1], ruta8d7[1], ruta9d0[1], ruta9d2[1], ruta9d3[1], ruta9d4[1], ruta9d5[1], ruta9d6[1],
ruta9d7[1], ruta10d0[1], ruta10d2[1], ruta10d3[1], ruta10d4[1], ruta10d5[1], ruta10d6[1], ruta10d7[1], ruta11d0[1], ruta11d2[1], ruta11d3[1], ruta11d4[1], ruta11d5[1], ruta11d6[1],
ruta11d7[1], ruta12d0[1], ruta12d2[1], ruta12d3[1], ruta12d4[1], ruta12d5[1], ruta12d6[1], ruta12d7[1], ruta13d0[1], ruta13d2[1], ruta13d3[1], ruta13d4[1], ruta13d5[1], ruta13d6[1],
ruta13d7[1], ruta14d0[1], ruta14d2[1], ruta14d3[1], ruta14d4[1], ruta14d5[1], ruta14d6[1], ruta14d7[1], ruta15d0[1], ruta15d2[1], ruta15d3[1], ruta15d4[1], ruta15d5[1], ruta15d6[1],
ruta15d7[1], ruta16d0[1], ruta16d2[1], ruta16d3[1], ruta16d4[1], ruta16d5[1], ruta16d6[1], ruta16d7[1], ruta17d0[1], ruta17d2[1], ruta17d3[1], ruta17d4[1], ruta17d5[1], ruta17d6[1],
ruta17d7[1], ruta18d0[1], ruta18d2[1], ruta18d3[1], ruta18d4[1], ruta18d5[1], ruta18d6[1], ruta18d7[1], ruta19d0[1], ruta19d2[1], ruta19d3[1], ruta19d4[1], ruta19d5[1], ruta19d6[1],
ruta19d7[1], ruta20d0[1], ruta20d2[1], ruta20d3[1], ruta20d4[1], ruta20d5[1], ruta20d6[1], ruta20d7[1], ruta21d0[1], ruta21d2[1], ruta21d3[1], ruta21d4[1], ruta21d5[1], ruta21d6[1],
ruta21d7[1], ruta22d0[1], ruta22d2[1], ruta22d3[1], ruta22d4[1], ruta22d5[1], ruta22d6[1], ruta22d7[1], ruta23d0[1], ruta23d2[1], ruta23d3[1], ruta23d4[1], ruta23d5[1], ruta23d6[1],
ruta23d7[1], ruta24d0[1], ruta24d2[1], ruta24d3[1], ruta24d4[1], ruta24d5[1], ruta24d6[1], ruta24d7[1], ruta25d0[1], ruta25d2[1], ruta25d3[1], ruta25d4[1], ruta25d5[1], ruta25d6[1],
ruta25d7[1], ruta26d0[1], ruta26d2[1], ruta26d3[1], ruta26d4[1], ruta26d5[1], ruta26d6[1], ruta26d7[1], ruta27d0[1], ruta27d2[1], ruta27d3[1], ruta27d4[1], ruta27d5[1], ruta27d6[1],
ruta27d7[1], ruta28d0[1], ruta28d2[1], ruta28d3[1], ruta28d4[1], ruta28d5[1], ruta28d6[1], ruta28d7[1], ruta29d0[1], ruta29d2[1], ruta29d3[1], ruta29d4[1], ruta29d5[1], ruta29d6[1],
ruta29d7[1], ruta30d0[1], ruta30d2[1], ruta30d3[1], ruta30d4[1], ruta30d5[1], ruta30d6[1], ruta30d7[1], ruta31d0[1], ruta31d2[1], ruta31d3[1], ruta31d4[1], ruta31d5[1], ruta31d6[1],
ruta31d7[1], ruta32d0[1], ruta32d2[1], ruta32d3[1], ruta32d4[1], ruta32d5[1], ruta32d6[1], ruta32d7[1], ruta33d0[1], ruta33d2[1], ruta33d3[1], ruta33d4[1], ruta33d5[1], ruta33d6[1],
ruta33d7[1], ruta34d0[1], ruta34d2[1], ruta34d3[1], ruta34d4[1], ruta34d5[1], ruta34d6[1], ruta34d7[1], ruta35d0[1], ruta35d2[1], ruta35d3[1], ruta35d4[1], ruta35d5[1], ruta35d6[1],
ruta35d7[1], ruta36d0[1], ruta36d2[1], ruta36d3[1], ruta36d4[1], ruta36d5[1], ruta36d6[1], ruta36d7[1], ruta37d0[1], ruta37d2[1], ruta37d3[1], ruta37d4[1], ruta37d5[1], ruta37d6[1],
ruta37d7[1], ruta38d0[1], ruta38d2[1], ruta38d3[1], ruta38d4[1], ruta38d5[1], ruta38d6[1], ruta38d7[1], ruta39d0[1], ruta39d2[1], ruta39d3[1], ruta39d4[1], ruta39d5[1], ruta39d6[1],
ruta39d7[1], ruta40d0[1], ruta40d2[1], ruta40d3[1], ruta40d4[1], ruta40d5[1], ruta40d6[1], ruta40d7[1], ruta41d0[1], ruta41d2[1], ruta41d3[1], ruta41d4[1], ruta41d5[1], ruta41d6[1],
ruta41d7[1], ruta42d0[1], ruta42d2[1], ruta42d3[1], ruta42d4[1], ruta42d5[1], ruta42d6[1], ruta42d7[1], ruta43d0[1], ruta43d2[1], ruta43d3[1], ruta43d4[1], ruta43d5[1], ruta43d6[1],
ruta43d7[1], ruta44d0[1], ruta44d2[1], ruta44d3[1], ruta44d4[1], ruta44d5[1], ruta44d6[1], ruta44d7[1], ruta45d0[1], ruta45d2[1], ruta45d3[1], ruta45d4[1], ruta45d5[1], ruta45d6[1],
ruta45d7[1], ruta46d0[1], ruta46d2[1], ruta46d3[1], ruta46d4[1], ruta46d5[1], ruta46d6[1], ruta46d7[1], ruta47d0[1], ruta47d2[1], ruta47d3[1], ruta47d4[1], ruta47d5[1], ruta47d6[1],
ruta47d7[1], ruta48d0[1], ruta48d2[1], ruta48d3[1], ruta48d4[1], ruta48d5[1], ruta48d6[1], ruta48d7[1], ruta49d0[1], ruta49d2[1], ruta49d3[1], ruta49d4[1], ruta49d5[1], ruta49d6[1],
ruta49d7[1], ruta50d0[1], ruta50d2[1], ruta50d3[1], ruta50d4[1], ruta50d5[1], ruta50d6[1], ruta50d7[1], ruta51d0[1], ruta51d2[1], ruta51d3[1], ruta51d4[1], ruta51d5[1], ruta51d6[1],
ruta51d7[1], ruta52d0[1], ruta52d2[1], ruta52d3[1], ruta52d4[1], ruta52d5[1], ruta52d6[1], ruta52d7[1], ruta53d0[1], ruta53d2[1], ruta53d3[1], ruta53d4[1], ruta53d5[1], ruta53d6[1], ruta53d7[1]};

Suma de km recorridos

In[107] = metrosrecorridos = Total[distRutasActuales];
Out[107] = 36493.5

```

## CLUSTERS (240 clientes x semana)

```
!>@900) CoorndadasBaseNestle = {{-2.118808325846659, -79.9328297543551}};
coorDbasenestleGEO = GeoPosition[CoorndadasBaseNestle[1]];
coorDbasenestleCART = GeoGridPosition[coorDbasenestleGEO, "UTMZone17South"];
coorDbasenestleCART1 = coorDbasenestleCART[1];
!>@901) posicionesClientes = GeoPosition[CoordenadasOriginales];
!>@902) coorDoriginalescartesianas = GeoGridPosition[posicionesClientes, "UTMZone17South"];
coorDoriginalescartesianas1 = coorDoriginalescartesianas[1];
!>@903) clustersCentroides = FindClusters[coorDoriginalescartesianas[1], 50, Method -> "Medoids", PerformanceGoal -> "Speed"];
!>@907) BuscarCentroides[lista_List] := Table[
  a = {x[[1]]};
  b = Polygon[a];
  c = RegionCentroid[b];
  {i, Length[{x}]}
];
!>@908) centroides = BuscarCentroides[clustersCentroides];
Validación: centroides no son clientes 
!>@909) (*Definir dos listas*)
listaCompleta = coorDoriginalescartesianas[1];
listaValidar = centroides;
(*Validar si todos los elementos de la segunda lista están en la primera lista*)
If[SubsetQ[listaCompleta, listaValidar], Print["Todos los elementos están contenidos en la lista completa."], Print["Al menos un elemento no está contenido en la lista completa."]]
!>@912) Intersection[listaCompleta, listaValidar]
Out[912]=
{}

Cluster MIP
!>@813) (*NO CORRER
(*Número de clientes y clusters*)
numClientes=9539;
numClusters=50;
(*Listas de coordenadas cartesianas para clientes y clusters*)
clientes=coorDoriginalescartesianas[1];
length(clientes);
clusters=centroides;
MD=DistanceMatrix[clientes,clusters,DistanceFunction->ManhattanDistance];*)
!>@814) (*NO CORRER
(*Variables de decisión binarias*)
vars=Array{x,{numClientes,numClusters}};
(*Restricciones:cada cliente debe ser asignado a exactamente un cluster*)
restricciones1=Total[vars,{2}]=1;
(*Restricciones:la suma de asignaciones para cada cluster debe ser menor o igual a 240*)
restricciones2=Total[vars,{1}]=192;
(*Restricciones:x_{ij} debe ser binario*)
restricciones3=Element[Flatten[vars],Integers]&&0;Flatten[vars];
(*Función objetivo: minimizar MD*x*)
objetivo=Total[Flatten[MD*vars]];
(*Resolver el problema de optimización*)
solucion=FindMinimum[objetivo,restricciones1,restricciones2,restricciones3,Flatten[vars]];
(*Mostrar la solución*)
Print["Matriz de asignaciones x_{ij}:"];
MatrixForm[Round[Array{x,{numClientes,numClusters}} /. solucion[2]]];*)
!>@815) excelMD = Import["MATRIZBINARIA2.xlsx"];
!>@816) mdTranspuesta = excelMD[2];
!>@817) obtenerPosiciones[matrix_] := Table[
  t = Position[matrix][1, 1];
  tt = Flatten[t];
  {i, Length[matrix]}
];
!>@818) posicionesClusters = obtenerPosiciones[mdTranspuesta];
!>@819) indiceToCoordUTM[cluster_] := coorDoriginalescartesianas[1][[cluster]];
resultadoCoordCart = indiceToCoordUTM/@posicionesClusters;
!>@821) resultado09 = Host[resultadoCoordCart];
!>@822) clustersCentroidesprueba1 = FindClusters[a, 6, Method -> "Medoids", PerformanceGoal -> "Speed"] & /@ resultado09;
cprueba1 = BuscarCentroides /@ clustersCentroidesprueba1;
!>@824) (*NO CORRER
(*Número de clientes y clusters*)
numClientes=192;
```

```

numClusters=6;
<-Crear una lista para almacenar las matrices>
matricesResultados={};

For[{i=1,i<=49,i++},<-Listas de coordenadas cartesianas para clientes y clusters>
<clientes=resultado49[i];
MD=DistanceMatrix[clientes,cpruebaci[],DistanceFunction->ManhattanDistance];
<-Variables de decisión binarias> vars=Array[0,{numClientes,numClusters}];
<-Restricciones:cada cliente debe ser asignado a exactamente un cluster->restricciones1=Total[vars,{2}]=1;
<-Restricciones:la suma de asignaciones para cada cluster debe ser igual a 40->restricciones2=Total[vars,{1}]=32;
<-Restricciones:x_{ij} debe ser binario->restricciones3=Element[Flatten[vars],Integers]&&0&&Flatten[vars];
<-Función objetivo: minimizar MD-> objetivo=Total[Flatten[MD*vars]];
<-Resolver el problema de optimización->solucion=FindMinimum[objetivo,restricciones1,restricciones2,restricciones3],Flatten[vars];
<-Agregar la matriz a la lista de matrices->matricesResultados=Append[matricesResultados,{"Iteracion"<->ToString[i]->Round[vars],solucion[2]}];
<-Imprimir la matriz calculada en esta iteración usando MatrixForm->Print["Matriz en la iteración ",i," ",MatrixForm[Round[vars],solucion[2]]];
<-Puedes acceder a las matrices calculadas en cada iteración desde matricesResultados->matricesResultados
+]

```


Visualización

Funciones

```

SubClusterToCoord[coord_>, subcluster_>] := Table[coord[[subcluster][i]], {i, Length[subcluster]}];
Show[Graphics[GraphicsGroup[{{FaceForm[Blue], FilledCurve[{{Line[Join[{(0, 0)}, {(Cos[ar], 3 + 5 Sin[ar])} & /@ Range[0, 2 Pi, Pi/20], {(0, 0)}], {Line[{{0.5 (Cos[ar], 6 + 5 Sin[ar])} & /@ Range[0, 2 Pi, Pi/20]}]}]}],
Axes -> True, ImageSize -> Small]

```



\*subclusters nuevos

```

posicionesclusters = obtenerPosicionesT[mdTranspuesta];
IndiceFoCoordUTR[cluster_>] := coordoriginalescartesianas[[cluster];
resultadocoordcart = IndiceFoCoordUTR/@posicionesclusters;
resultado49 = Max[resultadocoordcart];
pruebaci = resultadocoordcart[[1];
clusterscentrodespruebaci = FindClusters[pruebaci, 6, Method -> "KMedoids", PerformanceGoal -> "Speed"];
cpruebaci = BuscarCentroides[clusterscentrodespruebaci];
<-Número de clientes y clusters>
numClientes = 240;
numClusters = 6;
<-Listas de coordenadas cartesianas para clientes y clusters>
clientes = pruebaci;
MD = DistanceMatrix[clientes, cpruebaci, DistanceFunction -> ManhattanDistance];
Cluster con 131 elementos
cluster50 = resultadocoordcart[[50];
coordcart = {17.335, 8.934, 30.156, 10.755, 32.731, 0.036, 22.4*6};
cluster50 = Append[cluster50, coordcart];
centrodescluster50 = FindClusters[cluster50, 6, Method -> "KMedoids", PerformanceGoal -> "Speed"];
bcentrodescluster50 = BuscarCentroides[centrodescluster50];
numClientes = 132;
numClusters = 6;
<-Listas de coordenadas cartesianas para clientes y clusters>
clientes = cluster50;
MD = DistanceMatrix[clientes, bcentrodescluster50, DistanceFunction -> ManhattanDistance];
<-Variables de decisión binarias>
vars50 = Array[0, {numClientes, numClusters}];
<-Restricciones:cada cliente debe ser asignado a exactamente un cluster->
restricciones1 = Total[vars50, {2}] = 1;
<-Restricciones:la suma de asignaciones para cada cluster debe ser igual a 40->
restricciones2 = Total[vars50, {1}] = 22;
<-Restricciones:x_{ij} debe ser binario->
restricciones3 = Element[Flatten[vars50], Integers] && 0 && Flatten[vars50] & 1;
<-Función objetivo: minimizar MD->
objetivo = Total[Flatten[MD * vars50]];
<-Resolver el problema de optimización->
solucion = FindMinimum[objetivo, restricciones1, restricciones2, restricciones3], Flatten[vars50];
<-Mostrar la solución->
Print["Matriz de asignaciones u_{ij}:"];
MatrixForm[Round[Array[u, {numClientes, numClusters}] /. solucion[2]]]

```

### subclusters del cluster 50

```
In[854]:= #Mcluster50 = Import["~\CLUSTERS50.xlsx"] [1];
In[855]:= posicionescluster50 = obtenerPosicionesT [#Mcluster50];
(*resultado de coordcart40=IndiceToCoordFH(dposicionescluster40); PARA VIMOR CON GRASP*)
In[856]:= posicionescluster50
Out[856]:=
{{1, 2, 6, 7, 12, 17, 21, 22, 23, 24, 25, 26, 31, 34, 40, 116, 117, 121, 122, 124, 125, 129}, {9, 53, 54, 55, 56, 57, 63, 65, 66, 67, 68, 70, 71, 76, 82, 87, 93, 97, 98, 99, 105, 106},
{10, 14, 27, 28, 29, 36, 41, 43, 48, 49, 52, 62, 64, 69, 74, 95, 100, 101, 109, 118, 119, 123}, {45, 46, 47, 50, 51, 78, 80, 81, 84, 85, 86, 88, 110, 111, 112, 113, 114, 115, 120, 126, 127, 128},
{3, 15, 18, 19, 58, 59, 73, 75, 77, 79, 83, 89, 90, 91, 92, 94, 96, 102, 103, 104, 107, 108}, {4, 5, 8, 11, 13, 16, 20, 30, 32, 33, 35, 37, 38, 39, 42, 44, 60, 61, 72, 130, 131, 132}}
In[857]:= posicionescluster50[6] = DeleteCases[posicionescluster50[6], 132];
(*eliminar casos*)
In[858]:= r50 = posicionesclusters[50];
posicionescluster50
r50[posicionescluster50[1]];
subclusters del cluster 1 al 49
```

```
In[861]:= #Mcluster149 = Import["MATRIZTRANSPUESTA.xlsx"];
In[862]:= posiciones149 = obtenerPosicionesT [#Mcluster149];
In[863]:= variablesResultado = Table[Symbol["pc"<> ToString[i]], {i, Length[posicionesclusters]};
(*Tabla (columna) (construye a cada una de) (longitud)*)
Do[Evaluate[variablesResultado[[i]]] = posicionesclusters[[i]], {i, Length[posicionesclusters]}];
(*[i] (columna) (longitud)*)
In[864]:= posicionescluster49 = Most[posicionesclusters];
(*eliminar el ultimo*)
In[864]:= (*PARA IMPRIMIR POR CLUSTER*)
obtenerUnResultado[pc_, posiciones139_] := Module[{resultado}, resultado = Table[pc[posiciones139[[i, j]], {i, Length[posiciones139]}, {j, Length[posiciones139[1]]}]; resultado]
In[865]:= prueba50 = obtenerUnResultado[posicionesclusters[50], posicionescluster50];
In[866]:= (*PARA IMPRIMIR todos los CLUSTERS*)
obtenerResultados[posicionescluster39_, posiciones139_] := Module[{resultado}, resultado = Table[Table[posicionescluster39[[i, j]], {j, Length[posiciones139[1]]}, {i, Length[posicionescluster39]}];
resultado]
In[867]:= prueba = obtenerResultados[posicionescluster49, posiciones149];
In[868]:= obtenerResultadosCart[coordcart_, prueba_] := Module[{cart}, cart = Table[coordcart[prueba[[i, j]], {i, Length[prueba]}, {j, Length[prueba[1]]}]; cart]
(*columna (cart) (tabla (longitud) (longitud)*)
In[869]:= coordcartesianasSINBASE49 = obtenerResultadosCart[coordoriginalescartesianas1, prueba];
Length[coordcartesianasSINBASE49];
(*longitud*)
In[870]:= coordcartesianasSINBASE50 = obtenerResultadosCart[coordoriginalescartesianas1, prueba50];
Length[coordcartesianasSINBASE50];
(*longitud*)
In[871]:= coordcartesianas150 = Append[coordcartesianasSINBASE49, coordcartesianasSINBASE50];
(*tabla*)
In[872]:= (*coordcartesianas140[1:40][1:6]-*)
agregarBase[coordnestle_, coordcartesianas_] := {resultadosbase = Table[Prepend[coordcartesianas[[i, j]], coordnestle, {i, Length[coordcartesianas]}, {j, Length[coordcartesianas[1]]}],
(*tabla (tabla al principio) (longitud) (longitud)*)
coordcartesianasCONBASE50 = agregarBase[coordbasenestleCART1, coordcartesianas150];
In[873]:= (*coordcartesianasCONBASE40[1:50][1:6]-*)
coordcartesianasCONBASE50[1];
(*longitud*)
```

### Función MD

```
In[881]:= MatrizDist[coord_] := Module[{coord0 = coord},
(*columna*)
Table[MahattanDistance[coord0[[i]], coord0[[j]], {i, Length[coord0]}, {j, Length[coord0]}];
(*tabla (distancia Manhattan) (longitud) (longitud)*)
mprueba = MatrizDist[coordcartesianasCONBASE50[1]];
In[882]:= matricesDistancia = Map[MatrizDist, coordcartesianasCONBASE50, {2}];
(*tabla a lista*)
In[884]:= (*YA ESTAN EXPORTADAS LAS MATRICES DE TODOS LOS SUBCLUSTERS, NO ES NECESARIO VOLVER A CORRER ESTO DE NUEVO*)
Do[Export["MD"<> ToString[i] <> ToString[j] <> ".xlsx", matricesDistancia[[i, j], "xlsx"], {i, Length[matricesDistancia]}, {j, Length[matricesDistancia[1]]}];
(*[i] (columna) (construye a cada (columna de cada caso) (longitud) (longitud)*)
```

## RUTEO (grasp con VMCM)

### Función construirTourGRASP[n\_Integer,matrizDist\_]

```
In[891]:= construirTourGRASP[n_Integer, d_List] := Module[{nodos, n, dneJorGRASP,
(*columna (tabla) (columna)*)
tourmejor, mejorTourGRASP, dneJor, base, tour, nrusl, dist, lcr,
seleccion, k, d2Optmejor, tour2Optmejor, tour2Opt, d2Opt, i, j,
niteraciones},
niteraciones = 300;
nodos = Range[n];
(*rango*)
n = 3; (* La LCR va a estar formada por los 2 más cercanos *)
dneJorGRASP = *m;
Do[
(*columna*)
dneJor = *m;
```



```

Do[
  [nodo]
  base = 1;
  tour = {base};
  norut = DeleteCases[nodos, base];
  [eliminar casos]
  dist = 0;

Do[
  [repet]
  (* Selección GRASP *)
  If[Length[norut] < n, m = n - Length[norut]];
  In [longitud]
  lcr = TakeSmallest[d[[tour[-1], norut], n]]; (*lcr*)
  [toma los menores]

  seleccion = RandomChoice[lcr];
  [selección aleatoria]
  k = Position[d[[tour[-1], norut], seleccion][1, 1]];
  [posición]
  (* Fin selección GRASP *)

  AppendTo[tour, norut[[k]];
  [añadir al tour]
  norut = DeleteCases[norut, norut[[k]];
  [eliminar casos]
  dist += seleccion
  , {n - 1}];

  dist += d[[tour[-1], base]];
  AppendTo[tour, base];
  [añadir al tour]

  If[dist < dmejor, tourmejor = tour; dmejor = dist];
  [si]

  , {i, n}];

(* MEJORA LOCAL *)
d2Optmejor = dmejor;
tour2Optmejor = tourmejor;
For[i = 2, i < n - 1,
  [por cada]
  For[j = i + 1, j < n,
    [por cada]
    tour2Opt =
    Join[tourmejor[[1 ;; i - 1], Reverse[tourmejor[[i ;; j]],
    [invertir orden]
    tourmejor[[j + 1 ;; n + 1]]];
    d2Opt = Sum[d[[tour2Opt[[k], tour2Opt[[k + 1]]],
    [suma]
    If[d2Opt < d2Optmejor, d2Optmejor = d2Opt];
    [si]
    tour2Optmejor = tour2Opt[[j + 1]];
    i++;
  ];
  ];
  If[d2Optmejor < dmejorGRASP, dmejorGRASP = d2Optmejor,
  [si]
  mejorTourGRASP = tour2Optmejor];
  , {iteraciones}];
  Return[{dmejorGRASP, mejorTourGRASP}];
  [retorna]
]


```

Aplicación de función (no correr, sino correr RES) 

```

In[100]:= res = {}

For[i = 1, i <= 50, i++,
  (*para cada*)
  For[j = 1, j <= 6, j++,
    (*para cada*)
    prueba = matricesDistancias[[i][j]];
    longitud = Length[prueba];
    resultado = construirFourGRASP[longitud, prueba];
    AppendTo[res, resultado];]
]

distancia total recorrida 

In[101]:= eliminarUltimoElemento[lista_] := Rest /@ lista
(*quita excepto el i*)
distanciar = eliminarUltimoElemento[res];
mrecorridosGRASP = Total[Flatten[distanciar]]
(*total*) (*cifera*)

Out[101]:=
2.0346 * 10^7

In[102]:= kntsemanal = 20346000 / 1000
Out[102]:=
20346

In[103]:= kntanoal = kntsemanal * 52
Out[103]:=
1057892

In[104]:= kntmensual = kntanoal / 12
Out[104]:=
88166

In[105]:=
421.28 - 234.87
Out[105]:=
186.40999999999997

In[106]:=
186.41

clientes reales en rutaq x dia

In[104]:= eliminarPrimerElemento[lista_&lt;list] := Rest /@ lista
(*quita excepto el primero*)
res2 = eliminarPrimerElemento[res];
resF = Flatten[res2, 1];
eliminarUltimosExtremosEnListas[listaDeListas_&lt;list] := Map[Rest /@ Most, listaDeListas]
(*Map*) (*quita*) (*quita excepto el ultimo*)
resfnocd = eliminarUltimosExtremosEnListas[resF];
restarUnoAlLista[lista_&lt;list] := Map[# - 1 &amp;lt;list]
(*quita*) (*quita a todos*)
reslisff = restarUnoAlLista[resfnocd];

In[107]:= listavacia = {}
ele1 = r50[posicionescluster50[1]];
ele2 = r50[posicionescluster50[2]];
ele3 = r50[posicionescluster50[3]];
ele4 = r50[posicionescluster50[4]];
ele5 = r50[posicionescluster50[5]];
ele6 = r50[posicionescluster50[6]];
AppendTo[listavacia, ele1];
AppendTo[listavacia, ele2];
AppendTo[listavacia, ele3];
AppendTo[listavacia, ele4];
AppendTo[listavacia, ele5];
AppendTo[listavacia, ele6];
listavacia);

In[108]:= pclientes = Append[prueba, listavacia];
Out[108]:=
pclientes = Append[prueba, listavacia];

In[109]:= pclientesF = Flatten[pclientes, 1];
Out[109]:=
pclientesF;

In[110]:= tabla = Table[{i, d, pclientes[[i, d - 1]], {i, 1, 50}, {d, 2, 7}};
(*tabla*)

(*-Imprimir la tabla-*)
Grid[Flatten[tabla, 1], Frame -> All, FrameStyle -> Directive[Thickness[1], Black], Spacings -> {2, 1}, Alignment -> Center, Background -> {{LightBlue, None}} // TraditionalForm
(*tabla*) (*cifera*) (*quita*) (*quita de inicio*) (*directiva*) (*grosor*) (*negro*) (*negro/cifera*) (*diferenciando*) (*orden*) (*orden de imagen*) (*color claro*) (*negro*) (*forma tradicional*)

```

## GRAFICAS

### CLUSTERS

#### ARC Y GEO

```
h[726]= npc1 = pc1 + 1;
npc2 = pc2 + 1;
npc3 = pc3 + 1;
npc4 = pc4 + 1;
npc5 = pc5 + 1;
npc6 = pc6 + 1;
npc7 = pc7 + 1;
npc8 = pc8 + 1;
npc9 = pc9 + 1;
npc10 = pc10 + 1;
npc11 = pc11 + 1;
npc12 = pc12 + 1;
npc13 = pc13 + 1;
npc14 = pc14 + 1;
npc15 = pc15 + 1;
npc16 = pc16 + 1;
npc17 = pc17 + 1;
npc18 = pc18 + 1;
npc19 = pc19 + 1;
npc20 = pc20 + 1;
npc21 = pc21 + 1;
npc22 = pc22 + 1;
npc23 = pc23 + 1;
npc24 = pc24 + 1;
npc25 = pc25 + 1;
npc26 = pc26 + 1;
npc27 = pc27 + 1;
npc28 = pc28 + 1;
npc29 = pc29 + 1;
npc30 = pc30 + 1;
npc31 = pc31 + 1;
npc32 = pc32 + 1;
npc33 = pc33 + 1;
npc34 = pc34 + 1;
npc35 = pc35 + 1;
npc36 = pc36 + 1;
npc37 = pc37 + 1;
npc38 = pc38 + 1;
npc39 = pc39 + 1;
npc40 = pc40 + 1;
npc41 = pc41 + 1;
npc42 = pc42 + 1;
npc43 = pc43 + 1;
npc44 = pc44 + 1;
npc45 = pc45 + 1;
npc46 = pc46 + 1;
npc47 = pc47 + 1;
npc48 = pc48 + 1;
npc49 = pc49 + 1;
npc50 = pc50 + 1;
geo1 = NodosDefinitivos[npc1];
geo2 = NodosDefinitivos[npc2];
geo3 = NodosDefinitivos[npc3];
geo4 = NodosDefinitivos[npc4];
geo5 = NodosDefinitivos[npc5];
geo6 = NodosDefinitivos[npc6];
geo7 = NodosDefinitivos[npc7];
geo8 = NodosDefinitivos[npc8];
geo9 = NodosDefinitivos[npc9];
geo10 = NodosDefinitivos[npc10];
geo11 = NodosDefinitivos[npc11];
geo12 = NodosDefinitivos[npc12];
geo13 = NodosDefinitivos[npc13];
geo14 = NodosDefinitivos[npc14];
geo15 = NodosDefinitivos[npc15];
geo16 = NodosDefinitivos[npc16];
geo17 = NodosDefinitivos[npc17];
geo18 = NodosDefinitivos[npc18];
geo19 = NodosDefinitivos[npc19];
geo20 = NodosDefinitivos[npc20];
geo21 = NodosDefinitivos[npc21];
geo22 = NodosDefinitivos[npc22];
geo23 = NodosDefinitivos[npc23];
geo24 = NodosDefinitivos[npc24];
geo25 = NodosDefinitivos[npc25];
geo26 = NodosDefinitivos[npc26];
geo27 = NodosDefinitivos[npc27];
geo28 = NodosDefinitivos[npc28];
geo29 = NodosDefinitivos[npc29];
geo30 = NodosDefinitivos[npc30];
geo31 = NodosDefinitivos[npc31];
geo32 = NodosDefinitivos[npc32];
geo33 = NodosDefinitivos[npc33];
geo34 = NodosDefinitivos[npc34];
geo35 = NodosDefinitivos[npc35];
geo36 = NodosDefinitivos[npc36];
geo37 = NodosDefinitivos[npc37];
geo38 = NodosDefinitivos[npc38];
geo39 = NodosDefinitivos[npc39];
geo40 = NodosDefinitivos[npc40];
geo41 = NodosDefinitivos[npc41];
geo42 = NodosDefinitivos[npc42];
geo43 = NodosDefinitivos[npc43];
geo44 = NodosDefinitivos[npc44];
geo45 = NodosDefinitivos[npc45];
geo46 = NodosDefinitivos[npc46];
geo47 = NodosDefinitivos[npc47];
geo48 = NodosDefinitivos[npc48];
geo49 = NodosDefinitivos[npc49];
geo50 = NodosDefinitivos[npc50];
h[786]= tourgrafica = resultadoFinalConCeros + 1;
tourgrafica[1] = (para tour=
Out[786]= [1, 4637, 4710, 5077, 5078, 5064, 5036, 5025, 5105, 5163, 5222, 5084, 5122, 5153, 5161, 7661, 185, 92, 8103, 270, 99, 230, 4708, 87, 208, 148, 100, 5030, 5204, 5142, 5138, 5236, 7643, 1]
```

