



Team 36

Route Sense

FDR Presentation



Route
Sense

A large red graphic showing three location pins connected by a red line, forming a triangle.

Contents

Elevator Pitch
Customers' Needs
Specifications
Constraints
Gantt Chart
Design
Roles
Bill of Material and Budgeting
Meeting Specifications & prototype



ELEVATOR PITCH

Businesses and delivery services in Saudi Arabia face challenges with vague delivery times, leading to long waiting time. So, for e-commerce companies and individuals operating in the Kingdom who need to optimize routes and provide customers with precise delivery windows, Route Sense is the solution. It simplifies logistics by planning optimal routes and offering precise and accurate delivery windows.



Hi Nawaf,

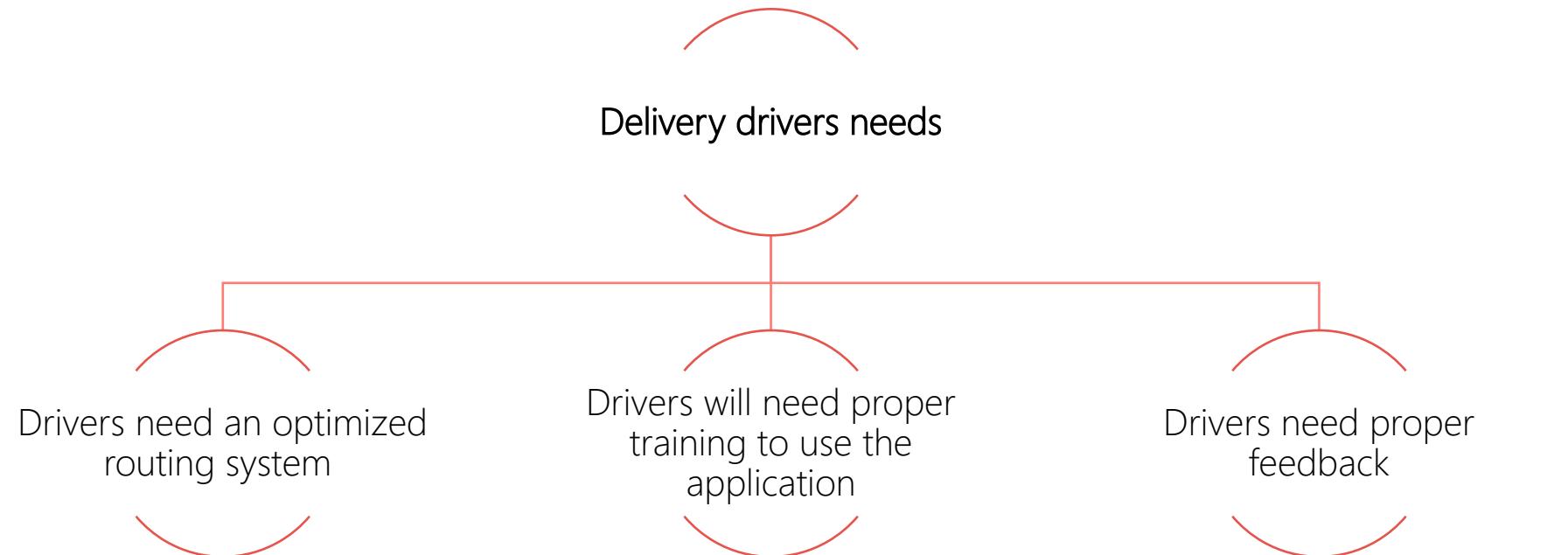
Your package will be delivered between 9:00 AM - 9:00 PM by our Amazon Delivery Agent.

[Manage Your Order](#)

This email was sent from an email address that can't receive emails. Please don't reply to this email.



Customers' Needs



Retailer/Company
needs

Will require an effective and efficient logistical approach

Will require that the application can be integrated easily

Will require training materials



PRODUCT CONSTRAINTS

Application

Limited map functionality

The application's database requires constant internet connection

Database queries are limited

TSP & VRTW model

Depend on data accuracy

Driver inputs (i.e removing customers that cancel from the set of nodes in the TSP)

Start and end tour at the DC (distribution center)

Capacity of delivery vehicles

Datasets and simulations

Limited free data for Saudi Arabia

Traffic

Relies on the Google Maps Directions API

Can change at any moment and changes are not possible to predict

Inaccurate traffic data is possible



PRODUCT'S TECHNICAL SPECIFICATIONS

- User-friendly interface
- Web portal for admins
- Firebase authentication and database
- Flutter multi-platform IOS & Android Runs on 97% of
Android devices and newest IOS
- Google maps API
- TSP model implementation
- Concept car





Gantt Chart



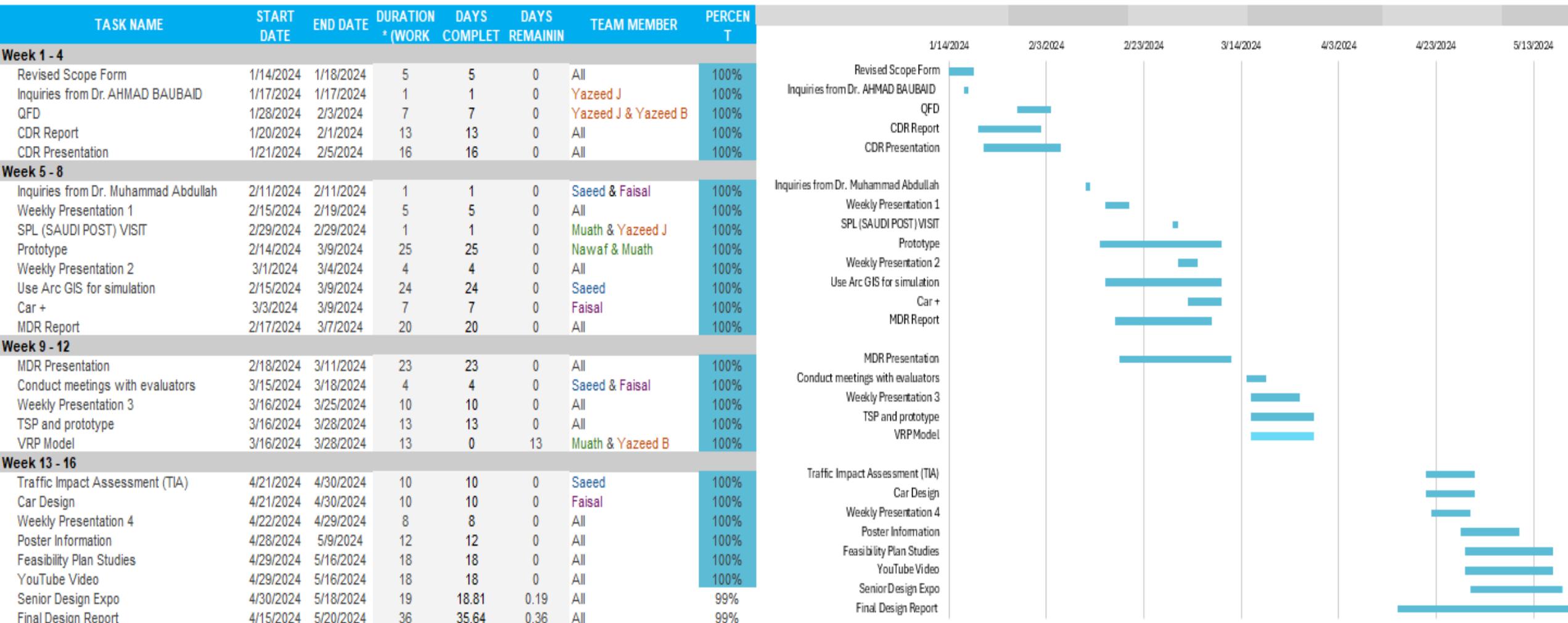
Team 36

ICS: Nawaf & Muath

ISE: Yazeed J & Yazeed B

CE: Saeed

ME: Faisal





FINAL DESIGN DETAILS - ISE

VRPTW MODEL

The code on the right represents the VRPTW model with an objective function that minimize the total distance traveled and Constraints:

- Each shipment must be served exactly once
- Time window constraints
- Capacity constraints for drivers

VRPTW MODEL IS REDACTED



FINAL DESIGN DETAILS - ISE

VRPTW MODEL - Output

Result - Optimal solution found

Objective value: 38.63704083

Enumerated nodes: 31

Total iterations: 294

Time (CPU seconds): 95.52

Time (Wallclock seconds): 100.01

Option for printingOptions changed from normal to all

Total time (CPU seconds): 95.68 (Wallclock seconds): 100.21

Driver 1 delivers the following shipments with respective time windows:

Shipment 3: Time Window 01:27 to 07:27
Shipment 9: Time Window 00:18 to 06:18
Shipment 10: Time Window 01:15 to 07:15
Shipment 18: Time Window 01:38 to 07:38
Shipment 28: Time Window 00:08 to 06:08
Shipment 36: Time Window 00:58 to 06:58
Shipment 40: Time Window 01:55 to 07:55
Shipment 44: Time Window 00:06 to 06:06
Shipment 51: Time Window 00:20 to 06:20
Shipment 53: Time Window 01:01 to 07:01
Shipment 59: Time Window 00:51 to 06:51
Shipment 63: Time Window 01:10 to 07:10
Shipment 72: Time Window 00:19 to 06:19
Shipment 74: Time Window 00:24 to 06:24
Shipment 75: Time Window 01:36 to 07:36
Shipment 76: Time Window 00:05 to 06:05
Shipment 86: Time Window 00:41 to 06:41
Shipment 88: Time Window 00:12 to 06:12
Shipment 90: Time Window 00:27 to 06:27
Shipment 98: Time Window 01:22 to 07:22

Driver 2 delivers the following shipments with respective time windows:

Shipment 6: Time Window 01:37 to 07:37
Shipment 14: Time Window 00:04 to 06:04
Shipment 17: Time Window 00:45 to 06:45
Shipment 21: Time Window 00:38 to 06:38
Shipment 30: Time Window 00:53 to 06:53
Shipment 31: Time Window 01:26 to 07:26
Shipment 47: Time Window 00:19 to 06:19
Shipment 49: Time Window 00:18 to 06:18
Shipment 56: Time Window 01:45 to 07:45
Shipment 58: Time Window 00:05 to 06:05
Shipment 67: Time Window 01:35 to 07:35
Shipment 68: Time Window 00:01 to 06:01
Shipment 69: Time Window 01:37 to 07:37
Shipment 73: Time Window 01:32 to 07:32
Shipment 77: Time Window 01:49 to 07:49
Shipment 82: Time Window 01:36 to 07:36
Shipment 84: Time Window 01:12 to 07:12
Shipment 85: Time Window 01:13 to 07:13
Shipment 87: Time Window 01:27 to 07:27
Shipment 97: Time Window 01:56 to 07:56

Driver 3 delivers the following shipments with respective time windows:

Shipment 1: Time Window 00:10 to 06:10
Shipment 7: Time Window 00:57 to 06:57
Shipment 11: Time Window 00:33 to 06:33
Shipment 20: Time Window 00:28 to 06:28
Shipment 25: Time Window 01:22 to 07:22
Shipment 33: Time Window 01:14 to 07:14
Shipment 35: Time Window 00:54 to 06:54
Shipment 39: Time Window 01:00 to 07:00
Shipment 41: Time Window 01:55 to 07:55
Shipment 43: Time Window 00:30 to 06:30
Shipment 50: Time Window 01:38 to 07:38
Shipment 54: Time Window 00:01 to 06:01
Shipment 60: Time Window 01:40 to 07:40
Shipment 64: Time Window 01:01 to 07:01
Shipment 65: Time Window 01:47 to 07:47
Shipment 80: Time Window 00:58 to 06:58
Shipment 81: Time Window 01:15 to 07:15
Shipment 94: Time Window 01:07 to 07:07
Shipment 95: Time Window 00:38 to 06:38
Shipment 96: Time Window 00:02 to 06:02

Driver 4 delivers the following shipments with respective time windows:

Shipment 2: Time Window 00:42 to 06:42
Shipment 5: Time Window 01:01 to 07:01
Shipment 8: Time Window 01:20 to 07:20
Shipment 12: Time Window 01:10 to 07:10
Shipment 13: Time Window 01:59 to 07:59
Shipment 15: Time Window 00:07 to 06:07
Shipment 23: Time Window 00:14 to 06:14
Shipment 27: Time Window 01:02 to 07:02
Shipment 29: Time Window 01:14 to 07:14
Shipment 37: Time Window 01:15 to 07:15
Shipment 42: Time Window 00:54 to 06:54
Shipment 45: Time Window 00:11 to 06:11
Shipment 52: Time Window 00:38 to 06:38
Shipment 55: Time Window 00:07 to 06:07
Shipment 70: Time Window 01:08 to 07:08
Shipment 71: Time Window 01:38 to 07:38
Shipment 78: Time Window 01:56 to 07:56
Shipment 91: Time Window 01:53 to 07:53
Shipment 93: Time Window 00:06 to 06:06
Shipment 100: Time Window 01:00 to 07:00

Driver 5 delivers the following shipments with respective time windows:

Shipment 4: Time Window 00:35 to 06:35
Shipment 16: Time Window 00:42 to 06:42
Shipment 19: Time Window 01:53 to 07:53
Shipment 22: Time Window 01:31 to 07:31
Shipment 24: Time Window 01:35 to 07:35
Shipment 26: Time Window 01:49 to 07:49
Shipment 32: Time Window 00:16 to 06:16
Shipment 34: Time Window 00:08 to 06:08
Shipment 38: Time Window 01:27 to 07:27
Shipment 46: Time Window 01:51 to 07:51
Shipment 48: Time Window 00:44 to 06:44
Shipment 57: Time Window 01:16 to 07:16
Shipment 61: Time Window 01:49 to 07:49
Shipment 62: Time Window 00:31 to 06:31
Shipment 66: Time Window 00:13 to 06:13
Shipment 79: Time Window 00:44 to 06:44
Shipment 83: Time Window 01:35 to 07:35
Shipment 89: Time Window 01:24 to 07:24
Shipment 92: Time Window 00:43 to 06:43
Shipment 99: Time Window 01:51 to 07:51



FINAL DESIGN DETAILS - ISE

TSP MODEL

The code on the right represents the TSP model with an objective function that minimize the total distance traveled and Constraints:

-Visit Each Location Exactly Once

-Start and End at the Depot

-Eliminate Subtours

TSP MODEL IS REDACTED



TSP MODEL Output

Order of the trip:

X0(Depot)-X7-X10-X11-X5-X15-X6-X4-X16-X13-X12-X14-X19-X18-X17-X8-X2-X9-X3-X1-X0(Depot)

Result – Optimal solution found

Objective value: 31.06025722

Enumerated nodes: 134

Total iterations: 10610

Time (CPU seconds): 2.99

Time (Wallclock seconds): 3.57

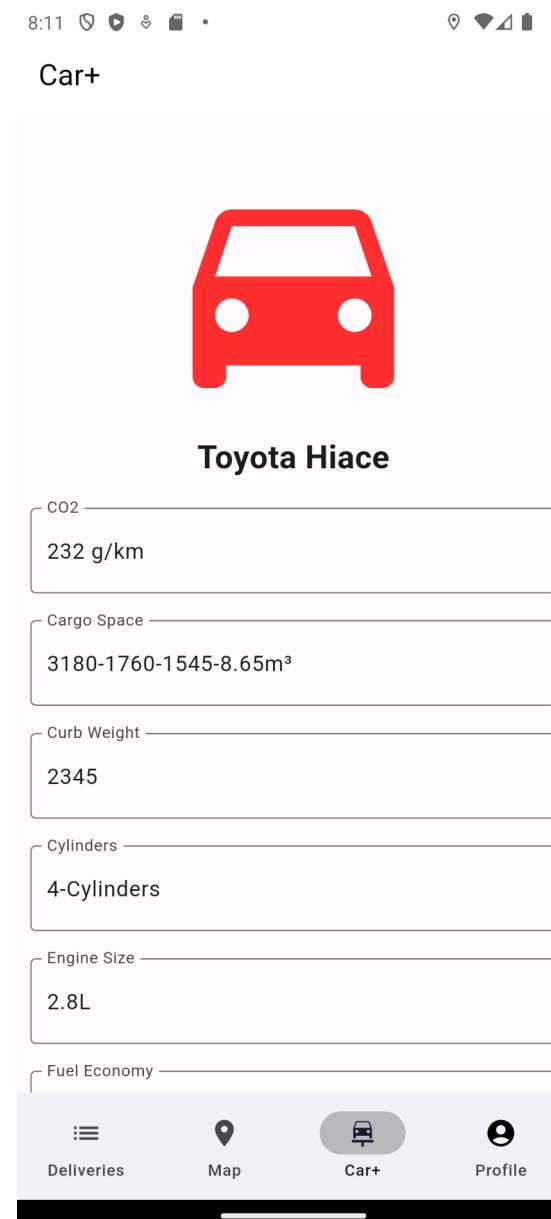
Option for printingOptions changed from normal to all

Total time (CPU seconds): 2.99 (Wallclock seconds): 3.58



CAR+

- Provide car info
 - Helps drivers with car information



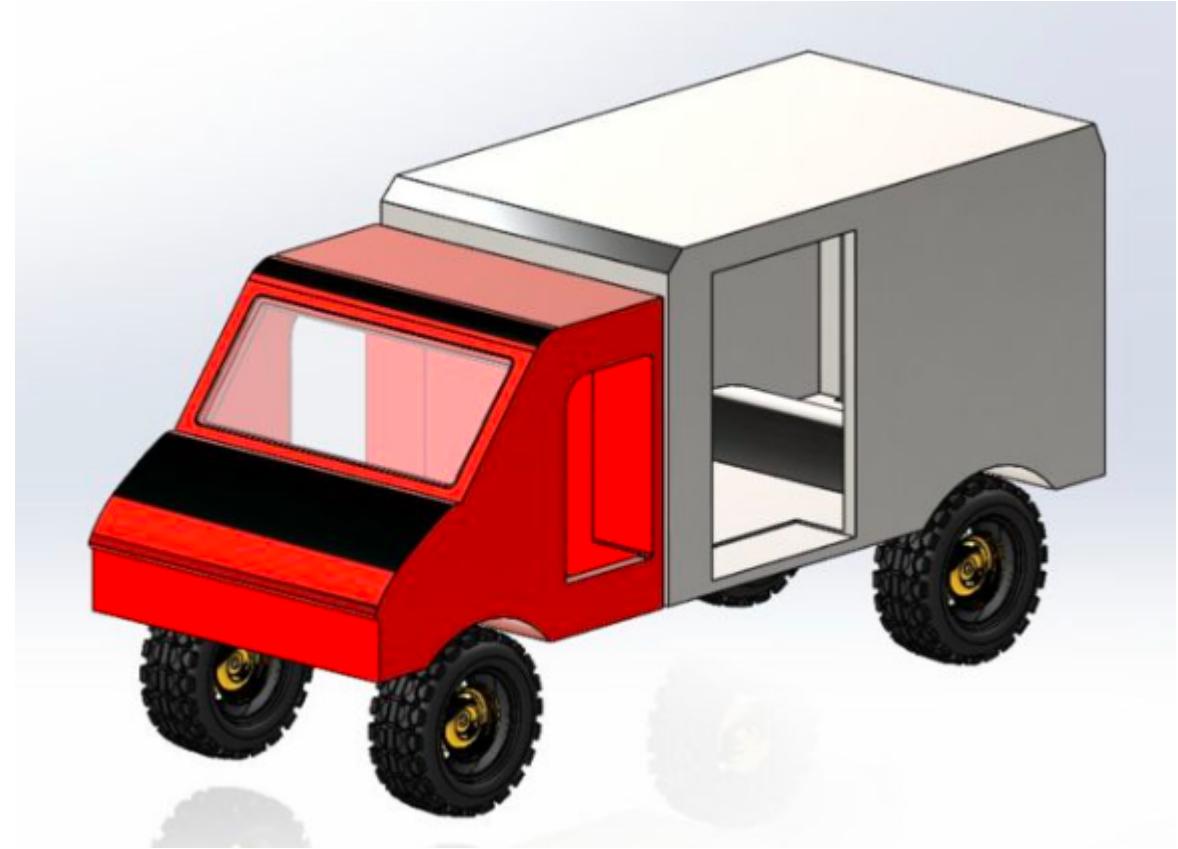


OPTIMAL CAR DESIGN - ME

Delivery Car Design Consideration

- Structural Integrity
- Aerodynamics
- Safety

Engine design study



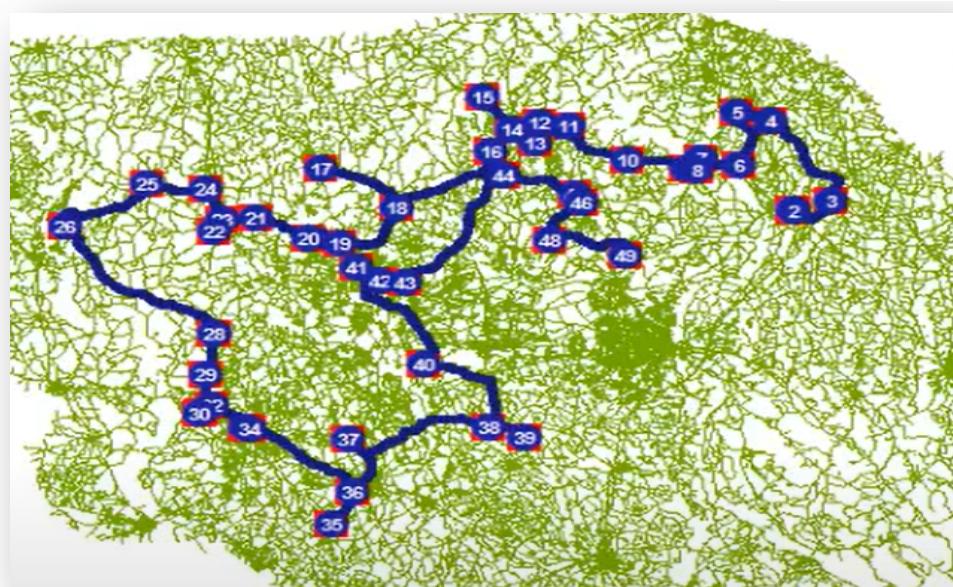
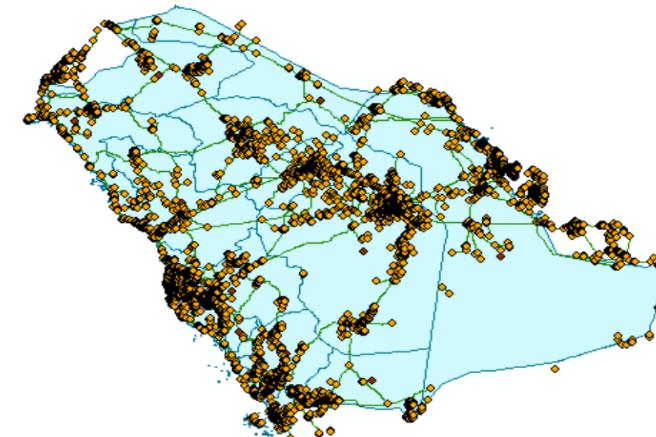


TRAFFIC IMPACT ASSESSMENT (TIA) - CE

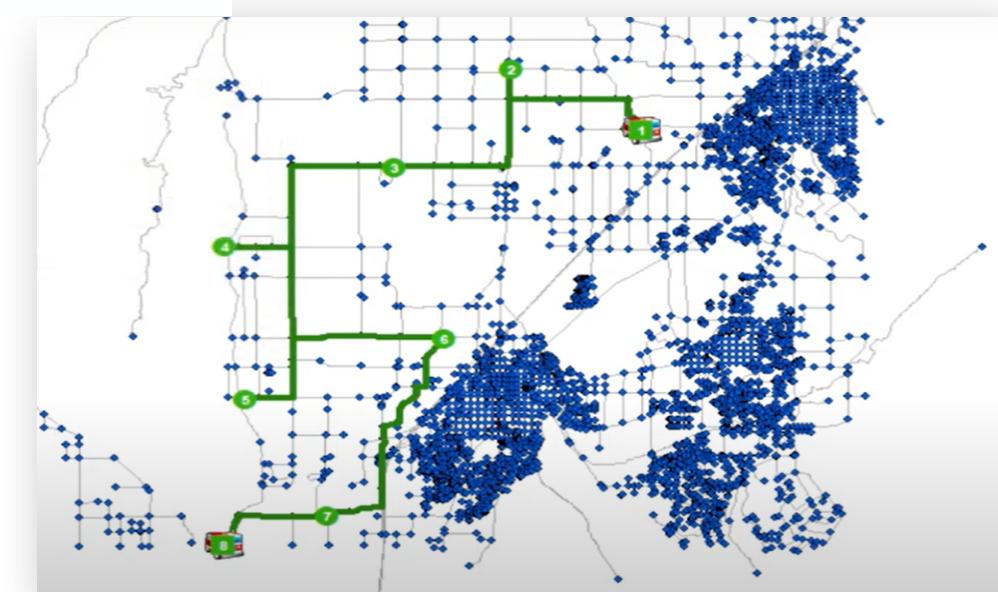
Infrastructure Planning:

- TIA provides insights for infrastructure planning.
- Understand the traffic effects of last-mile delivery activities.
- Make decisions about new construction, road upgrades, signal timing adjustments.





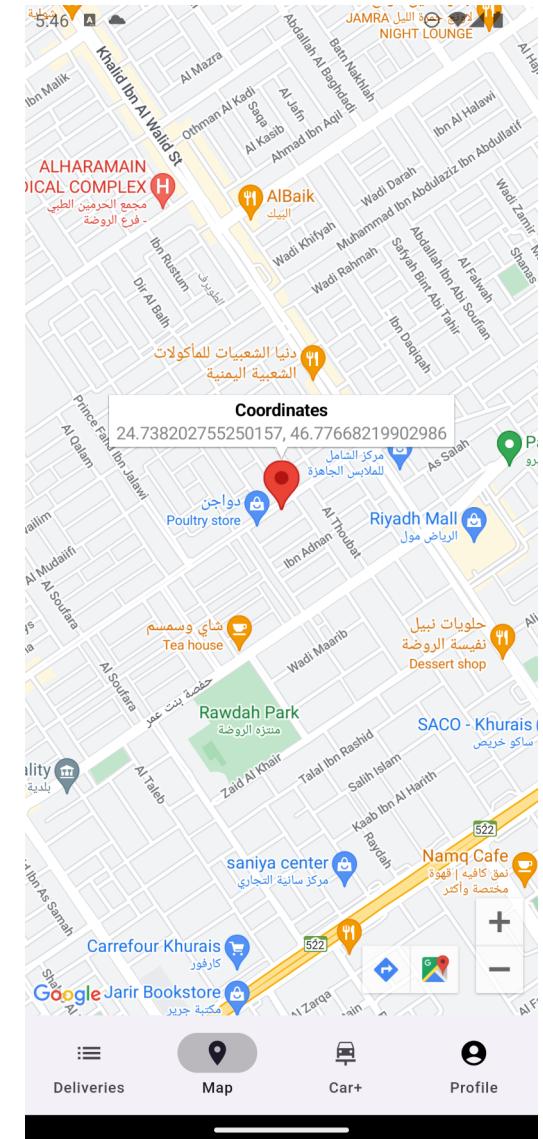
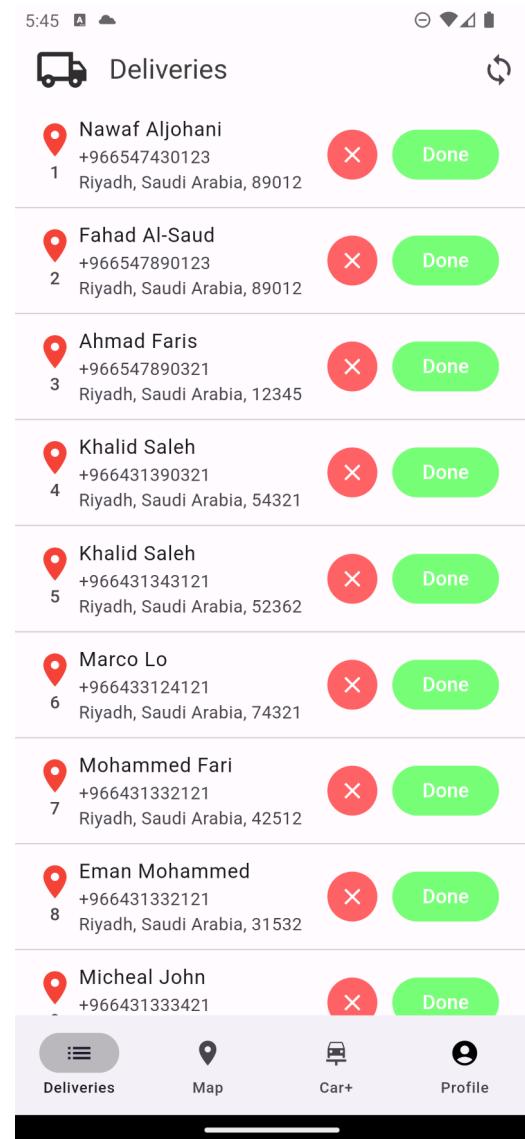
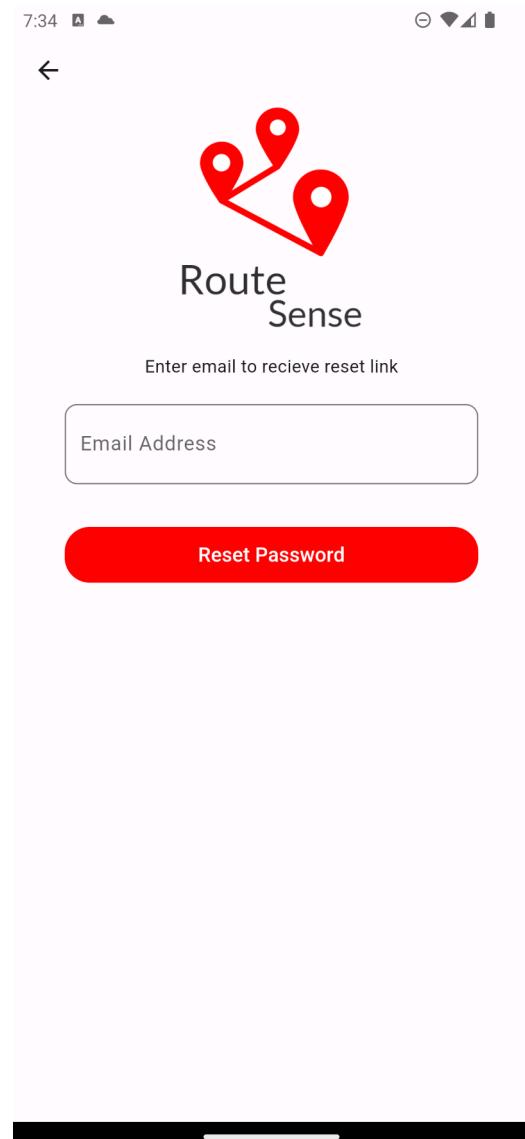
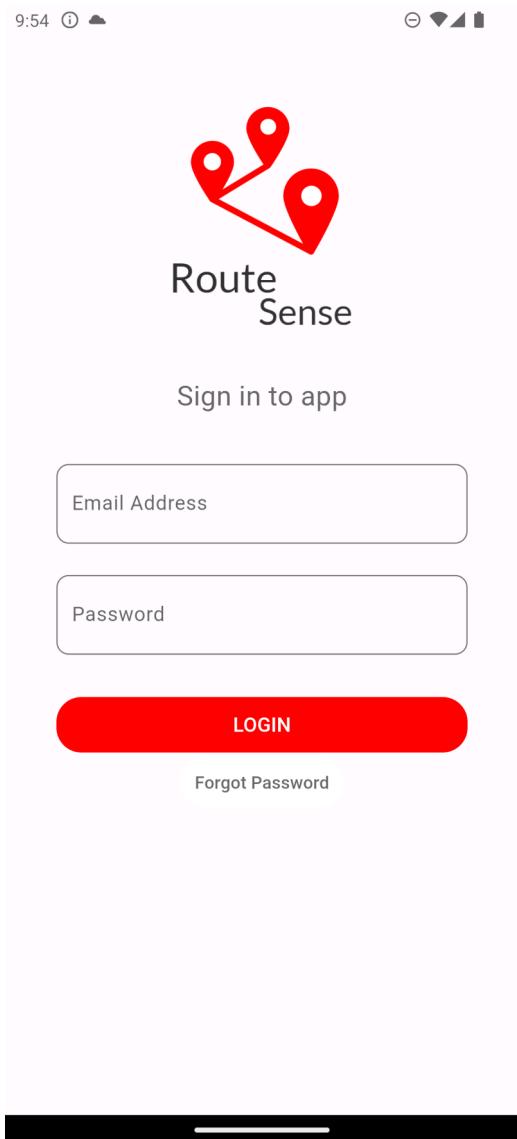
TSP



VRP



FINAL DESIGN DETAILS - ICS





FINAL DESIGN DETAILS - ICS

Route Sense

Log out

Welcome Operator

Dashboard

Add Drivers

Driver Management

Add Collections

Remove Collections

View Collections

TSP

VRP

Search Drivers:

Enter name, uid or email

Search

UID	Name	Email	Phone	Vehicle	Assignment	
94seLUy419VZ6qfSaKmNlcL25KU2	Abdulrahman Al-Ghamdi	abdulrahman.ghamdi@routesense.sa	+966562345678	none		
CgMBDCtbNTV3DWVUZzHGBWViExY2	Hassan Mahmood	hassan.mahmood@routesense.sa	+966562345678	none		
Fitf8nnHdpfMsgOOosqXzv03Ajq2	Mahmoud Abdelaziz	mahmoud.abdelaziz@routesense.sa	+966556789012	none		
M7gLrV0flzcf7VqGJd55AZ7hFck2	Farhan Siddiqui	farhan.siddiqui@routesense.sa	+966543456789	none		
MTs3cyjl6ZBUACyRyyBl448Zu82	Muhammad Khan	muhammad.khan@routesense.sa	+966501234567	none		
QE0YkyUhcPdNl8tuFPHFZxBnAl2	Omar Al-Masri	omar.almasri@routesense.sa	+966539012345	none		



FINAL DESIGN DETAILS - ICS

Route Sense

Log out

Welcome Operator

Dashboard

Add Drivers

Driver Management

Add Collections

Remove Collections

View Collections

TSP

VRP

Search...

Calculate

Search

Result:

Nodes in order:
0,17,18,5,2,19,13,16,11,8,1,4,3,14,10,9,6,12,7,15,0
Delivery Times:
9:00 AM,9:09 AM,9:31 AM,9:56 AM,10:20 AM,10:57 AM,11:39 AM,11:47 AM,0:04 PM,0:18 PM,0:24 PM,0:35 PM,0:45 PM,0:54 PM,1:04 PM,1:16 PM,1:22 PM,1:48 PM,2:09 PM,2:15 PM

client3

Showing Selected Collection Before TSP

Latitude	Longitude	Location	Mobile	Name	Order	Status
24.6341165	46.72863156	Saudi Arabia Riyadh Rawdah	966596022664	Ali Hassan	0	pending
24.79125857	46.76907975	Saudi Arabia Riyadh Rawdah	966596025236	Ahmed Mansour	0	pending
24.60429436	46.73895412	Saudi Arabia Riyadh Rawdah	966596022664	Aisha Al-Ali	0	pending
24.61409653	46.7217487	Saudi Arabia Riyadh Rawdah	966596022664	Youssef Ibrahim	0	pending
24.77988114	46.81456637	Saudi Arabia Riyadh Rawdah	966542532664	Yasmin Al-Haddad	0	pending
24.64426426	46.78911713	Saudi Arabia Riyadh Rawdah	966596022664	Ahmad Nasser	0	pending
24.71489129	46.78763522	Saudi Arabia Riyadh Rawdah	966596022664	Fatima Jamal	0	pending
24.64155354	46.72220853	Saudi Arabia Riyadh Rawdah	966596022664	Fatima Mahmoud	0	pending
24.64502876	46.77734138	Saudi Arabia Riyadh Rawdah	966596022664	Layla Hamdan	0	pending
24.63182301	46.75906571	Saudi Arabia Riyadh Rawdah	966596022664	Kareem Abu-Saleh	0	pending
24.65746825	46.74369584	Saudi Arabia Riyadh Rawdah	966596022664	Khalid Al-Said	0	pending
24.68642355	46.81407044	Saudi Arabia Riyadh Rawdah	966596022664	Omar Qasim	0	pending
24.68907834	46.744638	Saudi Arabia Riyadh Rawdah	966596022664	Rami Farid	0	pending
24.61791932	46.74736588	Saudi Arabia Riyadh Rawdah	966596022664	Dina Sharif	0	pending
24.72437858	46.78343692	Saudi Arabia Riyadh Rawdah	966596022664	Nawaf Aljohani	0	pending
24.68572641	46.7589691	Saudi Arabia Riyadh Rawdah	966596022664	Mariam Khoury	0	pending
24.72753902	46.79367473	Saudi Arabia Riyadh Rawdah	966596012452	Eman Aljohani	0	pending
24.74028747	46.83576469	Saudi Arabia Riyadh Rawdah	966542122664	Mohammed Ali	0	pending
24.75730143	46.70601037	Saudi Arabia Riyadh Rawdah	966596022664	Nour Al-Masri	0	pending

Showing Selected Collection After TSP

Latitude	Longitude	Location	Mobile	Name	Order	Status
24.72753902	46.79367473	Saudi Arabia Riyadh Rawdah	966596012452	Eman Aljohani	1	pending
24.74028747	46.83576469	Saudi Arabia Riyadh Rawdah	966542122664	Mohammed Ali	2	pending
24.77988114	46.81456637	Saudi Arabia Riyadh Rawdah	966542532664	Yasmin Al-Haddad	3	pending
24.79125857	46.76907975	Saudi Arabia Riyadh Rawdah	966596025236	Ahmed Mansour	4	pending
24.75730143	46.70601037	Saudi Arabia Riyadh Rawdah	966596022664	Nour Al-Masri	5	pending
24.68907834	46.744638	Saudi Arabia Riyadh Rawdah	966596022664	Rami Farid	6	pending



FINAL DESIGN DETAILS - ICS

```
app.get("/tspCalculate", isAdmin, async function (request, response) {
  const result = "No results.";
  let listToCalculate = [[24.731695254145624, 46.77555817340259]];
  try {
    const collectionListSnapshot = await db.collection("collectionList").get();
    let documentNames = [];
    collectionListSnapshot.forEach((doc) => {
      documentNames.push(doc.id);
    });

    const collectionNameSearch = request.query.collectionNameSearch;
    if (!collectionNameSearch || !documentNames.includes(collectionNameSearch)) {
      response.render("tsp.ejs", { documentNames: documentNames, collectionDetails: [],
        collectionDetailsUnordered: [], collectionDetailsOrdered: [], resultText: result });
    } else {
      const collectionSnapshot = await db.collection(collectionNameSearch).get();
      let collectionDetails = [];
      collectionSnapshot.forEach((doc) => {
        const data = doc.data();
        collectionDetails.push(data);
        listToCalculate.push([data.lat, data.lng]);
      });

      const [allTimes, orderedLocations] = await runTSP(listToCalculate);
    }
  }
});
```



FINAL DESIGN DETAILS - ICS

```
function runVRP(inputCollection) {
  return new Promise((resolve, reject) => {
    let outputData = '';
    const housesJSON = JSON.stringify(inputCollection);
    const pythonProcess = spawn('python', ['vrp.py']);

    pythonProcess.stdin.write(housesJSON);
    pythonProcess.stdin.end();

    pythonProcess.stdout.on('data', (data) => {
      outputData += data.toString();
    });

    pythonProcess.stderr.on('data', (data) => {
      console.error(`stderr: ${data}`);
      reject(data.toString());
    });

    pythonProcess.on('close', (code) => {
      if (code === 0) {
        try {
          const output = JSON.parse(outputData);
          resolve(output);
        } catch (error) {
          reject(error);
        }
      } else {
        reject(`Python process exited with code ${code}`);
      }
    });
  });
}
```

```
function runTSP(inputCollection) {
  return new Promise((resolve, reject) => {
    let outputData = '';
    const housesJSON = JSON.stringify(inputCollection);
    const pythonProcess = spawn('python', ['tsp.py']);

    pythonProcess.stdin.write(housesJSON);
    pythonProcess.stdin.end();

    pythonProcess.stdout.on('data', (data) => {
      outputData += data.toString();
    });

    pythonProcess.stderr.on('data', (data) => {
      console.error(`stderr: ${data}`);
      reject(data.toString()); // Reject with the error message
    });

    pythonProcess.on('close', (code) => {
      if (code === 0) {
        try {
          const output = JSON.parse(outputData);
          const allTimes = output.allTimes;
          const orderedLocations = output.orderedLocations;
          resolve([allTimes, orderedLocations]);
        } catch (error) {
          reject(error);
        }
      } else {
        reject(`Python process exited with code ${code}`);
      }
    });
  });
}
```



FINAL DESIGN DETAILS - ICS

```
// Update order fields in documents based on ordered locations
await Promise.all(orderedLocations.slice(1).map(async (index, i) => {
  const docRef = await db.collection(collectionNameSearch)
    .where('lat', '==', listToCalculate[index][0])
    .where('lng', '==', listToCalculate[index][1])
    .get();
  if (!docRef.empty) {
    const docId = docRef.docs[0].id;
    await db.collection(collectionNameSearch).doc(docId).update({ order: i + 1 });
  }
}));

const resultText = "Nodes in order:\n" + orderedLocations + "\nDelivery Times:\n" + allTimes;
const collectionSnapshot2 = await db.collection(collectionNameSearch).get();
let collectionDetails2 = [];
collectionSnapshot2.forEach((doc) => {
  const data = doc.data();
  collectionDetails2.push(data);
});

response.render("tsp.ejs", {
  documentNames: [collectionNameSearch],
  resultText: resultText,
  collectionDetailsUnordered: collectionDetails,
  collectionDetailsOrdered: collectionDetails2
});

} catch (error) {
  console.log(error);
}
});
```



STANDARDS AND CODES.

ISO and SOC compliance

All Firebase services (aside from App Indexing and Vertex AI for Firebase) have successfully completed the ISO 27001 and SOC 1, SOC 2, and SOC 3 evaluation process, and some have also completed the ISO 27017 and ISO 27018 certification process. Compliance reports and certificates for Firebase services governed by the GCP Terms of Service may be requested via the [Compliance Reports Manager](#)

Service name ▾	ISO 27001	ISO 27017	ISO 27018	SOC 1	SOC 2	SOC 3
Cloud Firestore	✓	✓	✓	✓	✓	✓
Cloud Functions for Firebase	✓	✓	✓	✓	✓	✓
Cloud Storage for Firebase	✓	✓	✓	✓	✓	✓
Firebase A/B Testing	✓			✓	✓	✓
Firebase App Check	✓			✓	✓	✓
Firebase App Distribution	✓			✓	✓	✓
Firebase Authentication	✓	✓	✓	✓	✓	✓

ISO 27001: International standard for information security management.

ISO 27017: Guidelines for cloud security.

ISO 27018: Focuses on privacy in cloud computing.

SOC 1: Controls for financial reporting.

SOC 2: Controls for security, availability, processing integrity, confidentiality, and privacy.

SOC 3: General-use report summarizing controls for security, availability, processing integrity, confidentiality, and privacy.

ISO 26262 specifies a safety lifecycle framework that includes stages such as idea, product development, manufacturing, operation, and decommissioning.

Route Sense must follow this paradigm and incorporate functional safety actions at every level of the concept delivery car's lifespan.



Bill of Material and Budgeting

As this is a software project, the only costs that may be incurred is if the project was scaled to handle thousands of drivers. The free credits and the use of Firebase are currently free and do not require further subscriptions.

For firebase, the normal read and write for a single user is around 100-1000 as such, after the free 50,000 reads and 20,000 writes, there will be a price point of 0.031 per 100,000 reads and 0.094 per 100,000 writes, the price scales up heavily with the addition of more users.

For google maps, the price is increased, when there are more users a consideration of using an alternative map SDK arises.



MEETING SPECIFICATIONS

Does not meet

- 01** App missing some map functionality
- 02** Environmental features need to be implemented
- 03** Individual drivers cannot sign up yet

Meets

- 01** Application lists deliveries in order
- 02** Connected to database and syncs in real-time
- 03** Website has the required features for further development
- 04** Easy to use



MULTIDISCIPLINARY ROLES ICS

NAWAF ALJOHANI

ICS

Application interface and design

User interface and functionality

Web portal

Frontend, backend and database

Database

Design and choice of database

Authentication

and TSP,VRP Integration

Server.js

Backend Node.js, EJS view engine, Firebase

Muath Albelaihed

ICS

Application User interface and functionality

Profile, car+

Frontend application

Database configuration

Firebase database

Implementation of TSP & VRP



MULTIDISCIPLINARY ROLES ISE

Yazeed Aljathlan - Yazeed Ababtain

ISE

Define Variable for the TSP and VRPTW model

Define objective for the TSP VRPTW model

Formulate constraints for the TSP VRPTW model

Convert the formulation to a code and use a solver to gain exact optimal answer

Develop a plan to offer an integrated solution (introducing VRP and TSP with time windows as a package)

Test the model

Monitor the prototype against specifications



MULTIDISCIPLINARY ROLES ME & CE

Faisal Almulhem

ME

Car+ design

Environmental impact

Optimal Car Design

Car+ information

Fuel cost analysis

Saeed Abu Msami

CE

Utilize ArcGIS for simulation

Traffic Impact Assessment

Environmental Impact

Infrastructure Planning

Distance Matrix Analysis

Team 36
Route Sense

24Slides.com

Thank you



**Route
Sense**