T.R.

GEBZE TECHNICAL UNIVERSITY

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

RECOGNITION OF ITEMS ON CASHIER CHECKOUT USING MACHINE VISION

SEDEF ERDOĞDU
EMINE SULTAN SAVRAN

SUPERVISOR
ASSOCIATE PROF. DR. HABIL KALKAN

GEBZE
2023

T.R.
GEBZE TECHNICAL UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

RECOGNITION OF ITEMS ON CASHIER CHECKOUT USING MACHINE VISION

SEDEF ERDOĞDU
EMINE SULTAN SAVRAN

SUPERVISOR
ASSOCIATE PROF. DR. HABIL KALKAN

2023
GEBZE

GEBZE TECHNICAL UNIVERSITY	GRADUATION PROJECT JURY APPROVAL FORM

This study has been accepted as an Undergraduate Graduation Project in the Department of Computer Engineering on 22/06/2023 by the following jury.

JURY

Member
(Supervisor) : Associate Prof. Dr. Habil Kalkan

Member : Assistant Prof. Dr. Burcu Yılmaz

ABSTRACT

Manual scanning and product recognition operations performed by traditional methods can be time-consuming and prone to errors. The product recognition system monitors the passage of each product through the safe and automatically registers it in the inventory system. This makes inventory tracking of enterprises easier and improves inventory management. The automatic product recognition system can detect fraud attempts such as fake bar codes or incorrect labeling. At the same time, it can also prevent cases of theft, such as scanning a product more than once. A model is trained using YOLOv5 to detect products passing through the cashier checkout and a web application is developed for users. The goal of the system is to automate the process of recognizing and counting products during the checkout process in retail environments. By training the model on a custom data set, accurate and efficient testing of various supermarket products was ensured. The trained model was integrated into a desktop application that allows users to receive real-time predictions.

Keywords: Recognition items, real-time detection, YOLOv5.

ÖZET

Geleneksel yöntemlerle gerçekleştirilen manuel tarama ve ürün tanıma işlemleri zaman alıcı ve hatalara açık olabilir. Ürün tanıma sistemi, her bir ürünün kasadan geçişini izler ve otomatik olarak envanter sistemine kaydeder. Bu, işletmelerin envanter takibini kolaylaştırır ve envanter yönetimini geliştirir. Otomatik ürün tanıma sistemi, sahte barkodlar veya yanlış etiketleme gibi sahtekarlık girişimlerini tespit edebilir. Aynı zamanda, bir ürünü birden fazla taramak gibi hırsızlık vakalarını da önleyebilir. Kasiyer kasasından geçen ürünleri tespit etmek için YOLOv5 kullanılarak bir model eğitildi ve kullanıcılar için bir web uygulaması geliştirildi. Sistemin amacı, perakende satış ortamlarında ödeme işlemi sırasında ürünleri tanıma ve sayma sürecini otomatikleştirmektir. Modeli özel bir veri seti üzerinde eğiterek, çeşitli süpermarket ürünlerinin doğru ve verimli bir şekilde test edilmesi sağlandı. Eğitilmiş model, kullanıcıların gerçek zamanlı tahminler almasını sağlayan bir masaüstü uygulamasına entegre edildi.

Anahtar Kelimeler: Ürün tanıma, gerçek zamanlı tespit, YOLOv5.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our main advisor, Associate Professor Dr. Habil Kalkan, for his invaluable guidance, patience, and support throughout this project. His professional mentorship has been instrumental in our successful completion of this endeavor. We would also like to thank Assistant Professor Dr. Burcu Yılmaz for serving as the jury member for this project.

Additionally, we would like to extend our heartfelt appreciation to our families. Their patience, understanding, and unwavering support have been indispensable in achieving the completion of this project. Without them, this accomplishment would not have been possible.

Emine Sultan Savran
Sedef Erdoğdu

LIST OF SYMBOLS AND ABBREVIATIONS

Symbol or Abbreviation	:	**Explanation**
YOLO	:	You Only Look Once
CNN	:	Convolutional Neural Network
mAP	:	Mean Average Precision

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

In today's fast-paced retail environments, manual scanning and recognizing products can take a lot of time and can lead to mistakes. Traditional methods of managing and tracking inventory often rely on humans, which can be inefficient and potentially inaccurate. To solve these problems, an automatic system has been developed that uses YOLOv5, an advanced object detection model, to recognize products. This system aims to make the checkout process in retail stores faster and easier by automatically identifying and counting products.

The product recognition system keeps track of each product as it passes through the checkout area and adds them to the inventory system automatically. This makes it simpler for businesses to manage their inventory and improves overall inventory management. Additionally, the system helps prevent fraud by detecting fake bar codes or incorrect labels, ensuring that the checkout process is honest. It also reduces theft by preventing products from being scanned multiple times, thereby increasing security.

To achieve accurate and efficient detection, a custom dataset was created and used to train the YOLOv5 model. The system has been trained on several of supermarket products, allowing it to recognize different items in real-time effectively. The trained model has been integrated into a desktop application, allowing users to receive immediate predictions during the checkout process.

This paper presents the development of an automatic product recognition system that uses the YOLOv5 model for real-time detection of items at the cashier checkout.

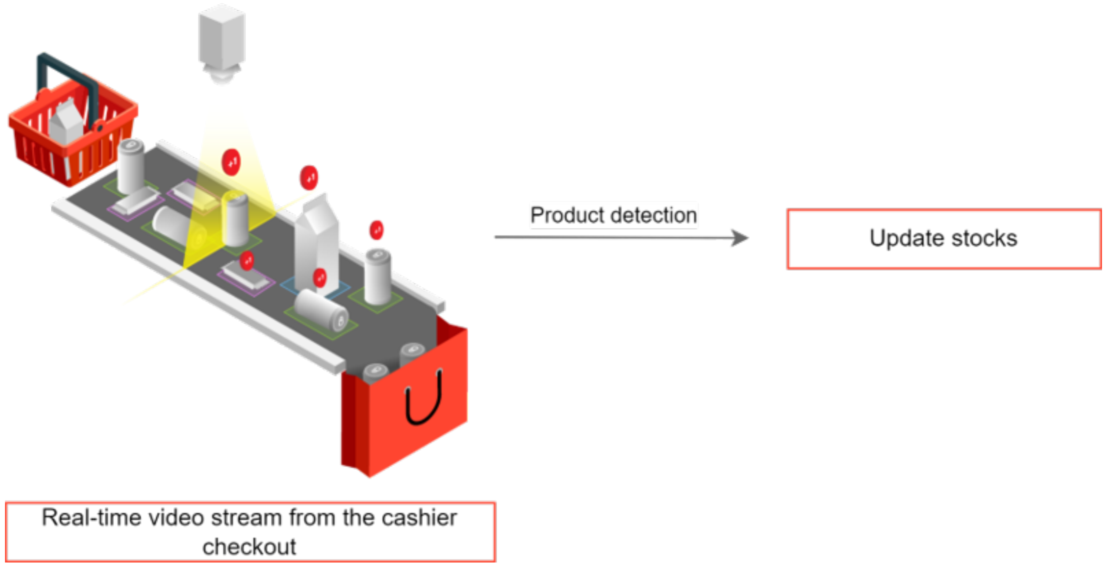

Figure 1.1: Schema of the project.[1]

1.1. Project Definition

The project focuses on a benefit object detection algorithm to enable real-time item recognition at cashier checkouts. The system will utilize a combination of image collection, feature extraction, and classification methods to identify various supermarket products. By analyzing the visual characteristics of the items, the system will accurately recognize them, reducing human errors and enhancing the overall efficiency of the checkout process.

1.2. Goal of The Project

The primary aim of this project is to create a reliable and precise system for recognizing items at cashier checkouts. By automating the process, the system will reduce checkout time, resulting in quicker and more efficient transactions. The system will prioritize accuracy in identifying items, minimizing errors caused by manual scanning or human mistakes.

Furthermore, the system can collect real-time data on sales and inventory, facilitating improved inventory management and restocking decisions.

1.3. Project Plan

1.3.1. Research and Literature Review

In line with the growing trend of companies adopting automation processes, this study contributes to the field by focusing on the development of an automated checkout system for stores using Computer Vision techniques. This study focuses on improving the YOLOv4 algorithm to create an automated checkout system for stores. The system detects objects, counts similar items, generates bills, and enables online payments. It offers benefits such as quick checkout, a user-friendly interface, and customer satisfaction, and promotes social distancing and a safe shopping experience, particularly during the Covid-19 pandemic. [2].

This study focuses on developing an automated inventory system for fast-food restaurants using object recognition, detection, and counting features. The system utilizes Raspberry Pi and the Pi Camera Module as hardware components. The YOLO algorithm is employed for object detection, and custom datasets are trained. A total of 725 datasets are annotated. The system includes a counting algorithm based on the outputs of the YOLO algorithm. The system's performance is evaluated using a confusion matrix, achieving an overall accuracy of 89.55%. The study aims to address the problem of inventory stock mismatches and provide easier access to specific products. [3].

1.3.2. Data Collection and Preparation

In this project, the data are collected by those who prepare the project. For twenty different products, videos are shot at different angles, in light, and then, photos from these videos are obtained as frames. Some preprocessing steps are applied to these photos. The items in the photos are labeled according to the appropriate classes. The LabelImg is used labelling step.

1.3.3. Model Development

The YOLOv5 model is trained for item recognition with the collected photos. YOLOv5 (You Only Look Once) is a popular object detection algorithm known for its speed and accuracy. Google Colab environment is used because it provides high GPU.

1.3.4. Development Applications and Integration

There is two applications: web application and recognition application. The web application has 6 users. Web application provides a user interface for users to view the stock status of the products via backend system. It retrieves the stock information from the database and displays it to the user. Recognition application is responsible for detecting the products in the video stream and updates the stock levels accordingly via backend service. It uses machine vision techniques to identify the products. Camera receives the input from the camera in real time. Database stores the stock information and is updated by backend service. TypeScript Angular for the frontend, Java for the backend and Python for the model is used. PostgreSQL is used for the database.

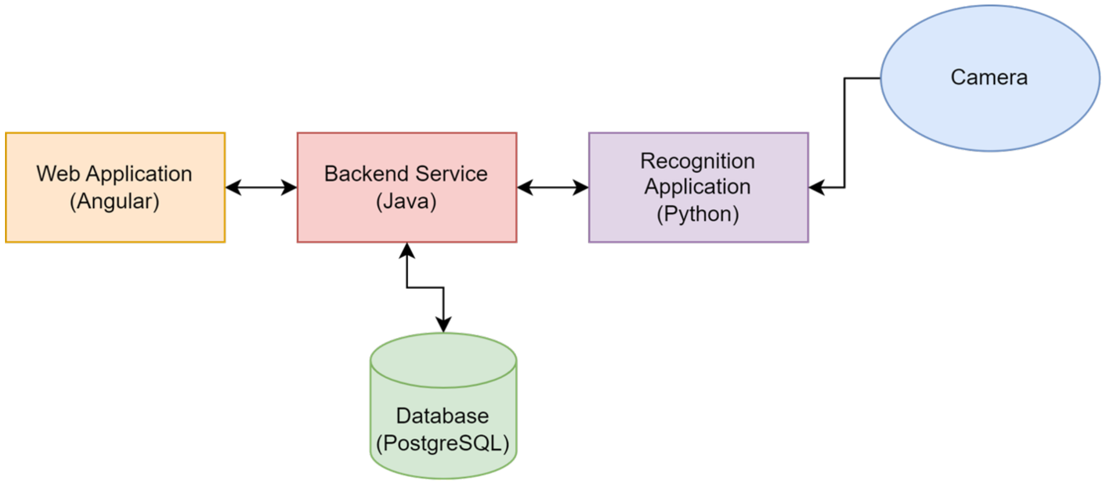

Figure 1.2: The arhitecture of the project applications.

2. IMPLEMENTATION AND EVALUATION OF RECOGNITION APPLICATION

2.1. Dataset Details

There is a custom dataset with twenty classes representing twenty different supermarket products. In the dataset creation part, it is decided which twenty supermarket products will be used first. These twenty products are videotaped from different angles, under different light. Also, the camera is positioned, so that it is above and directly opposite the part where the barcode is being read. When shooting videos, it is treated as if the product is being passed through the cashier checkout. When the video shooting of a product is completed, frames are taken from the videos at certain time periods and photos of the product are obtained. Each of them has a unique ID. Excessively blurred photos are thrown out and these steps are repeated for each product. The first version of the dataset is created in this way.

Figure 2.1: Example photos of the dataset.

In order to be compatible with the YOLOv5 model, the photo is resized to 512x512 pixels. Then, the labeling process begins. Each photo is labeled individually. LabelImg is used for the labeling process. When labeling, the bounding box of the product in the photo is draw. LabelImg gives a txt file containing the bounding box and class information.

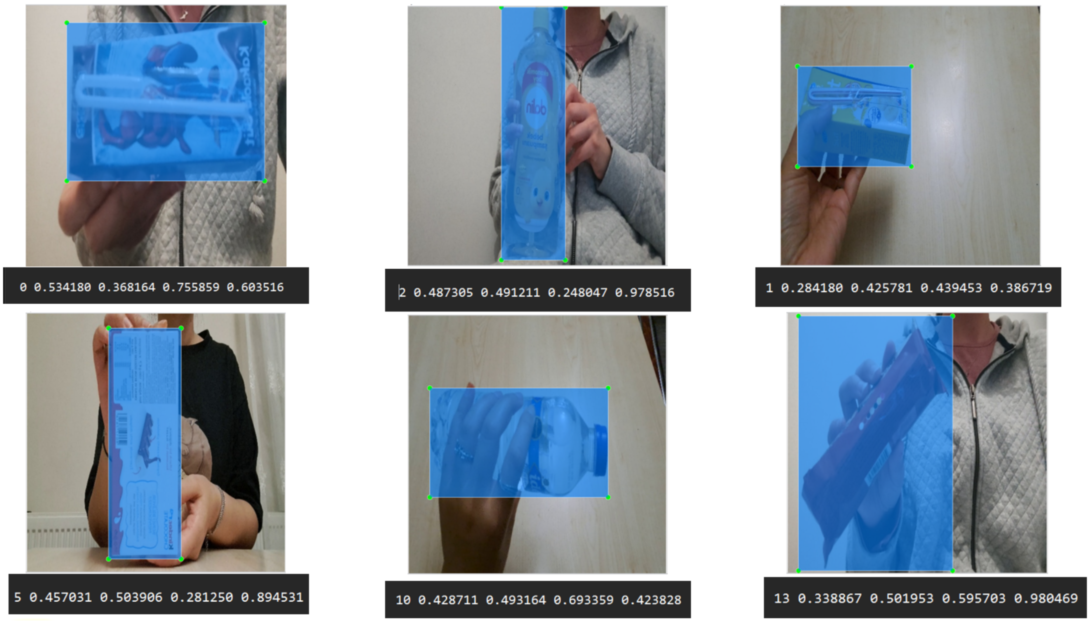

Figure 2.2: Example photos of the labelled dataset.

When the model was trained with the obtained dataset and real-time object detection was performed, the results were not satisfactory. For this reason, new photos taken with a lower-quality camera have been added to the dataset for each class. The information of the created dataset is in Table 2.1.

Table 2.1: The number of photos in the dataset as a class.

Label index	Class name	Total
0	Danone Chocolate Milk	154
1	Dost Banana Milk	131
2	Dalin Shampoo	134
3	Doğadan Sage 20s	151
4	Flormar HC28 Urban Escape	135
5	Kinder Chocolate	158
6	Capri-Sun Safari Fruits	125
7	Rexona Roll On	130
8	Whiskas Aged Full-Fed Bovine	124
9	Ülker Salty Pretzels	156
10	İdeal Water	165
11	Ülker Chocolate Wafer	139
12	Çokonat	174
13	Ülker Hanımeller	155
14	Ülker Dankek Poti Cocoa	149
15	Eti Hoşbeş Chocolate	133
16	Eti Karam	163
17	Eti Sesame Pretzel	142
18	Eti Nero	156
19	Eti Popkek Lemon Single	128
	Total Data	**2902**

2.2. Model Training and Evaluation

YOLOv5 is used for real-time object detection. Google Colab environment is used for training.

2.2.1. YOLOv5

YOLOv5 is an object detection algorithm developed by Ultralytics. The YOLO approach to object detection is known for its real-time performance by combining object detection and classification into a single network. YOLOv5 builds upon the success of its predecessors, YOLO models, by incorporating several architectural improvements and training strategies. The proposed framework achieves state-of-the-art performance in terms of both accuracy and speed, making it suitable for various computer vision tasks. YOLOv5 consists of various convolutional layers of a network to solve an object detection problem. Basically, the input image first passes through a convolutional neural network (CNN) layer, and then passes through the output layers to predict the detected objects.

2.2.2. Training

Before the training process, the data set was divided into three parts for each class training, testing, and validation. Each class was first divided into 80% train, 20% test data, and then the train data was divided into 80% train and 20% validation.

Epoch is a term that refers to how long the entire data set takes to complete with a training algorithm. An epoch means that all training examples are passed by the model once during the training process.

During the training, the data set is processed on the model by dividing it into smaller parts (mini-batch). An epoch is completed at the point where all of these mini-batches are used. In a training process, more than one epoch is usually used. Each epoch performs more iterations by updating the weights of the model and improving its performance.

The epoch number can affect how long the training process will take and how well the model will perform. Generally, more epoch training tends to achieve better results. However, using too many epochs can increase the risk of overfitting. Therefore, the epoch number should be carefully selected depending on the complexity of the model, the size of the data set, and the goals of the training process.

During the training, the size of the batch was set to 16 and the number of epochs to 90.

One of the batches used during training is shown in the Figure 2.3.

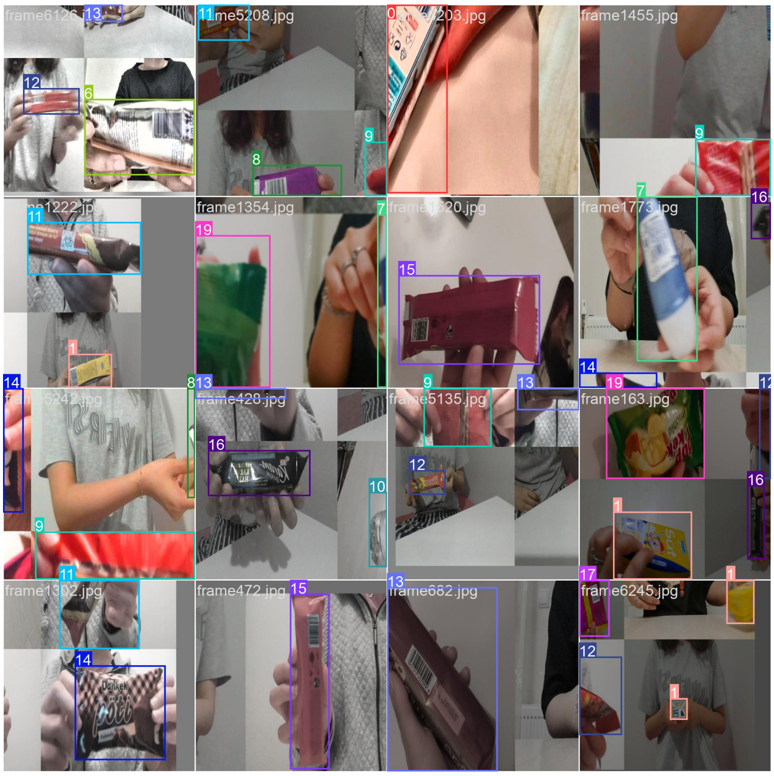

Figure 2.3: Train Batch.

The precision, recall, mAP graphs obtained after the training are shown in the Figure 2.4.

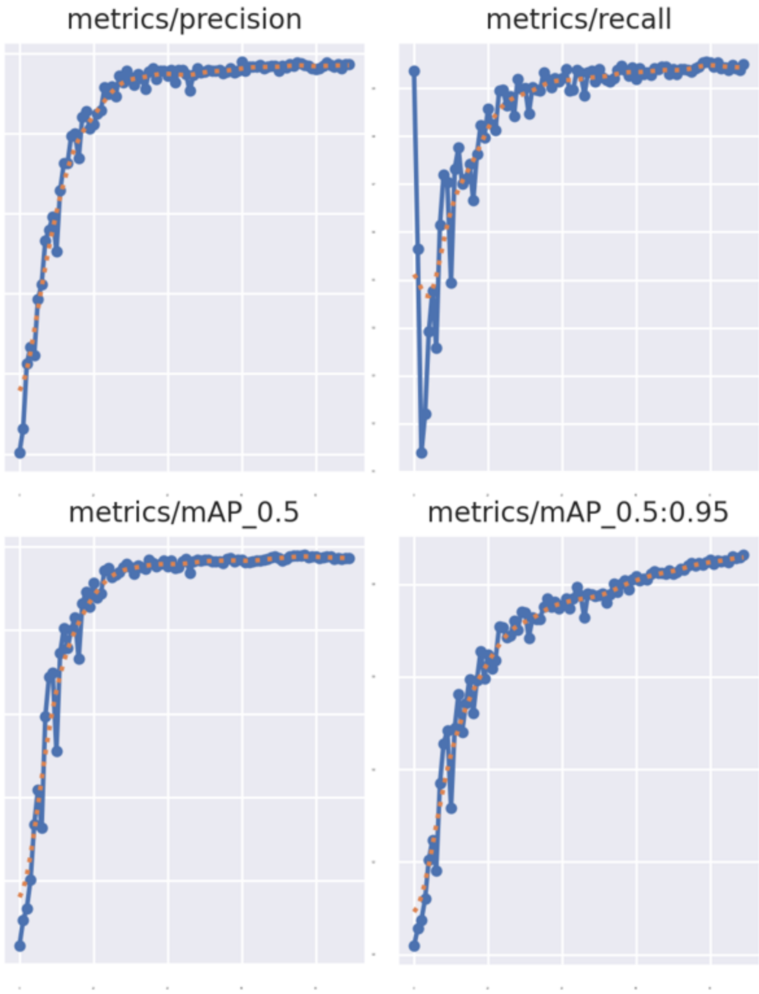

Figure 2.4: The Metrics Graph.

MAP represents the mean average precision, which is used to measure performance in a multi-class problem. A high MAP value indicates that the model is performing well overall.

Precision measures how accurate the predictions of the class are. As the precision value approaches 1, it means the model is making fewer false positive predictions.

Recall measures how well the model captures the true values of the class. As the recall value approaches 1, it means the model is making fewer false negative predictions.

The loss graph shows a measure of the model's performance during the training process. The loss of validation indicates the generalization ability of the model. The decrease in training loss over time indicates that the performance of the model on the data increases. In the ideal case, the training loss approaches zero or decreases steadily.

A bounding box loss graph is shown in Figure 2.5. Bounding box loss is used to accurately predict the position of the bounding boxes of detected objects. YOLOv5 estimates the coordinates and dimensions of the bounding box for each object. It measures the similarity of the actual bounding boxes to the predicted boxes and aims to increase this similarity.

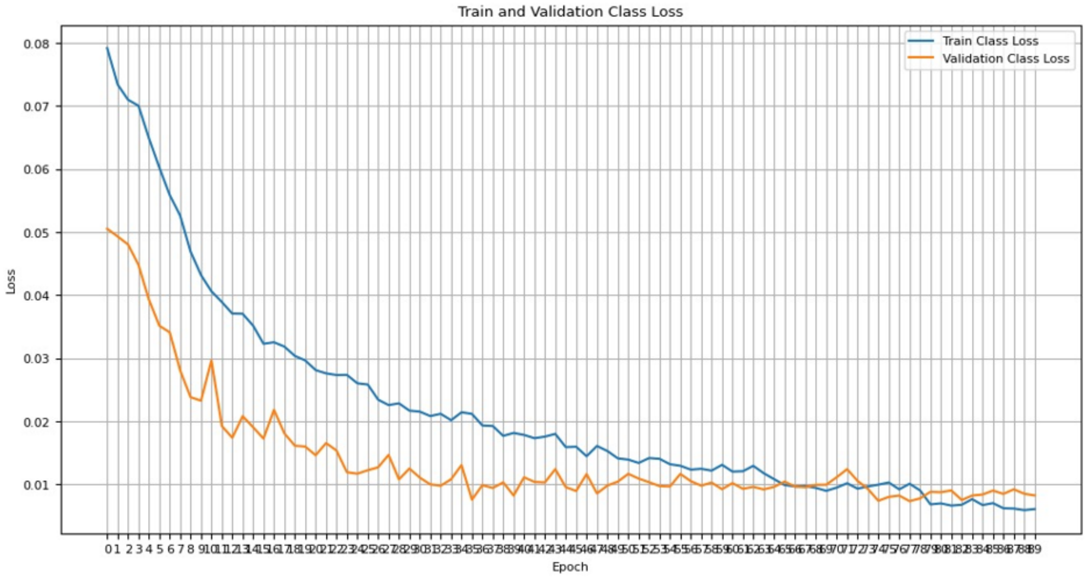

Figure 2.5: Train and Validation Box Loss Graph.

A class loss graph is shown in Figure 2.6. Class loss is used to accurately predict the classes of objects. YOLOv5 predicts the class label for each object. It measures the similarity between actual classes and predicted classes and aims to improve the accuracy of class predictions.

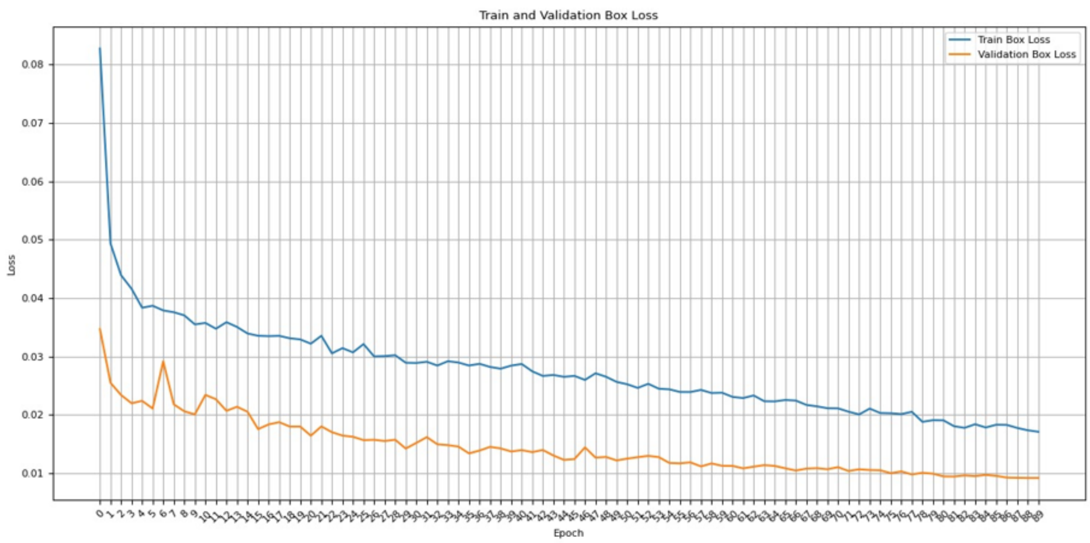

Figure 2.6: Train and Validation Class Loss Graph.

An objectness loss graph is shown in Figure 2.7. Objectness Loss is loss is used to determine the object's presence (objectness). YOLOv5 makes multiple predictions for each cell in each frame. It is related to the reliability of predictions in cells containing real objects. This loss encourages accurate prediction in cells for the detection of real objects.

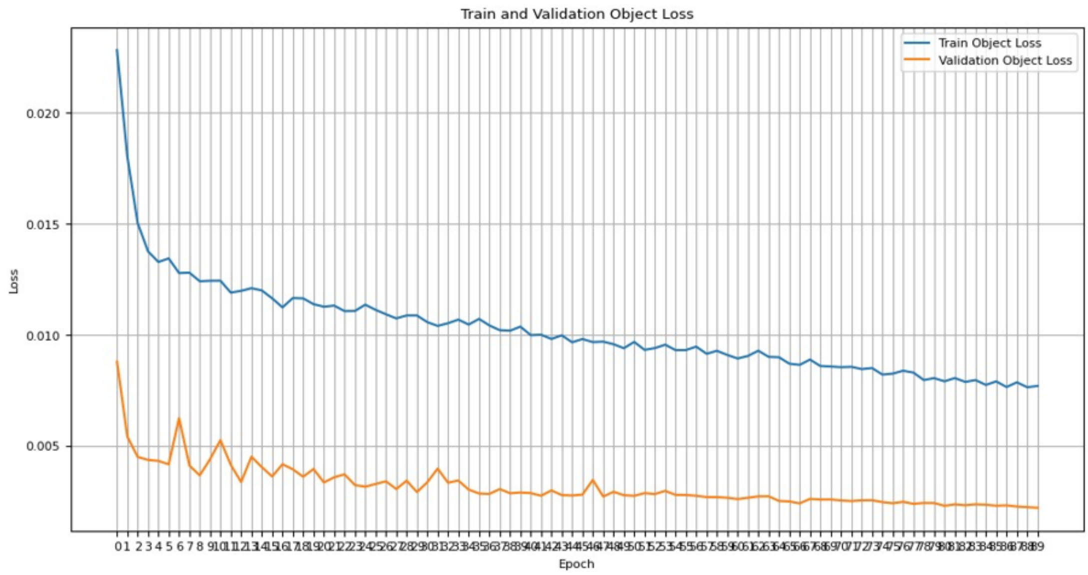

Figure 2.7: Train and Validation Object Loss Graph.

After the training model, the trained model was tested on the test data. Some of the test results are shown in Figure 2.8.

Figure 2.8: Test Results.

2.3. Recognition Application

Recognition application is a desktop application that performs real-time object detection with a camera using a trained model and sends a request to the API to reduce the stock of objects it detects. The user of this application can only be a cashier. This application should be built into every cashier checkout.

User login is requested when the application is launched. If the user is not a cashier, the error is given. Then, the camera turns on and the model waits to make object detection. The x-coordinate is followed from the first moment the model detects the product to the last moment. If there is a regular decrease or increase in the values and the end of the camera angle has been reached, a request is sent to the API to reduce the stock of the detected product. If no change is detected in the x-coordinate, that is, if the product is stationary, the stock is not deducted. The real-time product detection is as following three figures.

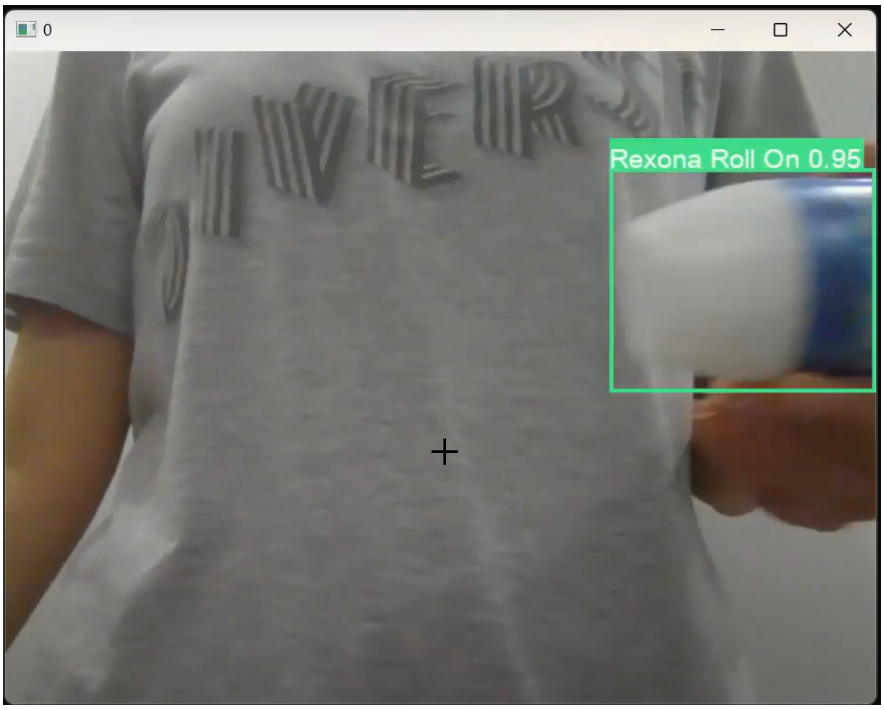

Figure 2.9: The application detects a product.

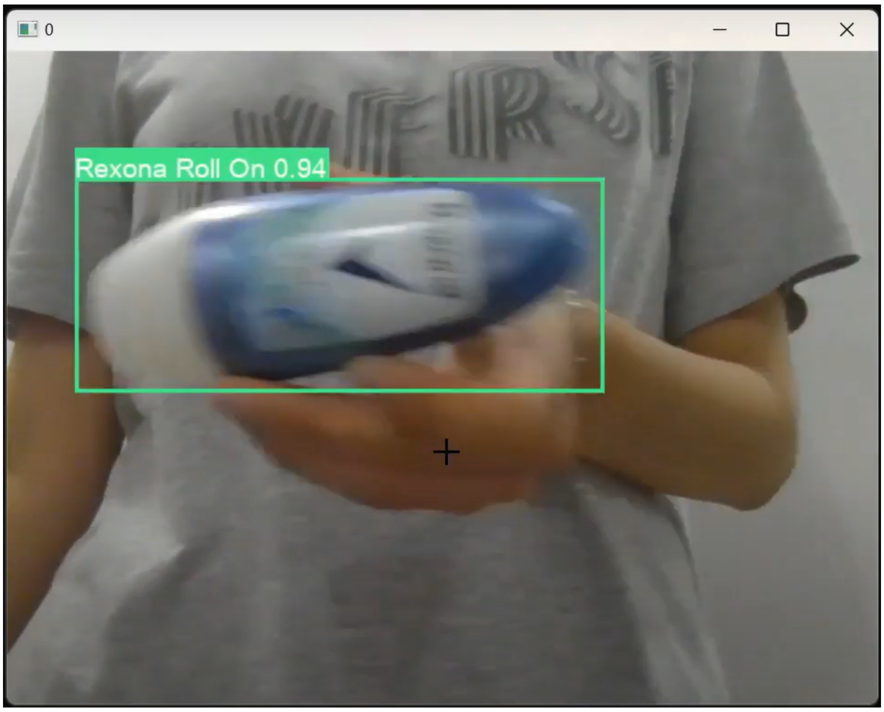

Figure 2.10: The product is still in the camera angle and the x-coordinate value is decreasing.

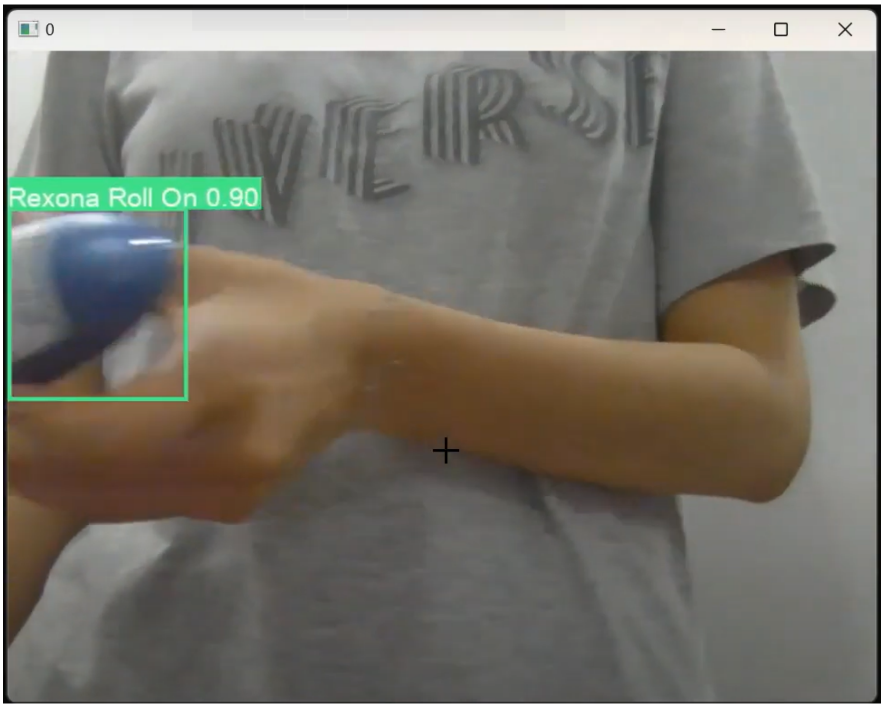

Figure 2.11: While the x-coordinate value continues to decrease, the product has started to come out from the camera point of view. The stock of the product is reduced.

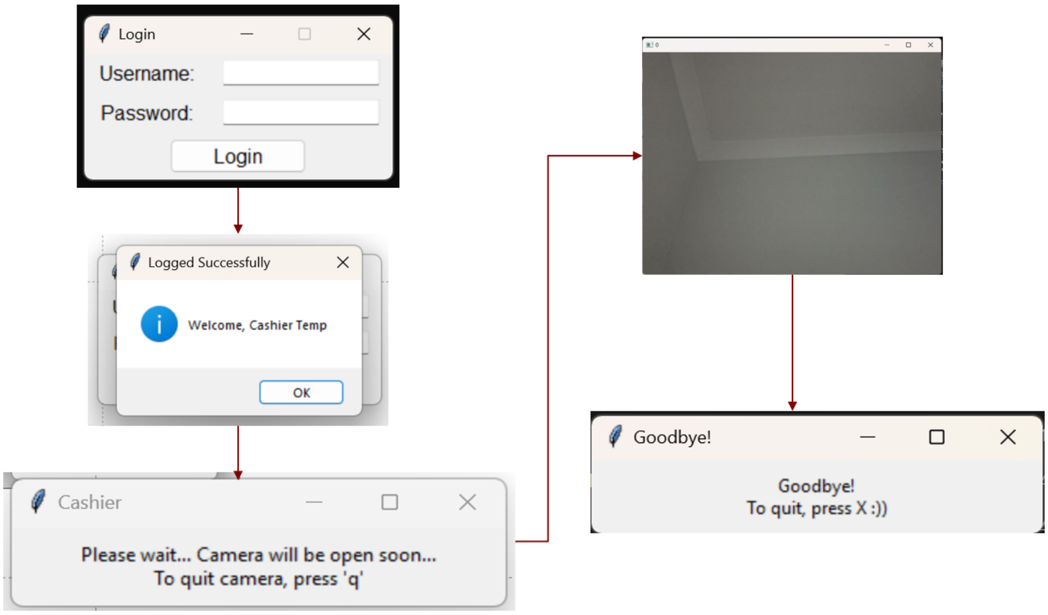

Figure 2.12: Recognition application user interfaces.

3. IMPLEMENTATION AND EVALUATION OF WEB APPLICATION

The web application will have 5 users. Users and what they can do in the system can be listed as follows:

- General Admin: The general admin can create a local admin and the product owner which means a user. Also, the general admin will create a product, such as Ülker, and assign a user to that product. The general admin will create a brand, such as Migros, and assign a local admin to this brand. General admin can list, edit and delete everything they created.

- Local Admin: The local admins can create a branch manager and a technical staff. In addition, they can create a branch, such as Gebze Migros, and assign a branch manager and technical staff to this branch. They can list, edit or delete everything they created.

- Branch Manager: The branch managers can create cashiers. Also, they can create a cashier checkout and assign a cashier to this cashier checkout. They can list, edit or delete everything they created.

- Product Owner (User): The users, that means product owners, can view the dashboard. They can see the list and detailed information about their own products. They can view which product has sold the most, and which branch has sold the most. They can see the stock status of the product, information such as how much has been sold in which branch, and how much has been left.

- Technical staff: The technical staff can train. They can add a zip file that contains images and label texts of the product to the system. With the file uploaded here, the model will be trained again and the recognition application provided to the user will be updated.

Frontend and backend have been developed according to user requirements. There are sample screenshots from the web application as follows:

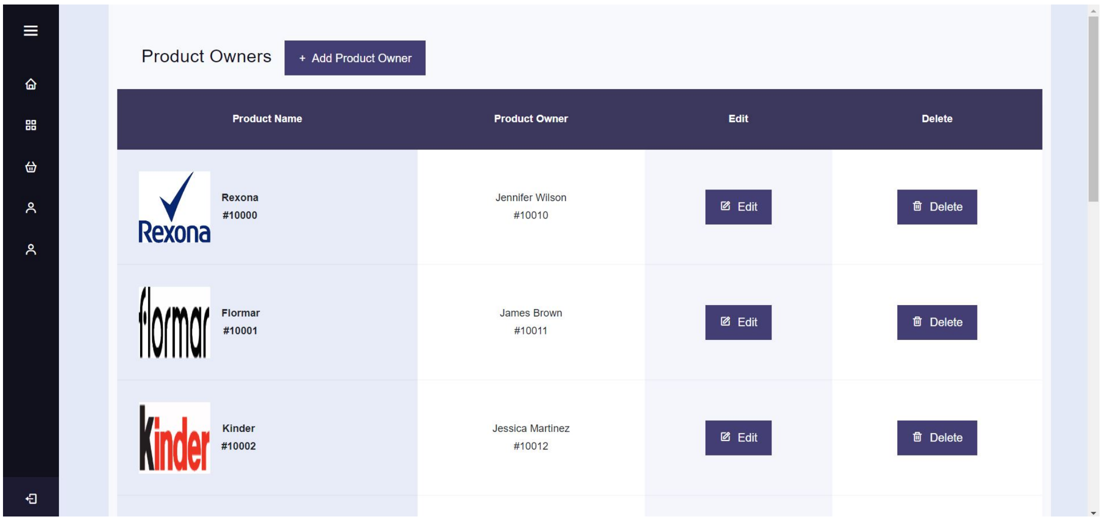

Figure 3.1: The product owner listing page of the general admin.

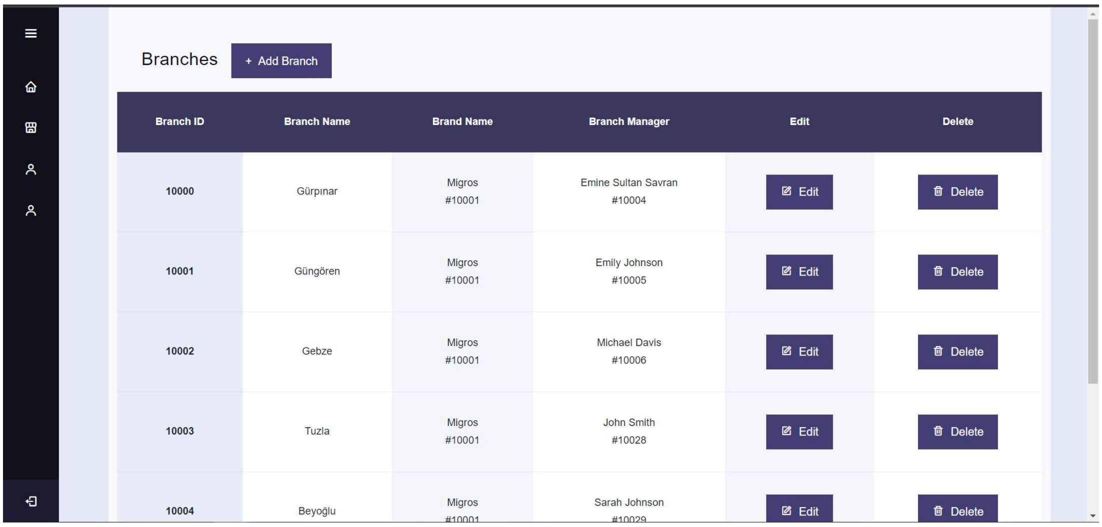

Figure 3.2: The branch listing page of the local admin.

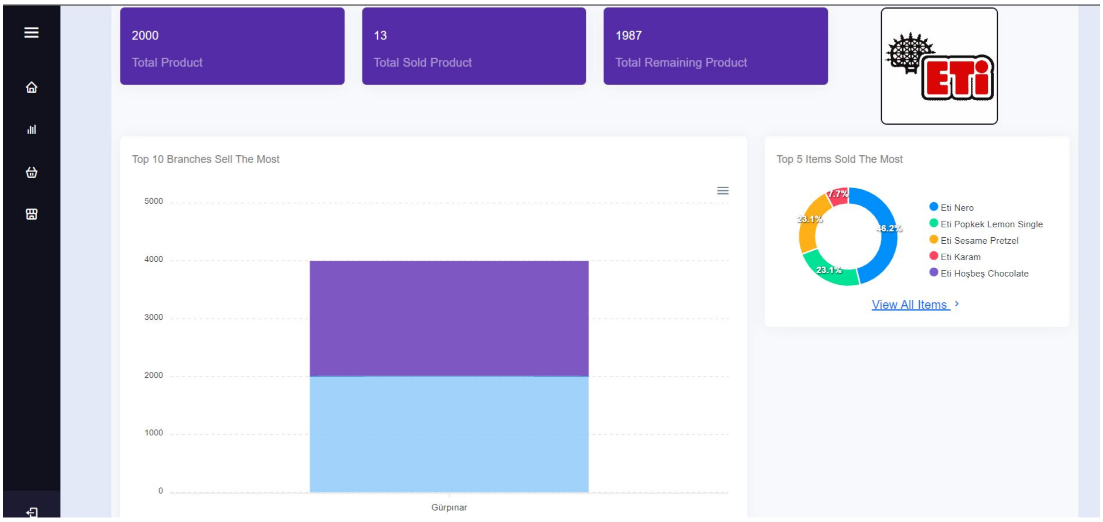

Figure 3.3: Dashboard page of the product owner.

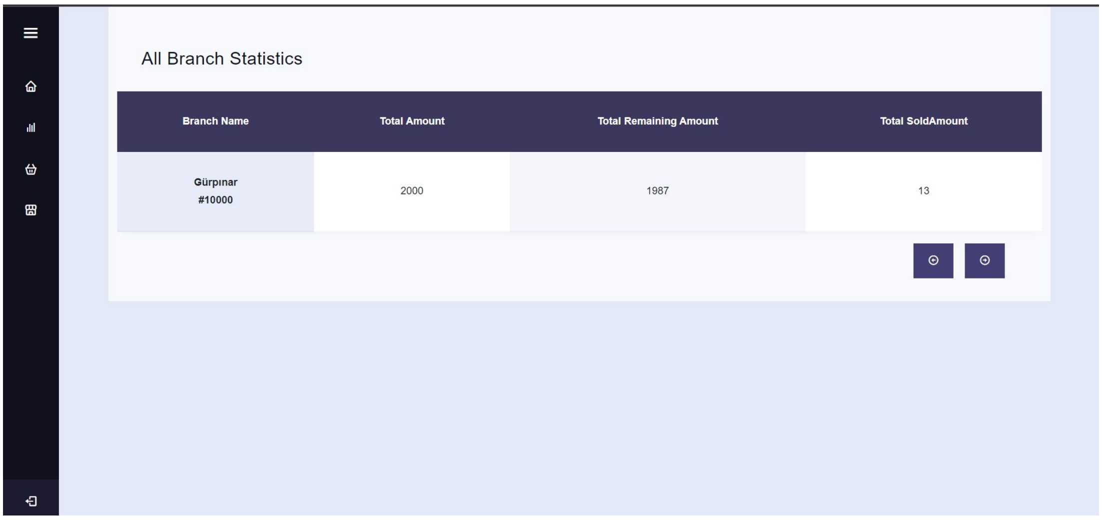

Figure 3.4: The page where the product owner sees the branch-based statistics of the products.

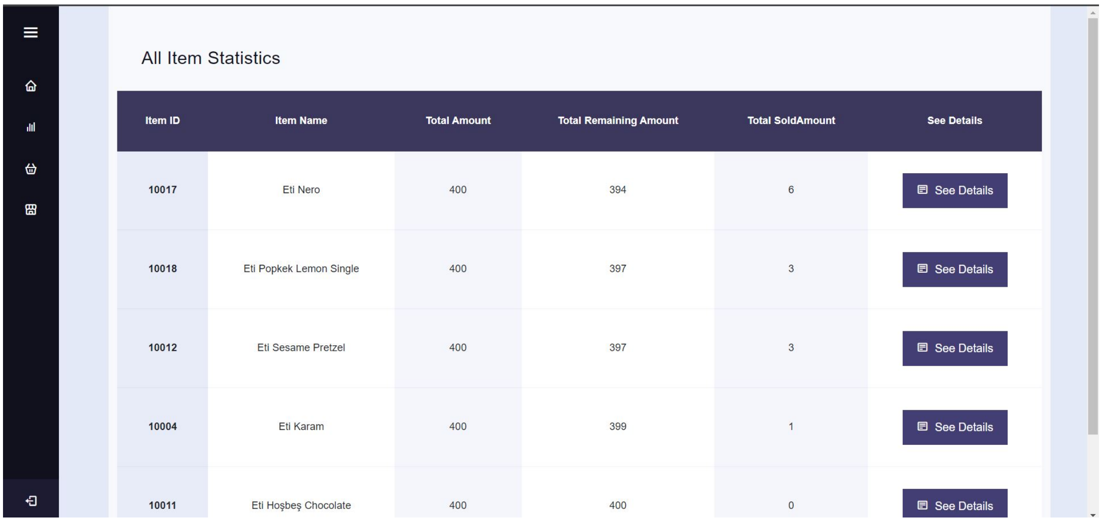

Figure 3.5: The page where the product owner sees the item-based statistics of the products.

4. CONCLUSIONS

This project presents a project that automates the process of item recognition at the cashier's checkout using the YOLO algorithm for real-time object detection. The project offers a real-time object detection solution by harnessing the power of YOLO, a high-performance and efficient computer vision technique.

The success of the project relies on the utilization of the YOLO algorithm, which can handle various challenges and accurately identify different objects with speed. The single-shot detection capability of YOLO enables faster processing compared to other algorithms, enhancing the speed of the checkout process and improving the overall customer experience. The real-time object detection implemented in the project enables swift scanning of items, minimizing errors and omissions in the scanning process.

In conclusion, this project successfully achieves the goal of automating item recognition at the cashier's checkout using the YOLO algorithm for real-time object detection. For the improvement of the project, the number of photos in the dataset can be increased and more different classes can be added.

BIBLIOGRAPHY

[1] Abto Software. "The Goldern Line: The Modern Solution for Railway Infrastructure." (2021), [Online]. Available: https://www.abtosoftware.com/blog/abto-golden-line.

[2] James, N., Antony, N. T., Shaji, S. P., Baby, S., Annakutty, "Automated checkout for stores: A computer vision approach," 2021.

[3] Yumang, Analyn N., Katipunan, John Anthony S., Galido, Abigail L., "Recognition and counting of materials used in fast food restaurants based on yolo object detection for inventory system," 2022.

CV

APPENDICES