

**CEN 308 SOFTWARE ENGINEERING**

PROJECT DOCUMENTATION

INVENTORY TRACKING SYSTEM

Prepared by:

**Amela Košpo**

**Benjamin Krehić**

Proposed to:

**Nermina Durmić, Assist. Prof. Dr.**

**Aldin Kovačević, Teaching Assistant**

21.6.2021.

Table of Contents

[1. Introduction 3](#_Toc75191991)

[1.1. About the Project 3](#_Toc75191992)

[1.2. Project Functionalities and Screenshots 3](#_Toc75191993)

[2. Project Structure 6](#_Toc75191994)

[2.1. Technologies 6](#_Toc75191995)

[2.2. Database Entities 6](#_Toc75191996)

[2.3. Architectural Pattern 7](#_Toc75191997)

[2.4. Design Patterns 7](#_Toc75191998)

[3. Conclusion 7](#_Toc75191999)

# 1. Introduction

For our project for Software Engineering, we decided to make an Inventory tracking system. Every company that sells physical merchandise needs to have a system that helps them keep track of many things. They would need to keep track of their stock of each product: how many of each product they have, do they need to order more, do they have enough for a specific order, …

They would also need to track which company supplies them with which product in stock, or materials for a product that they manufacture.

They need to track each order and the customer to whom it’s sold to.

## 1.1. About the Project

Our project has the following functionalities:

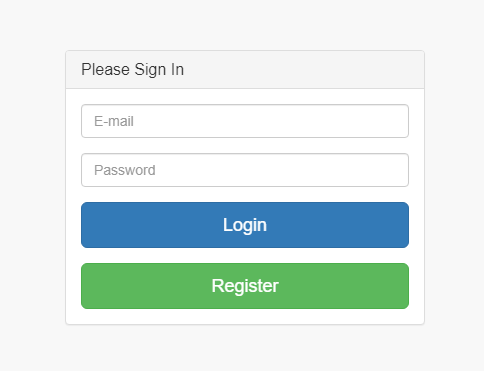
* Fill out an order with which customer it’s going to, and which products in what quantity they are buying
* View all orders ever filled out
* Add a new product into stock
* Sell a product from stock
* View all products that are currently in stock (along with which supplier supplies it, and how much of it there is in stock)
* Add a new customer
* View all customers
* Add a new supplier
* View all suppliers

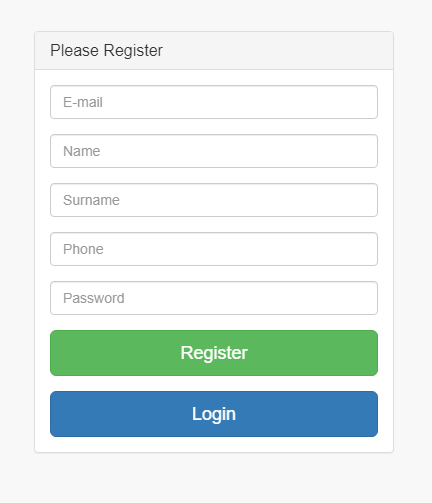
Our project repository is on GitHub, on this link: <https://github.com/krehicbenjamin/SE_project>

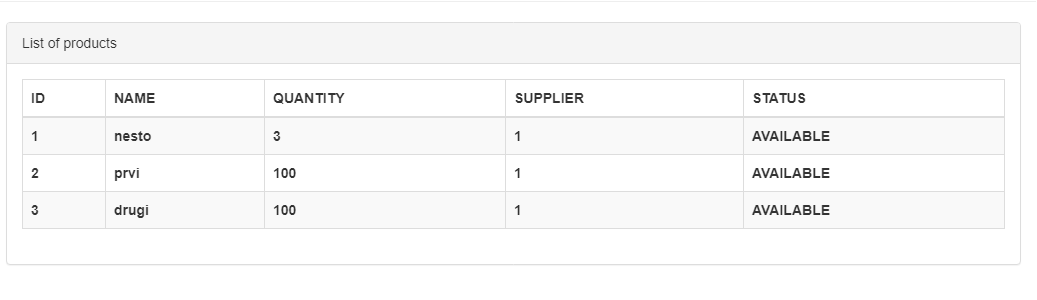
Our project is deployed on Heroku on this link: <http://seproject-amela-benjamin.herokuapp.com>

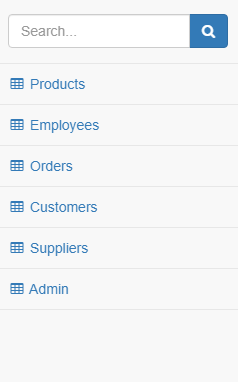
## 1.2. Project Functionalities and Screenshots

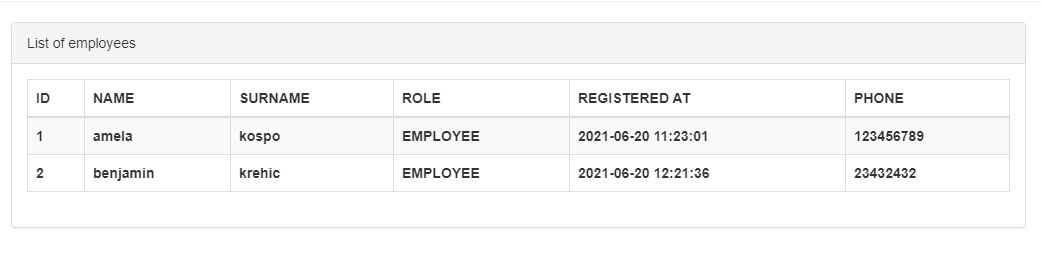
The main features of our application are described in the About the Project section of this report.

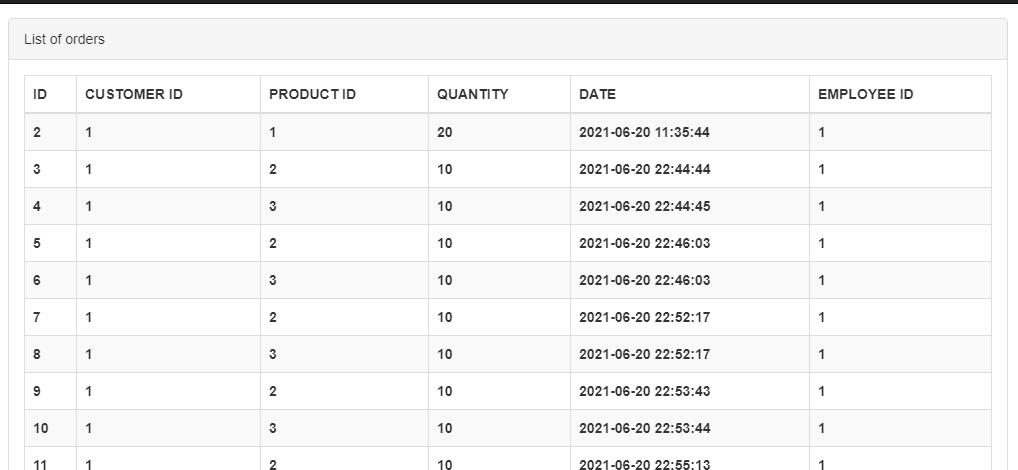


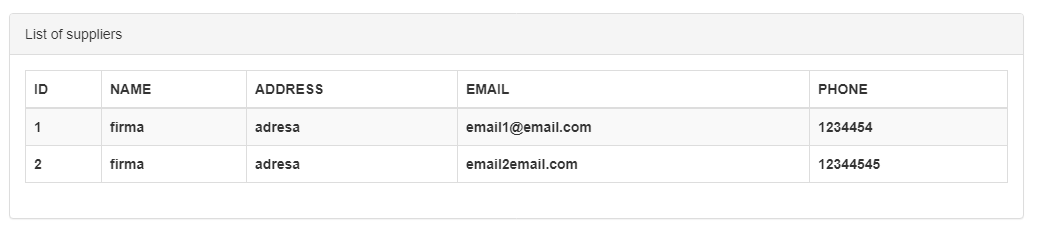












# 2. Project Structure

## 2.1. Technologies

We used php and Flight::php for the backend and API. We also used JSON Web Tokens(jwt) for login and register.

We used a bootstrap template for the frontend and adjusted it to our requirements. The fronted was mad using JavaScript, HTML, and CSS.

For backend we used PSR-12 coding standard.

## 2.2. Database Entities

* employees
* suppliers (the companies that “our company” buys their product to resell from)
* customers
* products
* orders

The product table has a foreign key supplier\_id that refers to the id of the supplier.

In the orders table there is three foreign keys: product\_id (product bought), customer\_id (customer it’s sold to), employee\_id (employee that filled out the order).

## 2.3. Architectural Pattern

We used the layered architectural pattern, because we have the most experience with it and we believed that it would be the most suited for our project.

## 2.4. Design Patterns

* builder pattern: DAO (for access to the database)
* builder pattern: singleton (for the database connection, so that there’s only one static instance of the connection)
* behavioral pattern: composite (for filling out orders with seemingly many different products, but in reality, there is a separate order for each product, only the customer is the same)

# 3. Conclusion

For us, the project was interesting to make, it provided valuable experience for collaborating with other people. So far, mostly all projects we have done for classes were done individually.

The challenging part for us was coordinating with each other about project tasks and our different development environments. We are overall satisfied with how we implemented our project. We do not believe it is perfect, as this was a studying opportunity, and there’s always room for improvement.

If we were to do the project again, or a similar project, it would be much easier, since we know how the workflow would go, and we understand the process better.