WIZTOCK

Software Design Specification

07.05.2021

150117062 Ahmet Tunahan Cinsoy
150115851 Enver Aslan
150116021 Mikail Torun

Prepared for
CSE3044 Software Engineering Term Project

Table of Contents

1. Introduction

1.1. Purpose

Many companies record and monitor their product purchases, sales and storage processes on their computer by using desktop applications. These applications enable companies to define their own products, determine their characteristics, keep records of products that have been purchased and sold, as well as in which warehouse the products are stored and instantly follow-up the transfer processes of the products.

With the development of technology, companies want to access such information instantly from anywhere. Because of that, in this project, we have aimed to respond to these requests of the companies..

1.2. Statement of scope

The scope of the project is a system that a company can define a product cards and that the information of the location and movement of these products can be kept and that commercial information of suppliers and buyers or customers can record by using web interface basically. This system is not a detailed accounting program. It is mostly aimed to follow the commercial movements of the products and companies.

1.3. Software context

We have aimed to develop a web application that enables registered companies to track their stocks. With the help of this application, companies will have a better understanding of their stock units.

1.4. Major constraints

We can simply state the major constraint as registered user has to have a computer that has access to internet to use the application. Additional constraints will be explained at section 2.

1.5. Definitions, Acronyms and Abbreviations

- A **product card** is a category that contains information about the original product. Firms acquire their products based on their product cards.
- **Companies (End-Users**) are the entities that have an active registration in Wiztock.
- **Customers** (**Individual or Corporate)** are the group of people or firms that end-users sell their items to.
- **Warehouses** are the storage houses for the products of end-users.
- **Suppliers** are companies which end-users buy product.
- **Stock** simply identifies the quantity of a product, i.e., the status of an item in warehouse.

1.6. References

- https://y.ebsyazilim.com:9090/depostok.dll?durum=demo
- https://uygulama.mobiliys.com/login
- https://panel.isbasi.com/giris-yap
- https://www.parasut.com/

2. Design Consideration

This section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution.

2.1. Design Assumptions and Dependencies

This section describes assumptions or dependencies regarding the software and its use. These may concern such issues as:

- It is accepted that the clients have sufficient ability of utilizing web program.
- The application requires the PC to be associated with the web.
- Any working framework is different conditions for end-client and engineers.

2.2. General Constraints

In this section, the general restrictions, and constraints that the application has and has a significant impact on the design of the project have been defined.

- System needs any web browser like Google Chrome, Mozilla, Internet Explorer
- Developers need back-end and front-end development technologies like Editor for PHP language, any server such as wamp-server.
- For end-user, any computer, tablets, or smartphone which is connected to internet is needed.
- For developers, any computer that holds their project file and database is needed.
- The back-end service must be running to perform requested tasks.
- There is a need to have a database system to store end-user information.
- The secure information about end-users must be stored encrypted in database.
- Memory and other capacity limitations depend on the properties of the web server that the system has built on.
- Wiztock delivers its services based on the approach of best effort.
- The network communication is provided by HTTP POST requests while data are being sent to the server.
- From the perspective of quality, responsive and elegant UI is aimed to deliver to end-users.

2.3. System Environment

- HTML5
- Bootstrap 3 / Bootstrap 4
- Javascript (React, JQuery, Chart.js etc.)
- Web Server to store project files and database.
- PHP
- Database Management System (MySQL, / MongoDB, / MariaDB)

2.4. Development Methods

During the creation of the application, iterative development methods will be performed to make sure that components are being created according to the specifications. It is aimed to build a development cycle based on scrum meetings with weekly sprints.

3. Architectural and component-level design

3.1. System Structure & Description of Components

- **Login Page:** Simple UI to provide login functionality of the application.
- **Dashboard:** Summarize product information view about the entire application.
- **Profile Settings:** The page that shows the profile info about end-users.
  - Categories: Page that company defines its products.
  - Employees: Employees of the company.
  - Company Information: Brief information about company.
  - Warehouse: Location that keeps product stocks
- **Products:** The web page that shows the products of the company.
  - Selling: The page that shows products which are being sold.
  - Buying: The page that shows products which are being bought.
  - Transfer: The page that shows products which are being transferred between warehouses.
- **Customers:** The web page that shows the customers of the user.
  - Add Customer: Functionality that enables company to add customer.
  - Contact Detail: If customer is corporate, details of contacts will be shown under this page.
  - Add Contact: If customer is corporate, end-users can add contact person for their corporate customers.
- **Suppliers:** The web page that shows the suppliers of the user.
  - Add Supplier: Functionality that enables company to add supplier.
  - Supplier Detail: If supplier is corporate, details of suppliers will be shown under this page.
  - Add Contact: If supplier is corporate, end-users can add contact person for their corporate suppliers.

3.1.1. Architecture diagram

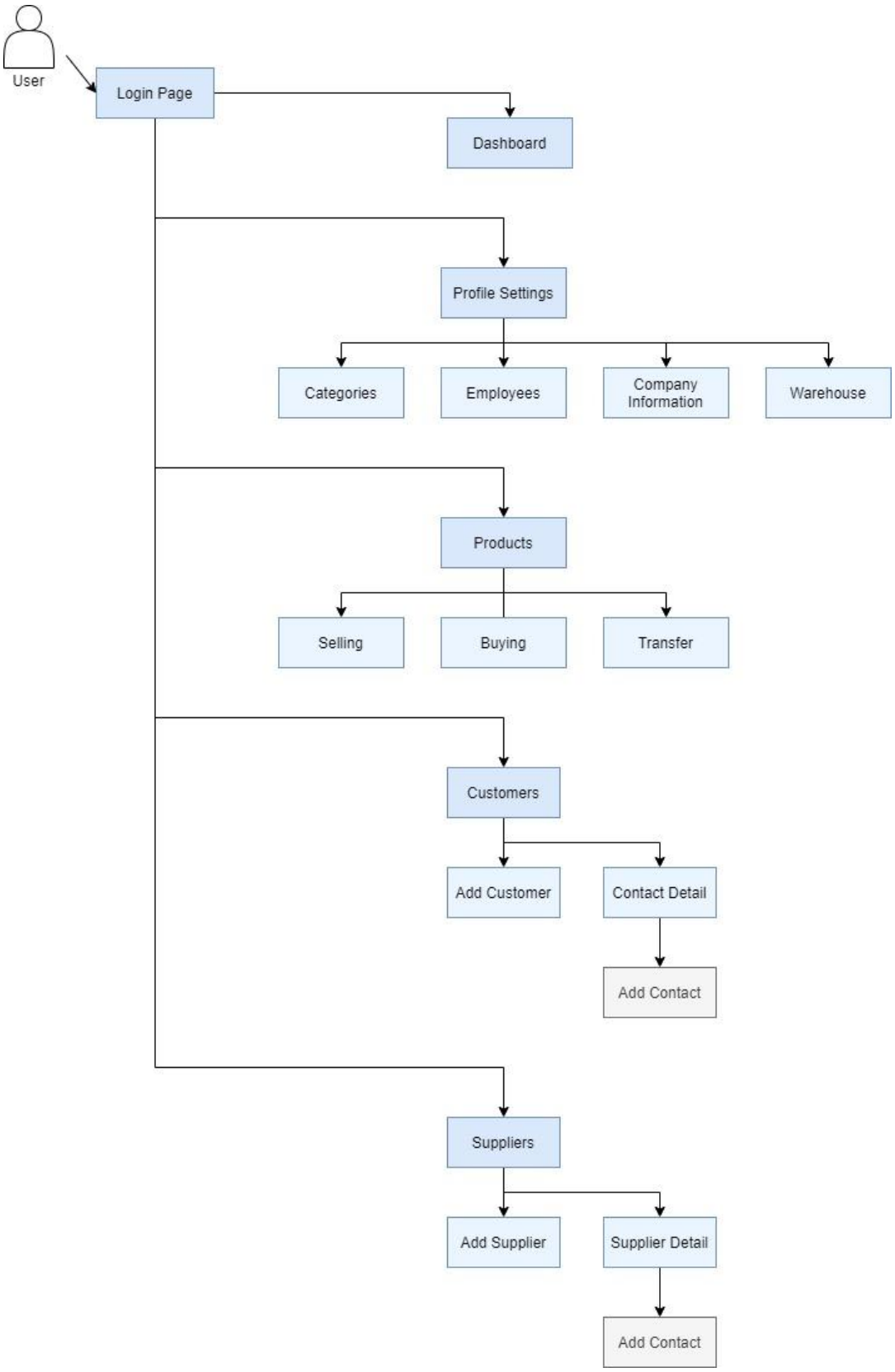

Figure 1- System Architecture Diagram

3.1.2. Entity Relationship Diagram

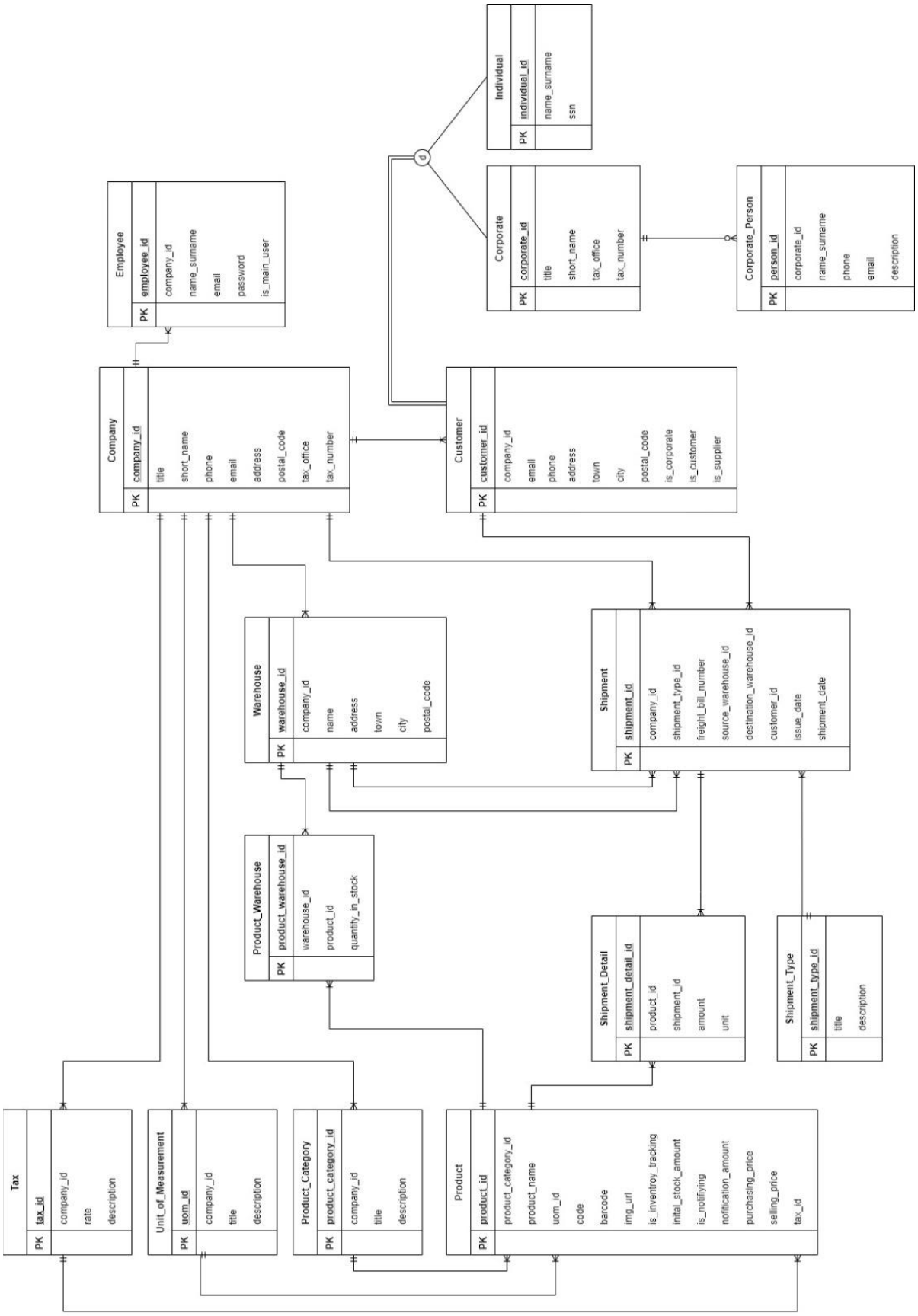

Figure 2- Database Entity Relationship Diagram

3.1.3. UML Class Diagram

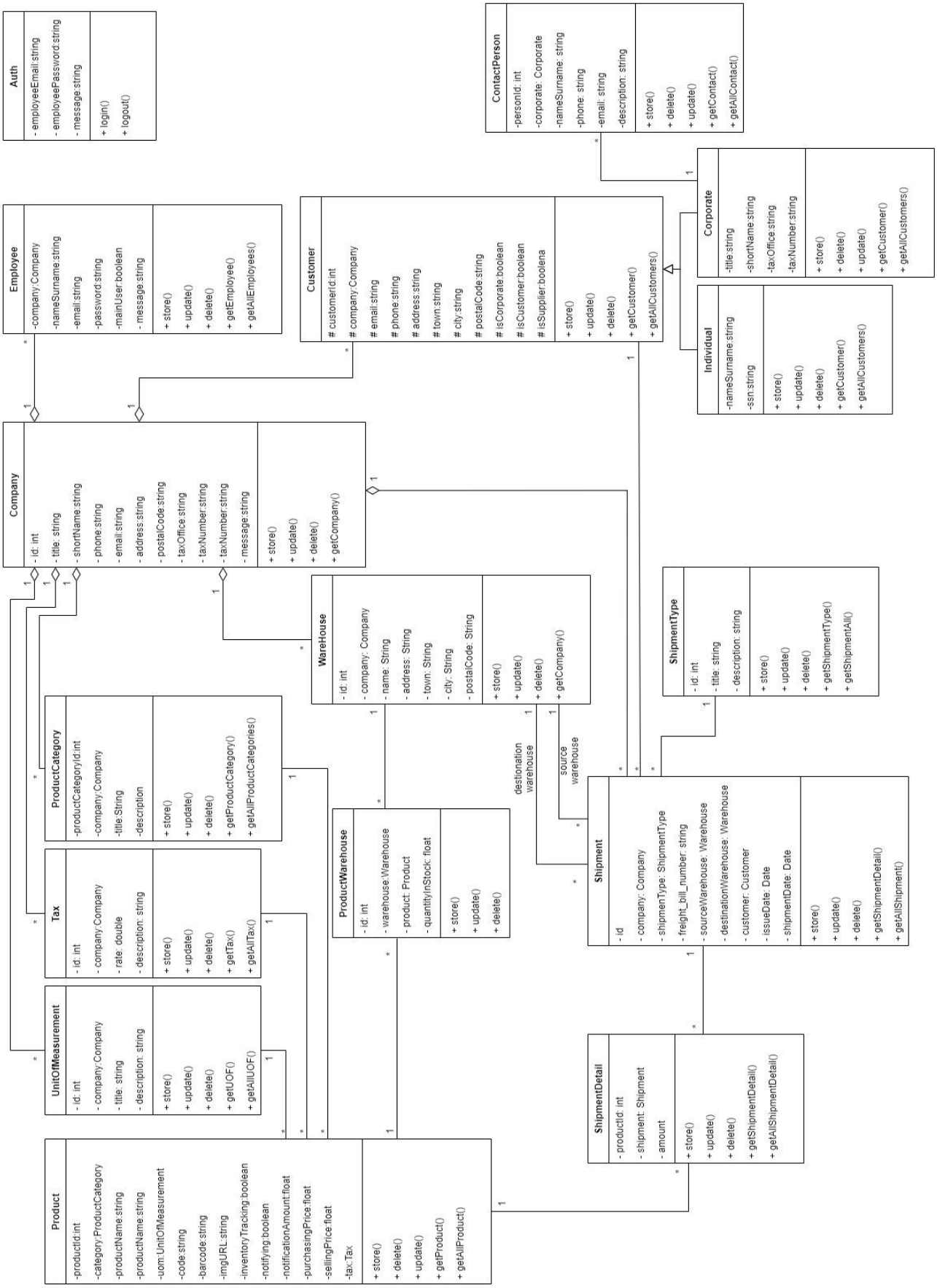

Figure 3- UML Diagram

3.2. Domain Sequence Diagrams

Some of the Sequence Diagrams are shown below. According to our needs, diagrams will be updated throughout the development process.

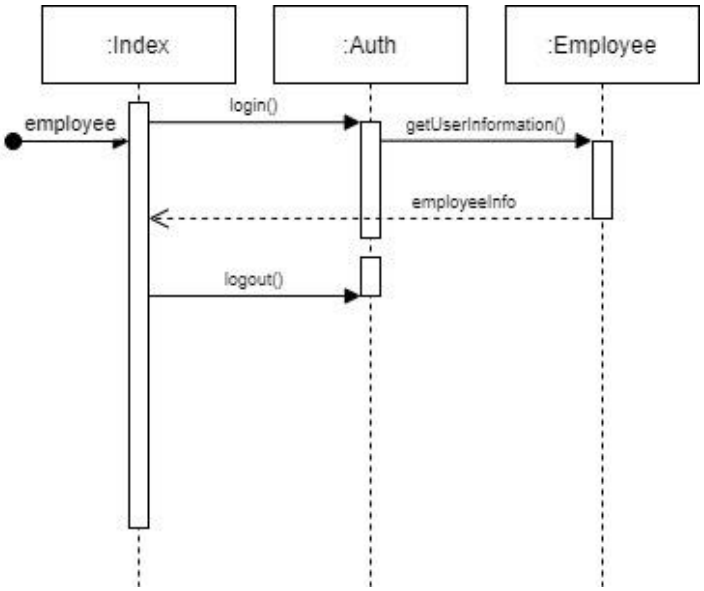

Figure 4 – Authentication Sequence Diagram

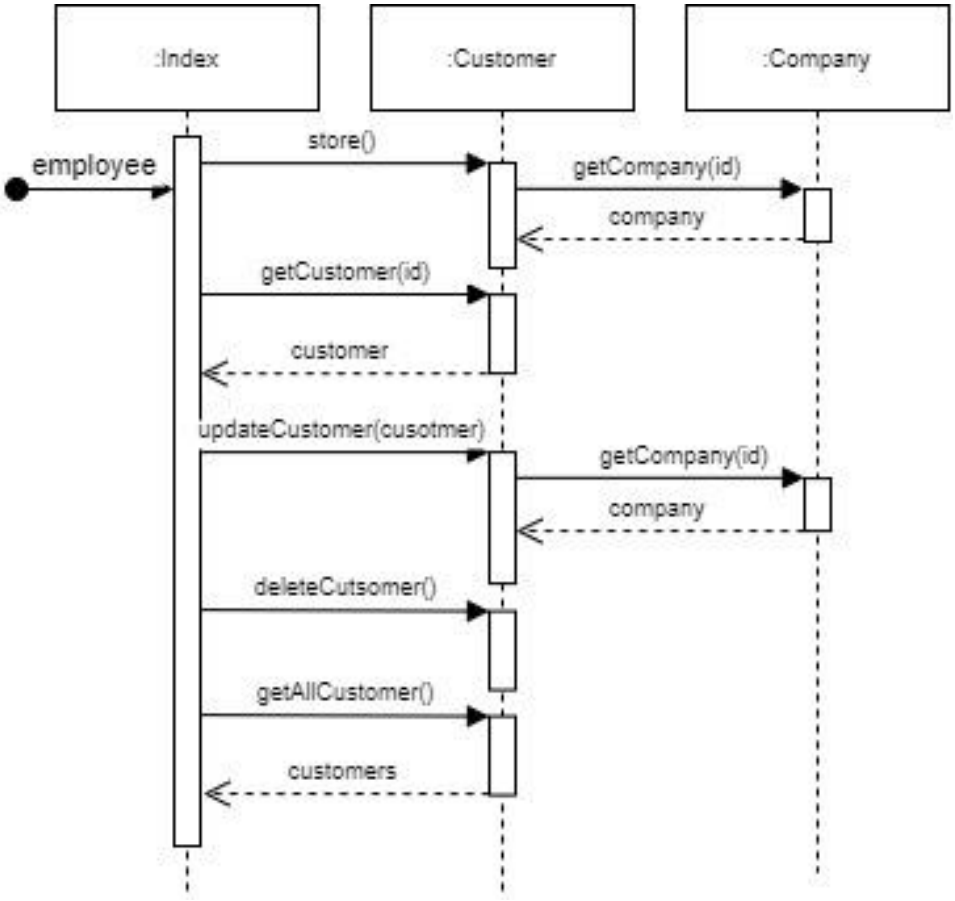

Figure 5 - Customer Sequence Diagram

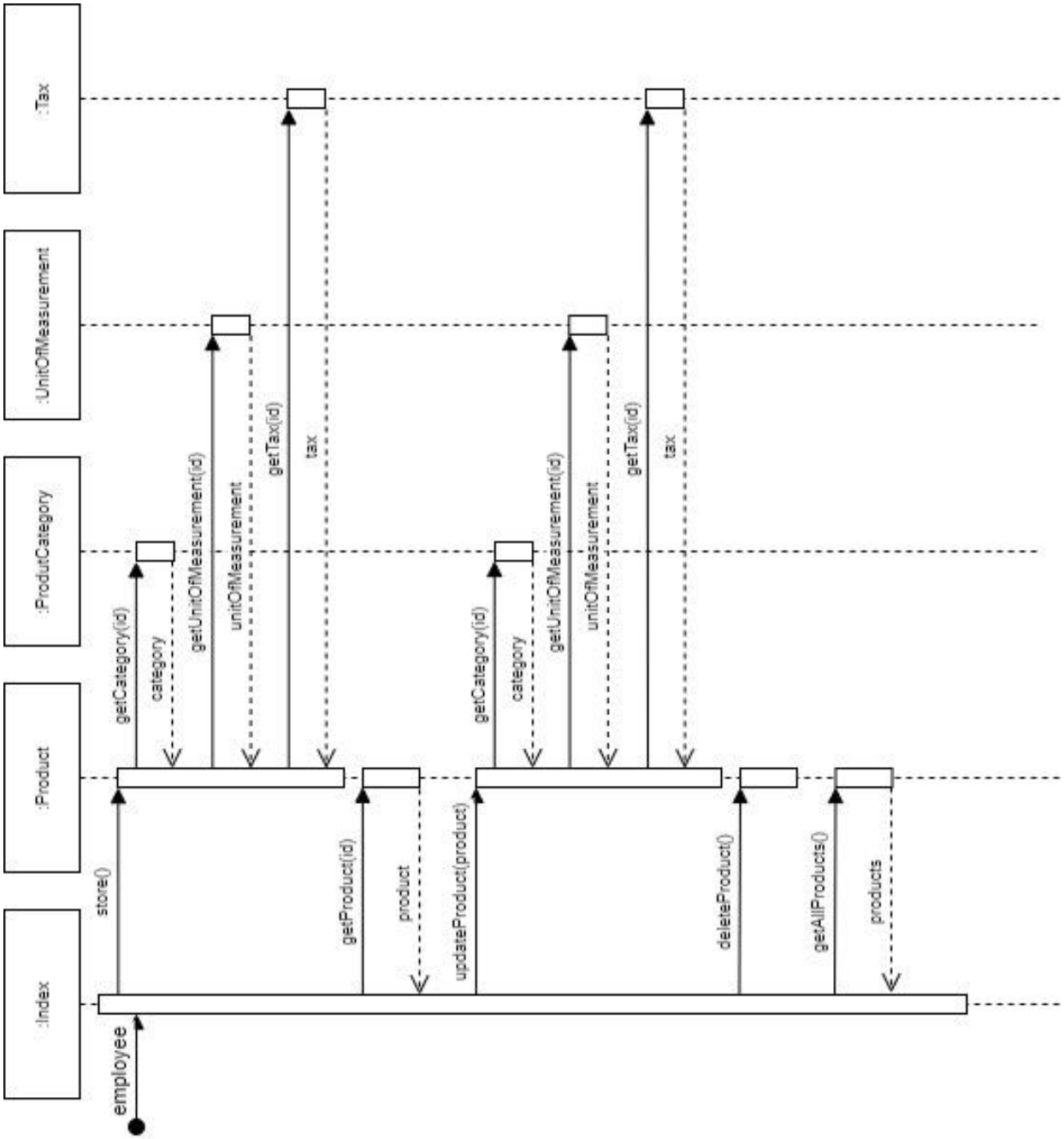

Figure 6 - Product Sequence Diagram

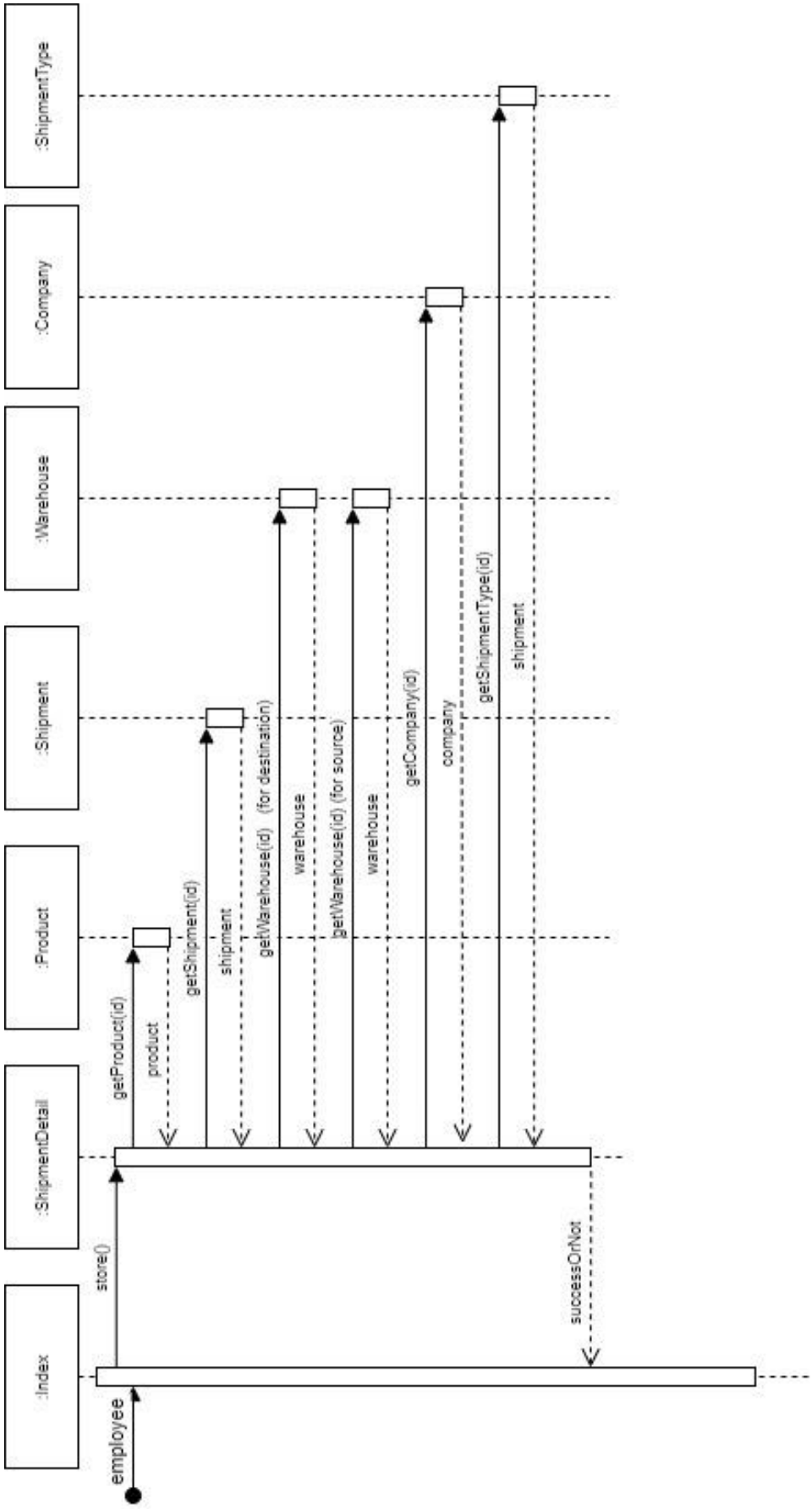

Figure 7 - Product Shipment Sequence Diagram

4. Conclusion

In conclusion, the contrived web application will come into use in every device which have a connection to the internet also have a web browser. Thanks to its user-friendly interface, there will not be any need for pre-using education on the end-user side.

5. Division of Labor

While we are implementing this report, we have arranged multiple Google Meet and Zoom sessions to discuss content. All the team members have joined this meeting, division of labor have been executed successfully.

Team Members	**Division of Labor**
Ahmet Tunahan Cinsoy	Equal load of work
Enver Aslan	Equal load of work
Mikail Torun	Equal load of work