



Bilkent University

Department of Computer Engineering

Senior Design Project

Group No: 2332

Project Name: Cyclops

Analysis and Requirement Report

21902348, Kaan Kurçer, kaan.kurcer@ug.bilkent.edu.tr

21902903, Osman Serhat Yılmaz, serhat.yilmaz@ug.bilkent.edu.tr

21903213, Ali Doğaç Urkaya, dogac.urbaya@ug.bilkent.edu.tr

21902358, Özgür Abi, ozgur.abi@ug.bilkent.edu.tr

21902035, Jankat Berslan Dinçer, berslan.dincer@ug.bilkent.edu.tr

Supervisor: Shervin Rahimzadeh Arashloo

Course Instructors: Erhan Dolak, Tağmaç Topal

09.11.2022

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Contents

1 Introduction	3
2 Proposed System	3
2.1 Overview	3
2.2 Functional Requirements	4
2.3 Non-functional Requirements	5
2.4 Pseudo Requirements	5
2.5 System Models	6
2.5.1 Scenarios	6
2.5.2 Use-Case Model	9
2.5.3 Object and Class Model	10
2.5.4 Dynamic Models	12
2.5.4.1 Activity Model	13
2.5.4.2 Sequence Diagrams	14
2.5.5 User Interface	15
2.5.5.1 Home Page	16
2.5.5.2 Login Page	17
2.5.5.3 Sign Up Page	18
2.5.5.4 Sign Up Page	19
2.5.5.5 Restaurant Information Page for Customers	20
2.5.5.6 Restaurant Information Page for Managers	21
2.5.5.7 Today's Customers Page for Customers	22
2.5.5.8 Customer Information Page for Managers	23
3 Other Analysis Elements	23
3.1 Consideration of Various Factors in Engineering Design	23
3.2 Risks and Alternatives	26
3.3 Project Plan	27
3.4 Ensuring Proper Teamwork	34
3.5 Ethics and Professional Responsibilities	34
3.6 Planning for New Knowledge and Learning Strategies	34
4 References	35

Analysis and Requirement Report

Cyclops: Restaurant Customer Tracker

1 Introduction

Restaurants are visited by hundreds of customers every day. Different types of customers have different spending habits, visiting times, and preferences. The traffic of the restaurant also varies greatly for different hours of the day. Although restaurant owners already record the inside of their restaurants for security reasons, they are unable to collect and utilize the data produced by their customers. The information gained from monitoring the customers' count and preferences could be used to enhance both the user experience of the visitors and the profits of the restaurant owners [1]. Cyclops aims to analyze the security feed of the restaurant to collect this information and provide it to the restaurant owner along with tips on how to improve their restaurant using said information.

2 Proposed System

Restaurants already collect a large amount of data via their security cameras, but this data is not utilized efficiently to improve dining experience. Our program will allow restaurant owners to optimize their restaurants in order to provide their customers with a personalized experience and a restaurant that fits their preferences. It would also let them get more customers during less busy hours through promotions or discounts. This way they can increase both the satisfaction and the number of their customers, which would lead to increased profits.

2.1 Overview

The first thing we want to do is to use a security camera facing towards the entrance to keep track of the customers while they are visiting the restaurant. The camera will use facial recognition technology to create an internal profile for each customer. Through this system the restaurant owner will be able to know exactly how many customers are in their restaurant at a given time. They will be able to look at a heatmap of how crowded their restaurant is at any given hour, and they will be able to use this information to promote customers to visit their restaurant more at less popular hours through discounts or special offers only available at those hours. We will also provide the information on how crowded a

restaurant is to the customers through a mobile application, so that they can plan their visits accordingly.

The customer profiles created by our facial recognition system will also be able to identify extra information about customers such as age and gender. This will allow restaurant owners to get a better idea of their user base. They can use this information to transform their restaurant to better fit the needs of their primary demographic, or to adhere to the preferences of customer types who don't usually visit the restaurant to make their restaurant more popular for different types of customers. The program will also count the number of people who sit together at a table and recognize them as a group. Similarly to the previous feature, the program will allow restaurant owners to know which types of groups of people visit their restaurants more regularly. These group types would include people with children, young friend groups, or people on dates, among others.

The customer profiles will not be deleted after the customers leave and will instead be kept in a database. This will allow restaurants to remember repeat customers. If the name of the customer was learned through some way by the waiters or payment systems, the restaurant staff will be able to input the name of the customer to the system and associate it with the profile. If the customer visits the restaurant again in the future, the waiters will be able to greet them by name or even give them recommendations on what to buy based on their previous activity in the restaurant. This will provide a personalized experience for the customers and make it more likely for them to have a positive, memorable experience in the restaurant, making it more likely that they visit again in the future.

2.2 Functional Requirements

- The customers and restaurant owner can view the number of people in the restaurant.
- The customers and restaurant owner will be able to view which tables are empty currently.
- The customers will be profiled by the security cameras (their age group and sex will be determined for now) and the resulting data will be presented to the restaurant owner.
- The customers will be recognized by security cameras by using facial recognition algorithms, and new customers will be saved to the database. Optionally, the name of the customer and information about the order can be obtained by the restaurant owner/personnel and saved to the database. Order information includes ordered food, size of the customer group and paid amount.

- Using the optional information mentioned above, the application will be able to extract customer specific information and the restaurant owner will be able to access this information and use it to enhance customer experience. The information includes the customer's preferred food and preferred table.
- The data collected by all above processes will be presented to the restaurant owner in charts. The data will be classified and several charts will be made for the restaurant owner to view (At which hours the restaurant is crowded, at what time male customers visit, which tables are usually preferred etc.).

2.3 Non-functional Requirements

User Friendliness

The app will be used by people who are not trained computer engineers therefore, it should be easy to understand and use for everyone.

Performance

The program should have a performance enough to work even during the busiest hours.

Dependability

Since the app will be used when the restaurants are actively working, the app needs to be dependable, it should not cause any problems for the users.

Maintainability

The app will have version control, and it will be designed in a way which allows it to be easily maintained.

Reusability

The app will be designed in a way that allows for it to be used in different settings and restaurants.

Security

The camera recording from the restaurant will be protected. Data belonging to users will also be stored safely.

2.4 Pseudo Requirements

- The project will have a web application, developed to work on modern browsers.
- We will have a client-server architecture.

- We will use Git for version controlling and tracking.
- We will build the web application using React which is based on JavaScript..
- Computer Vision part will be done using Python. However, we are also planning on using some other technologies for computer vision

2.5 System Models

2.5.1 Scenarios

1. Use Case: Login

Primary Actor: Manager

Stakeholders and Interests:

- The user wants to login to the system in order to use the application.

Pre-conditions:

- The user must open Cyclops from their computer.

Post-conditions:

- The user logs in and uses Cyclops.

Entry-conditions:

- The user must enter their username and password correctly.

Exit conditions:

- Login authorization is successful.
- The user terminates the application via the exit button.

Success Scenario Event Flow:

1. The user enters their username and password.
2. The user clicks on the login button.
3. Cyclops API approves the credentials and authorizes the user.
4. The user successfully logins.

Alternative Event Flows:

- The user provides incorrect login credentials.
- The user terminates the application.

2. Use Case: See Amount of Customers

Primary Actor: Manager and Customer

Stakeholders and Interests:

- The user wants to know the amount of customers currently in the restaurant.

Pre-conditions:

- The user must be on the “Restaurant Information” tab.
- Sufficient amount of cameras must be placed within points where they can observe the restaurant.
- The cameras must be linked with Cyclops in order to gather data for the application.

Post-conditions:

- The user sees the displayed amount of customers in the restaurant.

Entry-conditions:

- The user must switch to the “Restaurant Information” tab.

Exit conditions:

- The user switches to a different tab in the application.
- The user terminates the application.

Success Scenario Event Flow:

1. The user clicks on the “Restaurant Information” button on the navbar of the application.
2. The user sees the amount of current customers on their screen in the restaurant information tab.

Alternative Event Flows:

- The restaurant is empty.
- The cameras are malfunctioning, no data is received.

3. Use Case: See Restaurant Density on a Given Day

Primary Actor: Manager and Customer

Stakeholders and Interests:

- The user wants to see the density of their restaurant on a given day.

Pre-conditions:

- The user must be on the “Restaurant Information” tab.

Post-conditions:

- The user sees the restaurant density of a given day.

Entry-conditions:

- The user must click on the “Display Restaurant Density” button.

Exit conditions:

- The user switches to a different tab in the application.

- The user terminates the application.

Success Scenario Event Flow:

1. The user clicks on the “Display Restaurant Density” button.
2. The user chooses a specific date on the calendar that is displayed on the screen.
3. The user sees the density of customers in a given day as a chart separated by hours.

Alternative Event Flows:

- The chosen date has no data on customer density.

4. Use Case: See the Customers That Entered on a Day

Primary Actor: Manager

Stakeholders and Interests:

- The user wants to see the information related to a customer which entered the restaurant in a day.

Pre-conditions:

- The user must be on the “Today’s Customers” tab.

Post-conditions:

- The user is displayed the detailed information of a customer which entered the restaurant on that day.

Entry-conditions:

- The user must click on the customer’s number on the screen.

Exit conditions:

- The user switches to a different tab in the application.
- The user terminates the application.

Success Scenario Event Flow:

1. The user clicks on a specific customer’s number on the screen.
2. The detailed information of a customer is displayed on the screen.

Alternative Event Flows:

- No customers enter the restaurant.

5. Use Case: See a Specific Customer’s Information

Primary Actor: Manager

Stakeholders and Interests:

- The user wants to see information related to a specific customer.

Pre-conditions:

- The user must be on the “Detailed Customer Information” tab.

Post-conditions:

- The user is displayed detailed information about a customer.

Entry-conditions:

- The user must click on the search button.

Exit conditions:

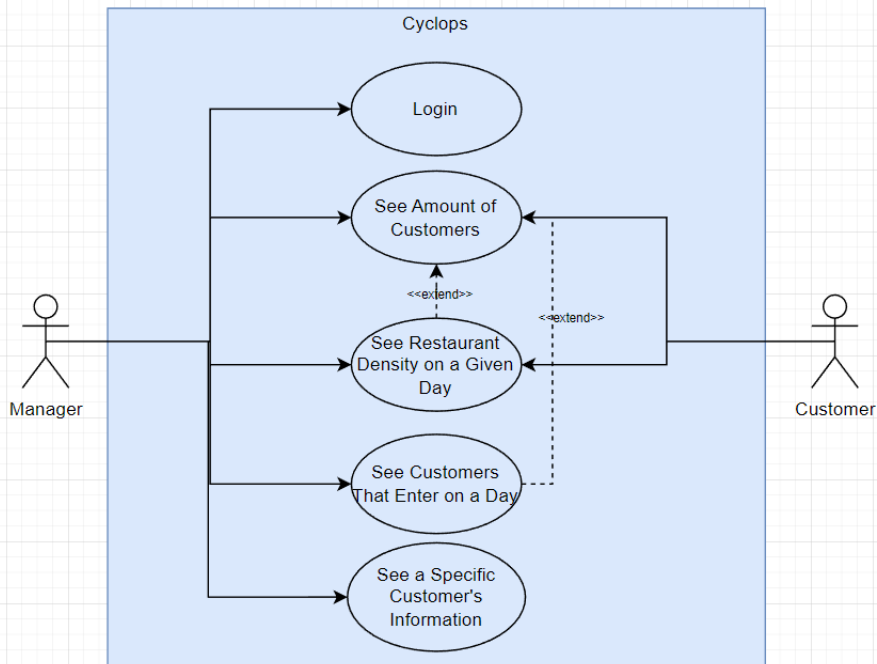
- The user switches to a different tab in the application.
- The user terminates the application.

Success Scenario Event Flow:

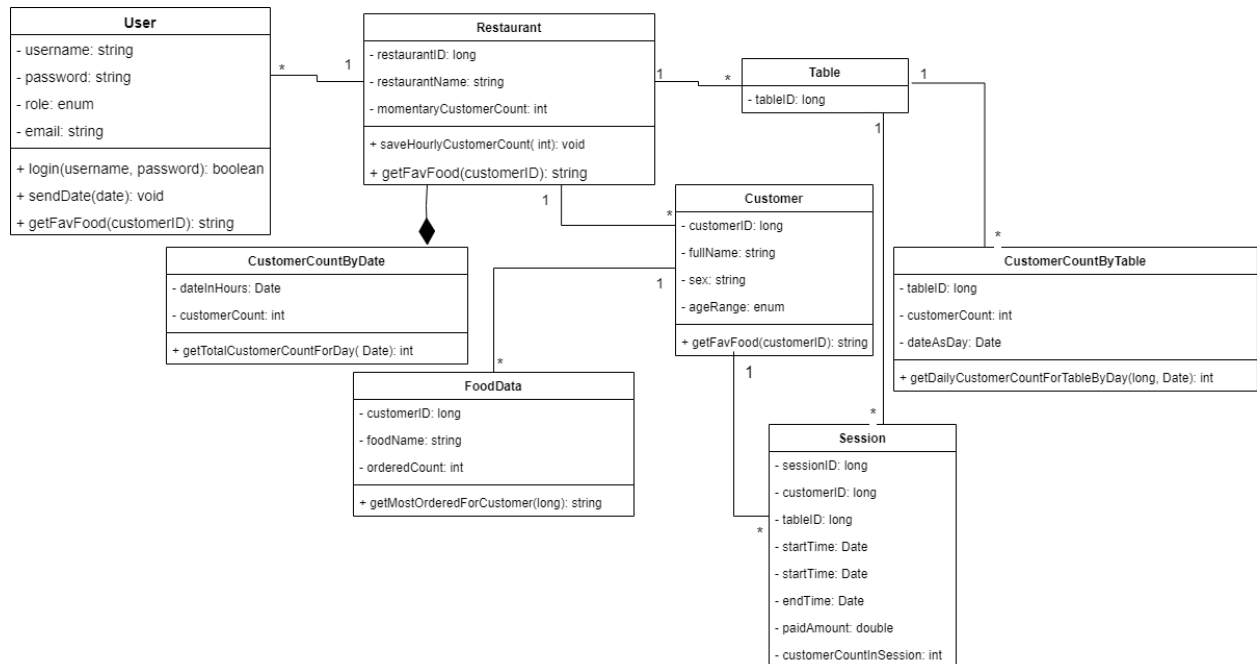
The user types in the name of the customer.

The user is shown the detailed information of the customer.

2.5.2 Use-Case Model



2.5.3 Object and Class Model



All the classes will have get-set methods for all of their fields, it is not written in this diagram for simplicity.

Explanation Of Classes

Restaurant

Restaurant class has information about the restaurant such as restaurant id, restaurant name. It also includes the momentary customer count which is going to be displayed in our system for customers to see. Also it saves the hourly customer count entry to the CustomerCountByDate table.

User

User class has information of all users such as username and password, and also role of the user to authorize the users inside the application.

Customer

Customer class has information about the customers such as customer id, name of the customer, sex and age range of customer.

Table

Table class represents each individual table in a restaurant, this class has a respective id for every table and it is used in different tables such as Session and CustomerCountByTable.

Session

Session class has information about a session of a customer, it has its own id, customer id, table id in it. It also keeps track of start time and end time as a date in minutes, also the number of customers in one table for a session.

FoodData

FoodData keeps track of a customer's past orders, it is a table consisting of food name and customer id, it keeps the count of ordered times for a meal by a customer.

CustomerCountByTable

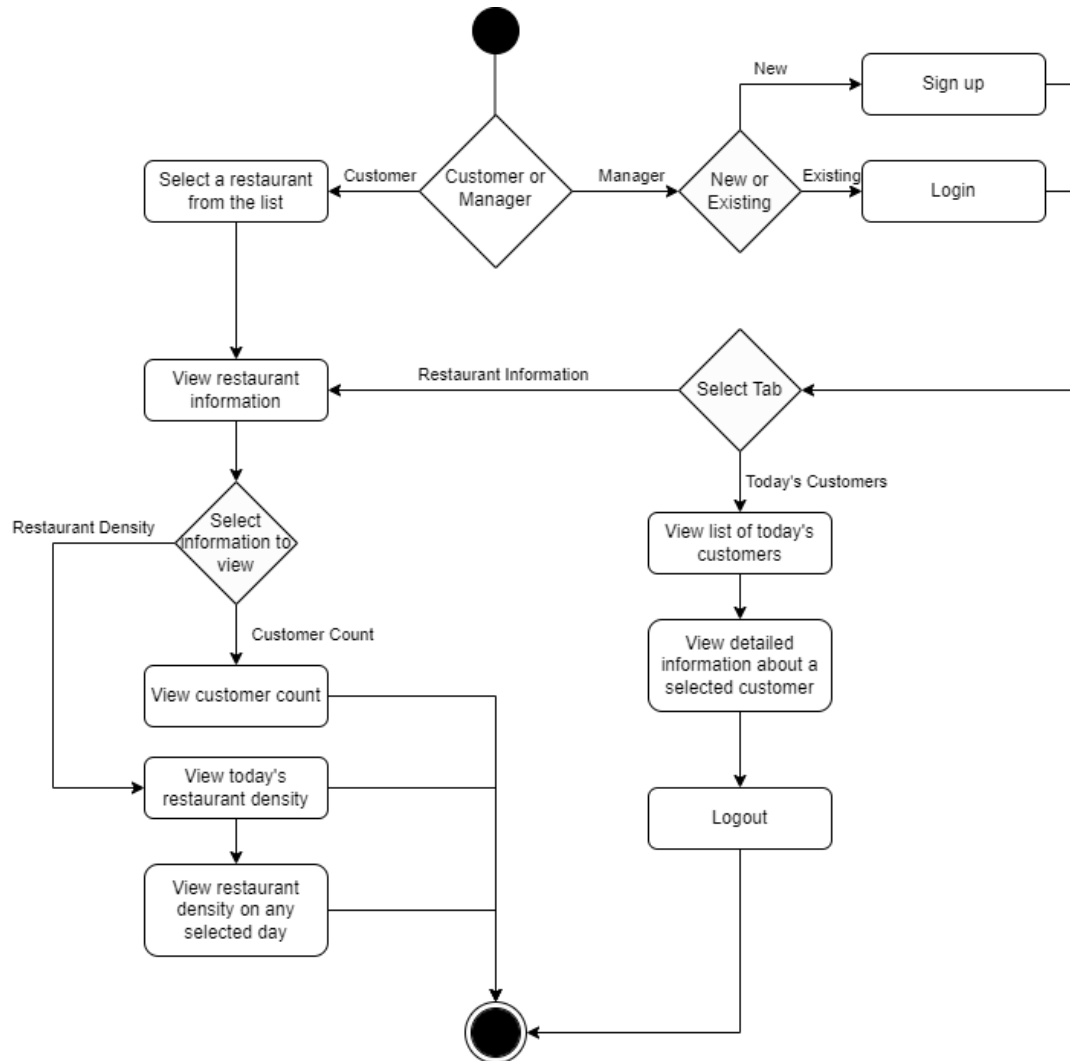
This class has the information of customer count for a given table id and a given day. This data will be used to display restaurant performance analysis to the restaurant owner.

CustomerCountByDate

This class keeps track of the customer count in every hour, it takes the information from the Restaurant class. This data will be used to display restaurant performance analysis to the restaurant owner.

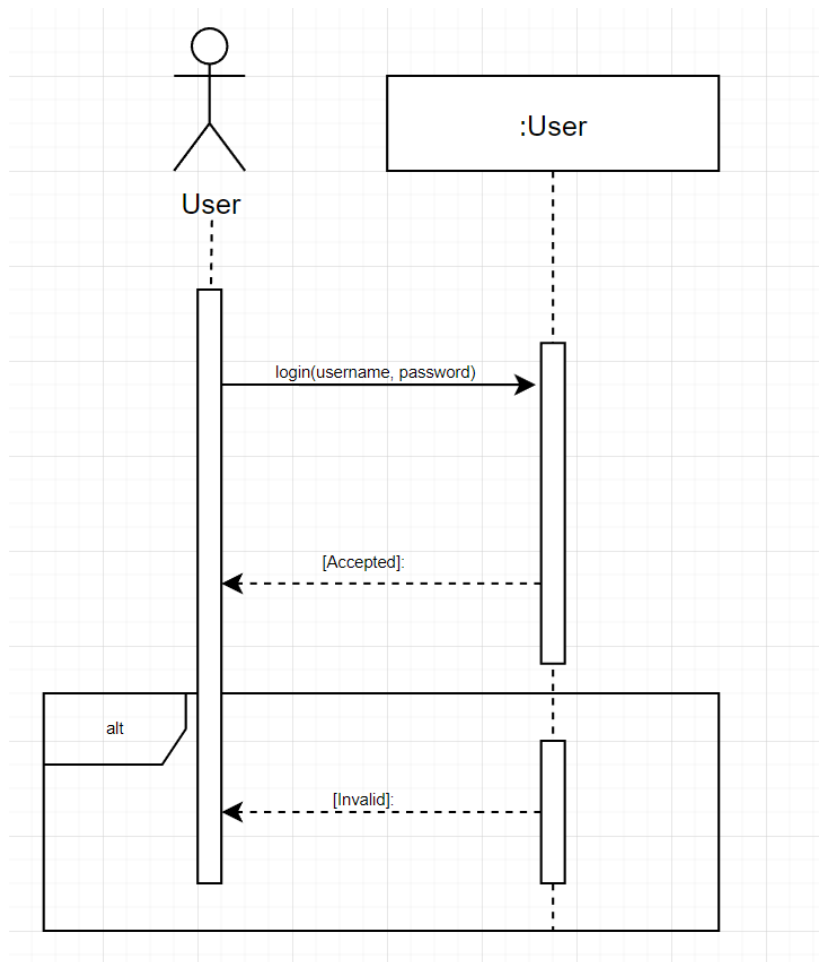
2.5.4 Dynamic Models

2.5.4.1 Activity Model

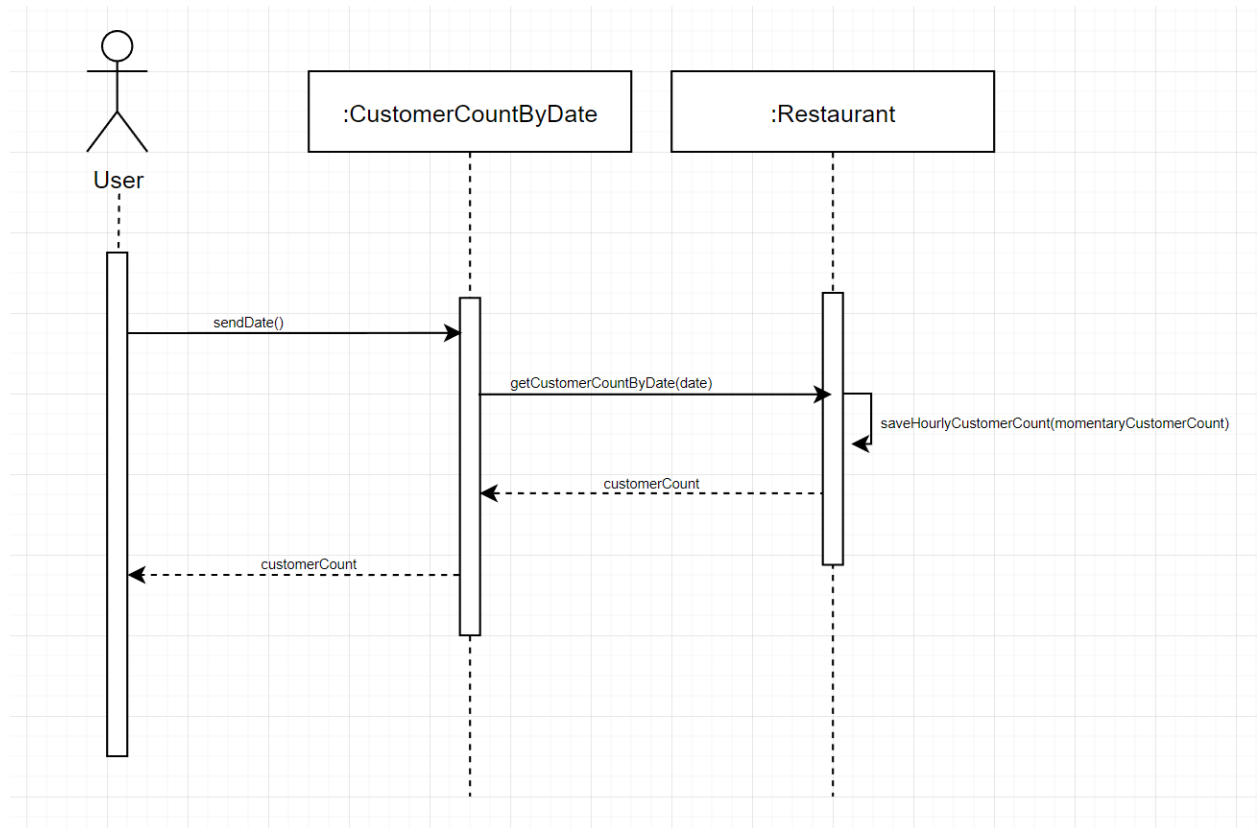


This diagram show how user can interact with the program and the operations that user can do by using our application.

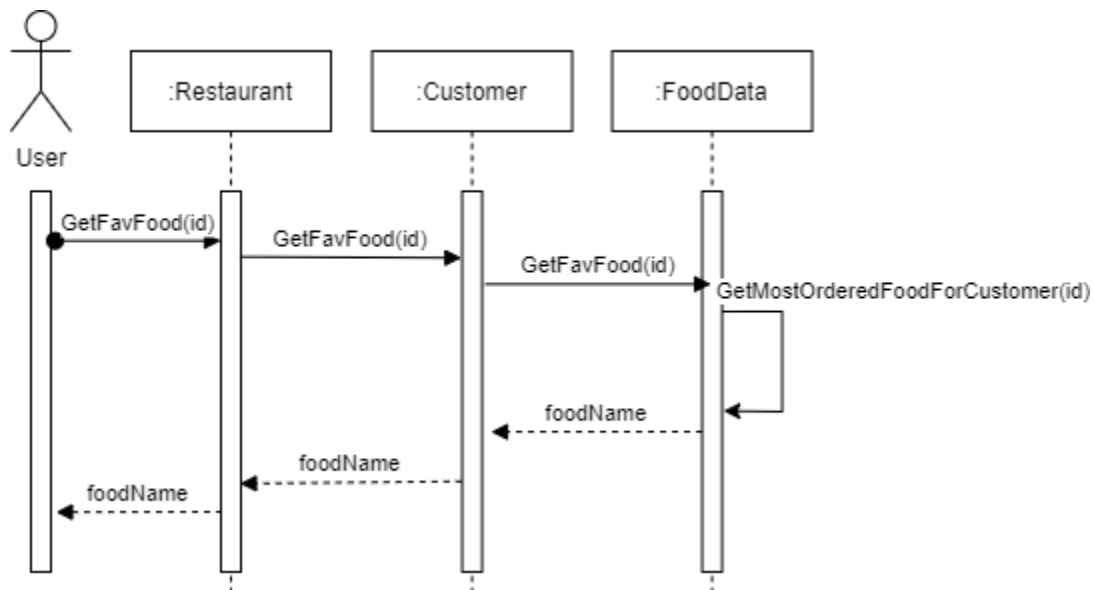
2.5.4.2 Sequence Diagrams



The user sees the login screen. The user logs in by providing their username and password. Then Cyclops authorizes the login and proceeds to the home page.



The user provides the date at which they want to get the customer data. The Cyclops API returns the customer count data by querying the restaurant object.

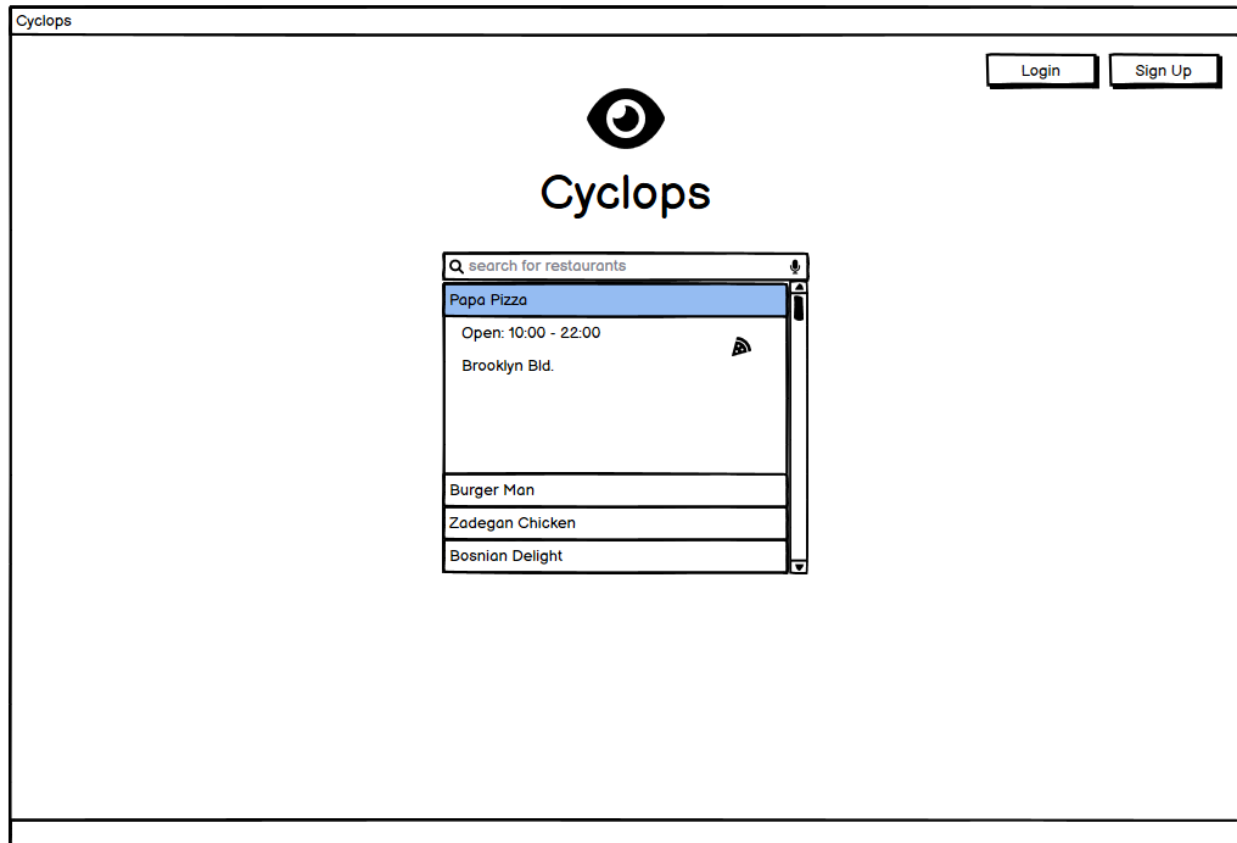


The user asks for the favorite food of a specific customer. The Cyclops API returns the favorite food of a given customer by querying the customer and food data objects.

2.5.5 User Interface

UI Mockups

2.5.5.1 Home Page



This is the first page users encounter when accessing the application. Customers can use the search function and restaurant list to select a restaurant. They will be directed to the restaurant information page. Managers can use this page to register or login to the application.

2.5.5.2 Login Page

Cyclops


Cyclops

Login

Email:

Password:

[Forgot password?](#)

Managers will use this page to login. They can also reset their password.

2.5.5.3 Sign Up Page

Cyclops


Cyclops

Sign Up

* Name:

* Email:

* Password: ?

* Re-type password:

☐ I agree to the [Terms of Use](#) and [Privacy Policy](#).

[Learn more](#)

Managers will use this page to register for the application.

2.5.5.4 Sign Up Page

Cyclops



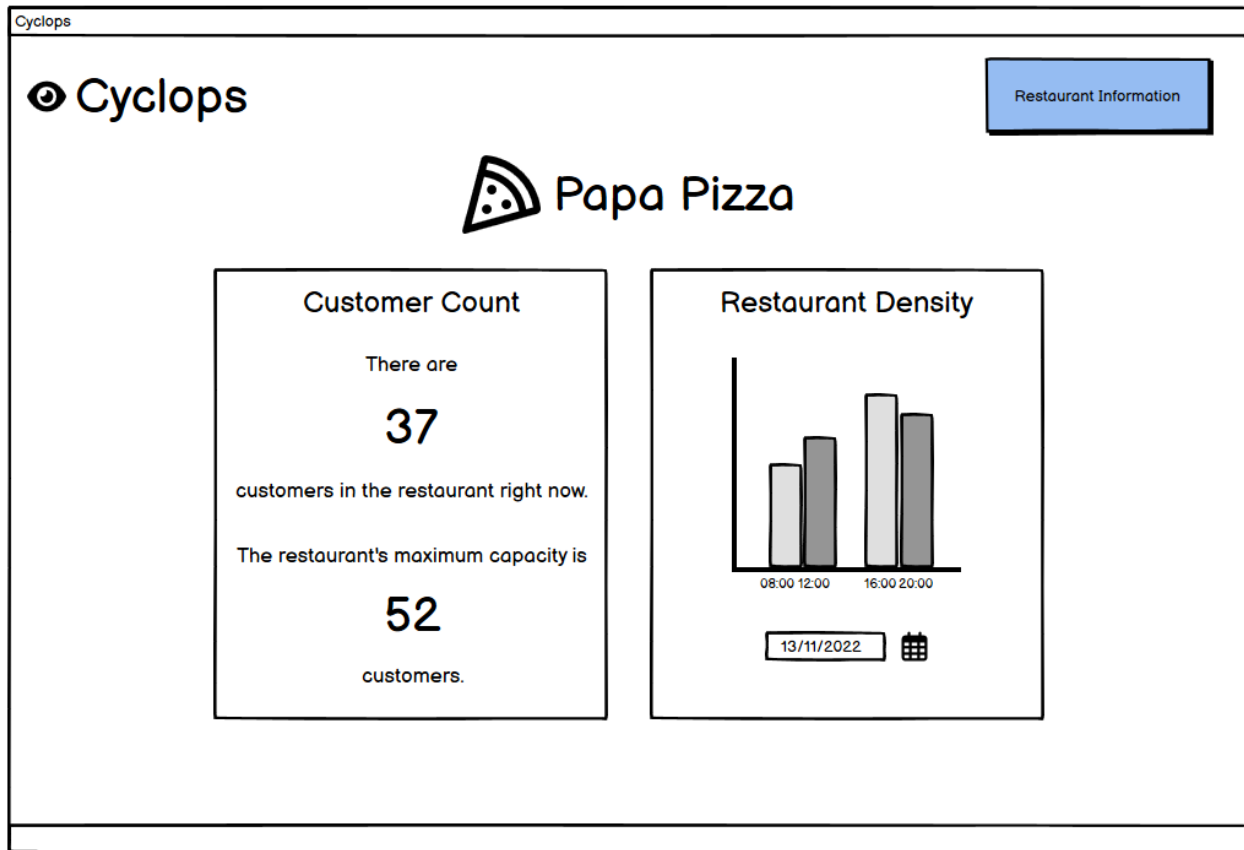
Cyclops

Forgot Password

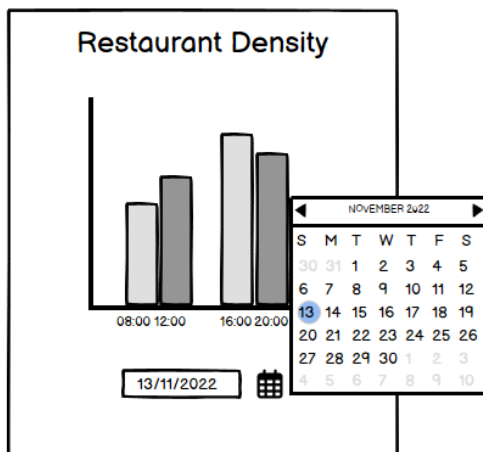
Email:

Managers will be able to reset their passwords if they forget their credentials.

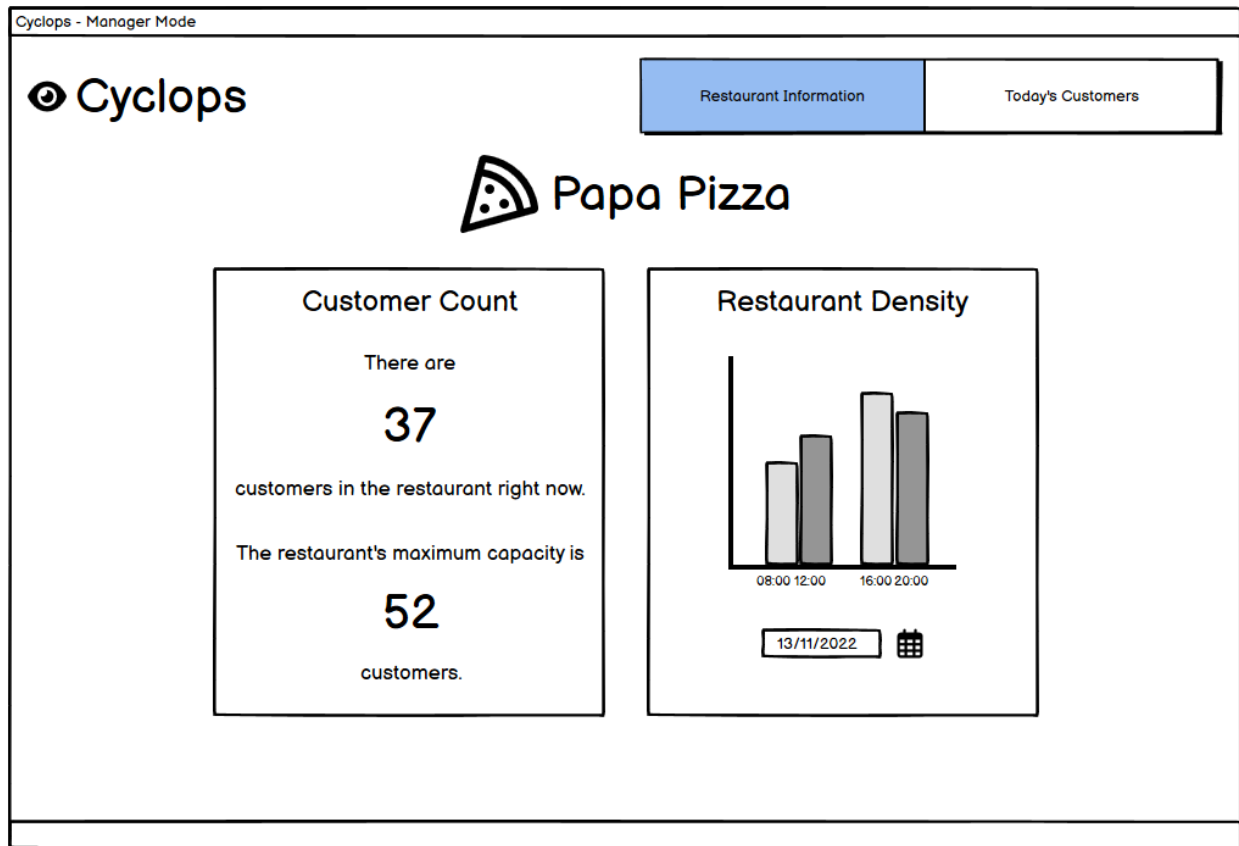
2.5.5.5 Restaurant Information Page for Customers



Customers will be able to view how many customers are currently in the restaurant and the capacity of the restaurant. They will also be able to view restaurant density to have information about when the restaurant is more full and when to make reservations. The page shows the restaurant density of the current day by default, but customers can click on the calendar to view restaurant density of other days.



2.5.5.6 Restaurant Information Page for Managers



This page is identical to the restaurant information page for customers, but it also has a navbar for managers to switch between tabs at will.

2.5.5.7 Today's Customers Page for Customers

Cyclops - Manager Mode



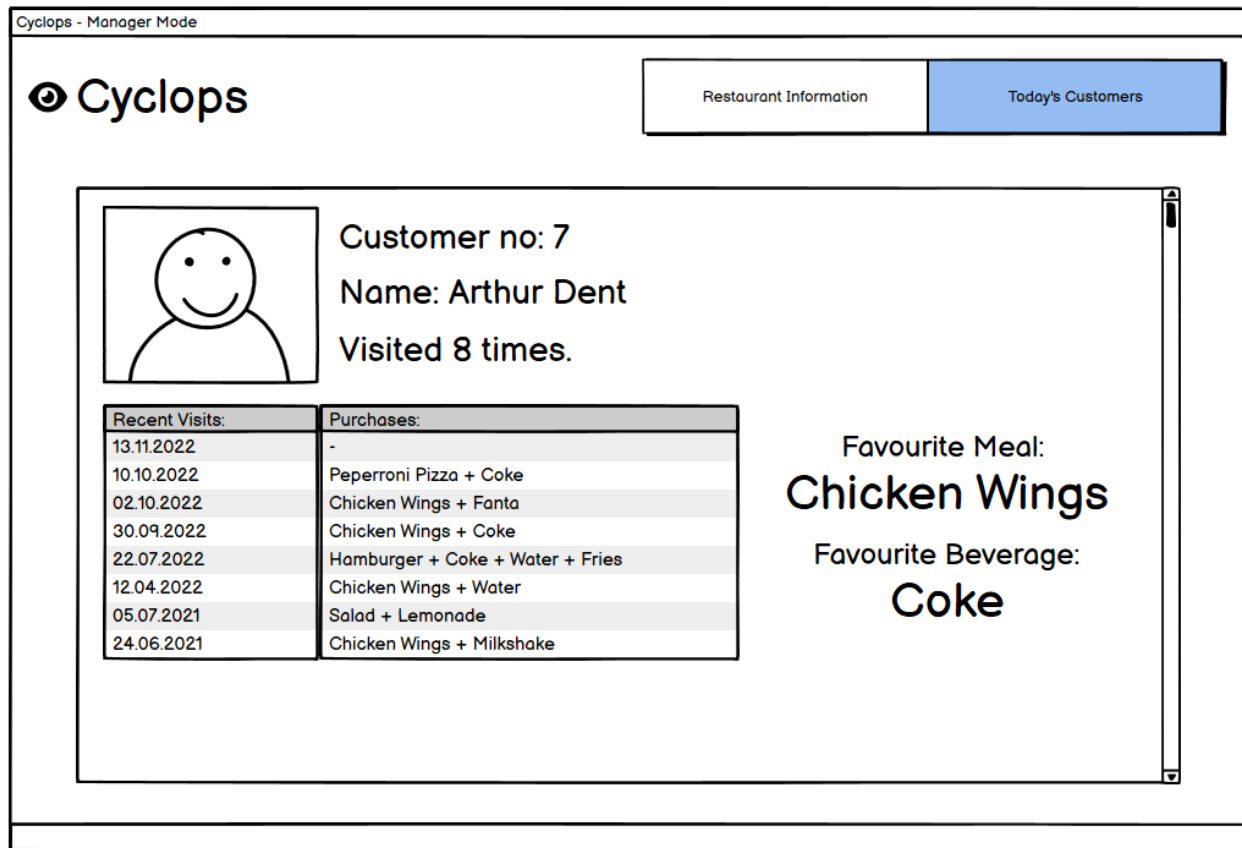
Restaurant Information

Today's Customers

 Name: Theodore Ted Entered at: 10:32 Left at: - Table no: 19 Customer no: 9	 Name: Ford Prefect Entered at: 10:11 Left at: - Table no: 42 Customer no: 8	 Name: Arthur Dent Entered at: 10:11 Left at: - Table no: 42 Customer no: 7	 Name: Unknown Entered at: 09:23 Left at: 09:28 Table no: 23 Customer no: 6	 Name: Recep Kılıçoğlu Entered at: 09:05 Left at: - Table no: 19 Customer no: 5
 Name: Unknown Entered at: 08:24 Left at: 09:05 Table no: 7 Customer no: 4	 Name: Unknown Entered at: 08:24 Left at: 09:06 Table no: 7 Customer no: 3	 Name: Unknown Entered at: 08:06 Left at: 08:42 Table no: 3 Customer no: 2	 Name: Unknown Entered at: 08:02 Left at: 08:24 Table no: 2 Customer no: 1	

Managers and employees will be able to view a list of profiles for the customers currently in the restaurant. They will be able to click on profiles to get detailed information.

2.5.5.8 Customer Information Page for Managers



Managers and employees will be able to view individual profiles for each of the customers currently in the restaurant. They will be able to click on profiles to get detailed information about previous visits and preferred foods.

3 Other Analysis Elements

3.1 Consideration of Various Factors in Engineering Design

Public Health

This application will be used by customers to see the momentary number of customers in a restaurant, therefore customers will be able to select the restaurants that are not dense, which will prevent diseases from spreading.

Public Safety

The application does not intend to have any effects on public safety directly. However, it can still be used for safety because of its ability to keep track of customers.

Customers who harass others or create disturbances can be automatically detected by the software and banned from restaurants. This feature is not among our primary goals, but it can be implemented if the project is built upon in the future.

Public Welfare

This application allows customers to pick the most suitable restaurant for them by displaying the customer density at the moment, therefore customers will be able to choose a restaurant that they won't wait in a queue.

This application increases restaurant owners capabilities to enhance the restaurant's customer experience and management efficiency, depending on the customer preferences.

Global Factors

This application provides personalized experiences to customers in places where it is not a cultural aspect of the service industry. Through the usage of our project such experiences may become more popular in such places.

Cultural Factors

This application will allow restaurant personnel to recognize incoming customers and greet them with their name, also suggesting their favorite food to them while taking the meal order. These actions can be done in small restaurants by restaurant owners since their customers are not as much as big restaurants and their customers are usually the same. But this is not applicable for big restaurants since the customer number is bigger and personnel can change.

Social Factors

This application will allow restaurant personnel to provide personalized experiences by greeting customers by name and remembering their previous visits. This will increase customer satisfaction and socialization between customers and restaurants.

Table 1: Factors that can affect analysis and design.

	Effect level	Effect
Public health	4	Restaurant density displays can be used by customers if they wish to avoid crowded restaurants during pandemics.
Public safety	3	Although the project does not have built in safety features, its features can be used to enhance the safety of restaurants by keeping track of troubling customers.
Public welfare	6	The project allows restaurant customers to pick the best restaurant for them at a time, without waiting in a queue at the restaurants. The project increases restaurant owners capabilities to enhance the restaurant's customer experience and management efficiency, depending on the customer preferences.
Global factors	1	The culture of greeting customers and providing personalized experiences does not exist in some countries, which might now

		start to exist because of our project.
Cultural factors	7	Similarly to the previous section, this culture of greeting customers might start to spread among different restaurants. It will also be moved from primarily existing in smaller restaurants to being present in larger restaurants and franchises where employees would normally have no chance of remembering information about their many customers.
Social factors	7	Restaurants will be able to provide personalized experiences by greeting customers by name and remembering their previous visits. This will increase customer satisfaction and socialization between customers and restaurants.

3.2 Risks and Alternatives

Table 2: Possible Risks

	Likelihood	Effect on the project	B Plan Summary
Risk of leaking customer data	Very Low: With sufficient safety features the likelihood of this risk can be lowered.	Customers' visitation data being leaked would cause problems for the application and might reduce the chance of customers visiting restaurants using our program.	In order to prevent this risk, our program must be built with proper cybersecurity measures to prevent hacks and similar attacks. The risk must be prevented.
Risk of restaurant owners trying to sell customer data	Medium: Restaurant owners might try to use the data they are collecting maliciously.	Our program does not collect data illegally, or produce malicious data. If the data we provide is used maliciously, this reflects on the users and would legally not affect our project. It could, however, produce bad publicity for the project.	Restaurant owners would be informed of the legal problems they would face upon using our data maliciously, and would have to sign legal documents accepting the responsibilities. Restaurants that are detected to be using our data maliciously will be marked by the system and the authorities will be informed immediately.

3.3 Project Plan

Although we have divided the project into the given work packages below, we plan on working flexibly and switching work packages when needed. This will also be important for integrating the systems into the project.

Table 3: List of work packages

WP#	Work package title	Leader	Members involved
WP1	Human Counting System	Osman Serhat Yılmaz	Kaan Kurçer Ali Doğaç Urkaya Özgür Abi
WP2	Web Application	Ali Doğaç Urkaya	Jankat Berslan Dinçer Özgür Abi
WP3	Facial Recognition System	Kaan Kurçer	Jankat Berslan Dinçer Ali Doğaç Urkaya Osman Serhat Yılmaz
WP4	Server Application	Özgür Abi	Ali Doğaç Urkaya
WP5	Database Design	Jankat Berslan Dinçer	Ali Doğaç Urkaya Kaan Kurçer Özgür Abi

WP6	Integration of All Systems	Osman Serhat Yılmaz	Jankat Berslan Dinçer Özgür Abi Ali Doğaç Urkaya Kaan Kurçer
-----	----------------------------	---------------------	---

WP 1: Human Counting System			
Start date: 25.11.2022 End date: 28.03.2023			
Leader:	Osman Serhat Yılmaz	Members involved:	Kaan Kurçer Ali Doğaç Urkaya
Objectives: <i>In this work package, we aim to count the customer number in a restaurant by using computer vision. We will analyze the restaurant tables and reach the customer number to display it for the user and also to save the restaurant database on an hourly basis.</i>			
Tasks: Task 1.1 Research on optimal camera placement: <i>In this task we will focus on how to place cameras to capture the accurate human count in the restaurant.</i> Task 1.2 Count all humans in the restaurant: <i>In this task we aim to count the exact number of humans in the restaurant including the personnel.</i>			

Task 1.3 Customer and personnel differentiation : *In this task we aim to differentiate personnels and customers and then provide more accurate data for restaurant owners.*

Deliverables

D1.1: *Report on implementation and design choices of this system*

WP 2: *Web Application*

Start date: 20.11.2022 **End date:** 29.04.2023

Leader:	<i>Ali Doğaç Urkaya</i>	Members involved:	Jankat Berslan Dinçer Özgür Abi
----------------	-------------------------	--------------------------	---------------------------------------

Objectives: This web app will allow the users of the app, like restaurant owners, to see the data gathered by the app. Users will be able to review restaurant statistics such as customer count.

Tasks:

Task 2.1 Create a UI : *Design a webpage that is usable by restaurant owners.*

Task 2.2 Connect UI and backend : *Connect the website with the backend of the application*

Deliverables

D2.1: Report on implementation and design choices of the web app

D2.2: Web app will be displayed on web

WP 3: Facial Recognition System

Start date: 10.03.2023 **End date:** 10.04.2023

Leader: Kaan Kurçer

**Members
involved:**

Jankat Berslan
Dinçer

Ali Doğaç Urkaya

Osman Serhat
Yılmaz

Objectives: *In this part of the project, we aim to recognize the customer by using computer vision and face recognition systems if it is already saved before in the system. If the customer is not recognized by the system, save it to the database.*

Tasks:

Task 3.1 Research on optimal camera placement: *In this task we will focus on how to place cameras to reach optima performance in a facial recognition system.*

Task 3.2 Tests on how to implement: *In this task we will make experiments after we implemented our facial recognition system and report its accuracy.*

Task 3.3 Integration with database: In this task we will connect our facial recognition system with the database of a restaurant.

Deliverables

D3.1: Report on performance of facial recognition system and implementation of it.

WP 4: Server Application

Start date: 20.11.2022 **End date:** 21.04.2023

Leader: Özgür Abi

Members involved:

Ali Doğaç Urkaya

Objectives: In this package, we will create the server application which restaurant owners will install to their own system. It will gather the information from security cameras and put them to the database. It will also manage the information to create usable data, such as average customer count during day hours.

Tasks:

Task 4.1 Application skeleton : In this task we will create the basic features of the application such as running constantly on the server and connecting with the database.

Task 4.2 Camera connection : We will establish the connection between the restaurant's security cameras and the application.

Task 4.3 Processing data : Using our previous work, we will get the data from cameras and put them in the database with a certain format. Then we will use the retrieved data to create useful statistics.

...

Deliverables

D4.1: *Report on implementation and design choices of this system*

WP 5: Database Design

Start date: 20.11.2022 **End date:** 20.04.2023

Leader:	Jankat Berslan Dinçer	Members involved:	Ali Doğaç Urkaya Kaan Kurçer Özgür Abi
----------------	-----------------------	--------------------------	--

Objectives: *We will be designing a database to hold information about the restaurants we are serving. The database will also keep all restaurant customers and their information.*

Tasks:

Task 5.1 Restaurant Database : *Our program will serve many different restaurants, and we will need an internal database to keep information about the restaurants. We will list the restaurants and their information on our homepage, which will utilize this database.*

Task 5.2 Customer Databases : *Each restaurant will want to keep information about their customers. This information includes visual data, visitation dates, and preferred foods.*

Deliverables

D5.1: ER Diagram for restaurant database

D5.2: ER Diagram for customer databases

D5.3: Report on the databases we have designed and our design choices

WP 6: Integration of All Systems

Start date: 20.03.2023 **End date:** 10.05.2023

Leader:	Osman Serhat Yılmaz	Members involved:	Jankat Berslan Dinçer Özgür Abi Ali Doğaç Urkaya Kaan Kurçer
----------------	---------------------	--------------------------	--

Objectives: *The goal of this work package is to bring everything we have done together and complete the project, integrating all the systems we have developed into a working program.*

Tasks:

Task 6.1 Integration of systems : The developed systems will be brought together and integrated into a final working program.

Task 6.2 Additional polish : *The project will be given a final look and necessary improvements will be made as required to improve project quality.*

Deliverables

D6.1: Final project demo

D6.2: Project presentation

3.4 Ensuring Proper Teamwork

We will have regular meetings and we will constantly be in communication to ensure that everyone is able to meet their goals and complete their responsibilities. We don't plan on having everyone work fully individually, we will assist each other with points we are stuck on and problems we are facing.

3.5 Ethics and Professional Responsibilities

We are using facial recognition, so the data we collect might be sensitive for some users. Customers will be informed that they are being recorded in restaurants, which already currently happens in Turkey. We will inform restaurants about what they are legally allowed to do with the data they collect, and legal authorities will be informed by us in cases of misuse of our program.

3.6 Planning for New Knowledge and Learning Strategies

4 of our members are currently taking machine learning courses, and we plan on taking computer vision next semester. We will also be doing a lot of individual research on topics such as AI and facial recognition on our own.

4 References

[1] R. Carriere, "Is it time for facial recognition in restaurants?," QSR magazine, 21-Jan-2022. [Online]. Available: <https://www.qsrmagazine.com/outside-insights/it-time-facial-recognition-restaurants>. [Accessed: 16-Oct-2022].

[2] Object-Oriented Software Engineering, Using UML, Patterns, and Java, 2nd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2004, ISBN: 0-13-047110-0.