TurboTrucks: Delivery System Project

SOEN 343 – Software Architecture and Design

Phase 1 Submission

Dr. Joumana Dargham

Team Code Ninja

Andra Vele – 40175695

Jonathan Della Penta – 40128210

Julie Makary – 40243160

George Ezzat – 40245502

Thursday, September 26th, 2024

Table of contents

I – Project Definition

Objectives

Starting the project, the main objective that was identified through reading the requirements was to create a "Delivery" service application (or website) that would be user-friendly and efficient for the user's needs. In making such a product, its goal would be to streamline the integrate the following list of elements that would target the user's needs. The delivery service should be able to:

1. Create delivery quotes and inform the user of how expensive the service would be.
2. Allow communication between the user and the service provider.
3. Handle delivery requests such as specific pick up and drop off times.
4. Track orders either for shipping out or waiting for a package.
5. Handle payments whether there is a pending balance or the user needs to pay shipping fees in or out.
6. Allow the user to use a Chatbot for basic application or communication purposes.

As a requirement of the project, each team was tasked to come up with a feature unique to their project. The one that this team has chosen was the ability to change the person who will pick up the package as a setting within the app. This will allow the person to designate someone else to pick up the package in their place.

Defining Method of Approach

As this project requires building a website and having some kind of database, working with Agile methodology is the best practice as things need to be able to change and adapt quickly. Each team member will have a main role that they need to fulfill, which will be based on their expertise to make sure the work gets done within the set deadlines. However, since this is a very collaborative project, allowing team members to collaborate on more difficult tasks is key. Additionally, clear deliverables will be set for each sprint to stay on track throughout the development.

Elements and Deliverables of the Project

As detailed within the course outline and the syllabus, each sprint will be broken down into key deliverables:

- **Sprint 1:** Problem definition, context diagram, and domain model.
  This current phase allows the team to focus on starting the project. It requires a definition of the project scope, creating the context diagram to identify external entities, and outlining the domain model.
- **Sprint 2:** System architecture with sequence diagram (due Oct 19).
  This phase will focus on the overall system architecture and detailing interaction flows.
- **Sprint 3:** Implementation of four core features with GUI (due Nov 15).
  This second to last phase will involve building and demonstrating the website's core functionalities, including user interfaces, database interaction, and user experience.
- **Sprint 4:** Final presentation, project report, and peer review (due Nov 30). This final sprint will involve the complete project presentation and submission of the project report, including a review of each team member's contribution.

Project scope

The project's overall scope is to work on the main 6 features of the website, which have been listed above, and have those functional. The goal is to create the product so that it can be scalable to different needs and follow different design principles. The system will focus on making the user experience simple and intuitive as that has been a problem with other systems already on the market. Even though the time is limited for this project, the team aims to deliver a product that will be reliable and an efficient solution for the problems that are present.

II – Problem Definition

What is the problem?

Normally, when shipping services are required, a person must either drop off their package at a mailbox or a post office. Additionally, a shipping label must be created, printed, and placed onto the package. For most people, this can grow to be a long and meticulous process, especially for those who do not have easy transportation to a post office or quick access to a printer.

There is often little to no communication between the client and the courier after the package is picked up.

Finally, delivery tracking can sometimes be unreliable, leaving clients with no precise idea of the status of their package.

How did the problem emerge?

These problems emerge from the client's lack of access to certain resources, like transportation (car, bus, taxi), time in their schedule, or a printer. Even if someone does have access to a car, they may live very far to the nearest post office which requires them to sacrifice their personal schedule to make a trip to drop-off the package.

What is your solution?

Our solution is a delivery service that comes to you. You can organize a designated pick-up location and date to meet with a courier who will accept (or reject) the package. Once the package is secured, the courier will attach any shipping labels with a printer that is onboard every vehicle, as well as immediate set up of precise package tracking.

If the package is to be shipped internationally, the client has up until shipping to fill out a customs questionnaire on the delivery service application. Once completed, the delivery service will use the given information to generate and attach an international shipping label to the package.

What are the advantages of your solution in comparison with the existing solutions?

The main advantage from our solution is the time and inconvenience saved from dropping off a package. If someone lives far from a post office, doesn't have access to a car, or simply doesn't have the time in their schedule to drop off a package, our delivery service helps by saving that trip for them.

Additionally, the live chatbot allows clients to easily communicate with the courier and give quick response times to any questions or concerns that they may have.

Also, our service saves people the task of having to print the labels themselves, especially if they don't have quick access to a printer.

To conclude, our solution saves people time and simplifies the shipping process, while also providing the clients with quick and responsive support.

III — Technology Used

Team collaboration

For the team's communication, Discord and WhatsApp were the apps used. Additionally, GitHub will be used for collaborating on code portions of the project. Along with that, some sort of card system will be used, and as GitHub has a good platform, that will be prioritized. Dividing the work by epics and then creating stories shown as cards on the board will allow all members to work equally on the project.

Monitoring and verification

The team will be using GitHub to pull and push modifications to the code. As it shows the amount of time one works on the code and what modifications have been done, it would be easy to monitor everyone's work. Also, creating different branches will allow for verification of implementation before committing changes to the final version.

Design and modeling work

For designing and modeling work, websites such as Draw.io or Lucidchart will be used. Those are simple to use and the team is familiar with at least one of those platforms.
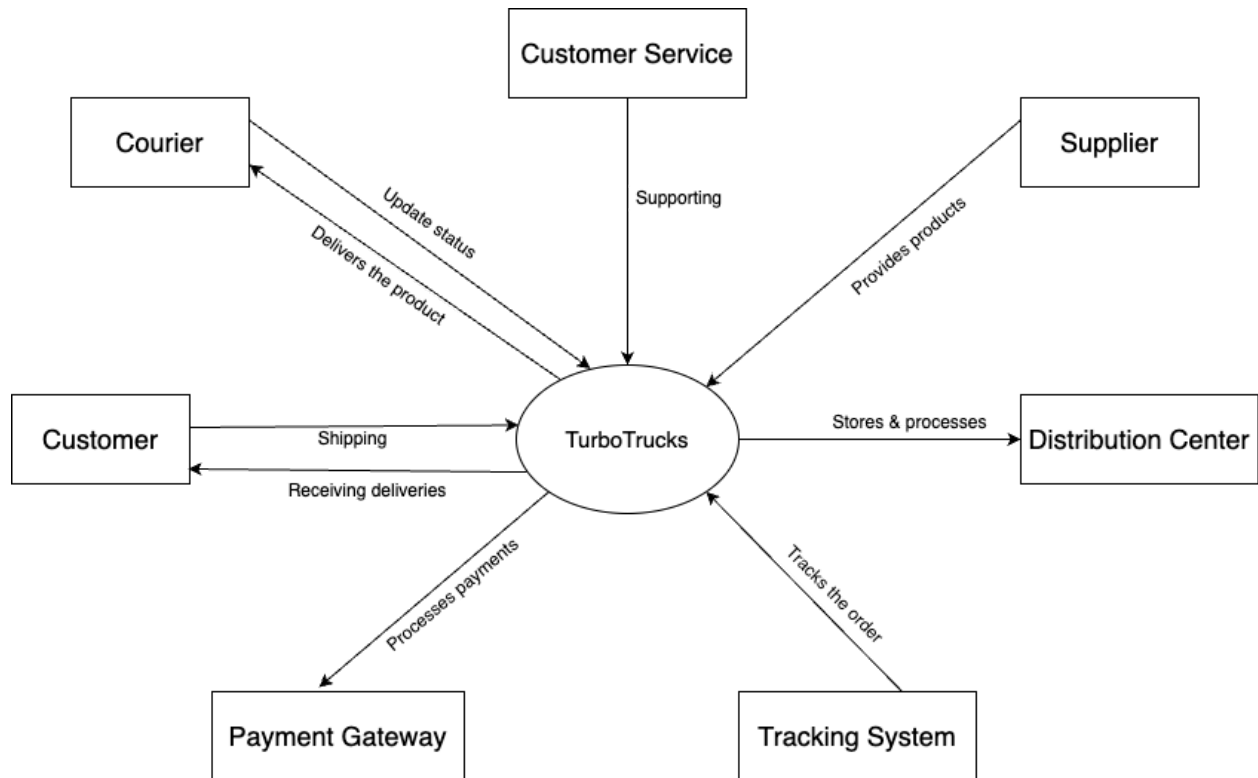
Interface

The creation of the interface will mainly be done via Figma as some of the team members are familiar with the platform. Additionally, there can be a rendition of the website that could be created on platforms such as WordPress, but that would not be part of the priorities.

Coding

The chosen languages for the website have been narrowed down to the following options:
- HTML and CSS will be used for the front end.
- JavaScript or Java will be used for the back end. (this remains to be discussed by the team for a final decision.)
- SQLite will be the main database that will be used.

IV – Context diagram

V – Domain model

Customer
- -Name: String
- -contactInfo: String

ChatBot

Delivery
- -pickupLocation: String
- -dropoffLocation: String
- -orderNumber: int
- -status: String

orderTracking
- -trackingID: int

ServiceCommunication

Quotation
- -estimatedCost: double
- -quotationId: int

paymentInfo
- -amount: double
- -paymentMethod: String