

PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

Ministry of Higher Education and Scientific Research



UNIVERSITY OF ECHAHID
HAMMA LAKHDAR - EL OUED
FACULTY OF EXACT SCIENCES
Computer Science Department



Licence Thesis Title:

**Development and Implementation of a
Queue Management System for University
Social Services**

Specialty : Information Systems

Presented by:

KIRAM Mohammed Ali
NACEUR Safouane

**Proposed and
Supervised by:**

Dr. BOUCHERIT Ammar

Discussants:

Dr. KHEBBACHE Mohibeddine
Dr. SETTOU Tarablesse

-Academic year: 2023/2024-

Acknowledgements

We extend our deepest gratitude to all those who have contributed to the success of our startup. Special thanks to our dedicated team members whose hard work and commitment have been instrumental in bringing our vision to life.

We are immensely grateful to our advisors and mentors for their invaluable guidance, wisdom, and support throughout our journey. Their expertise has been indispensable in shaping our strategy and navigating challenges.

We would like to express our appreciation to our investors for their confidence in our vision and their unwavering support. Their belief in our mission has enabled us to realize our goals and aspirations.

We also want to thank our early adopters and customers for their trust and feedback. Your insights have been invaluable in refining our products and services to better meet the needs of our users.

Last but not least, we are thankful to our friends, families, and loved ones for their understanding, encouragement, and unwavering belief in us. Your endless support has been the foundation upon which we have built our startup.

Abstract

In today's world, service quality is no longer measured only by meeting requirements but also by completing tasks with high efficiency and in the shortest possible time. Therefore, organizing services for workers has become a necessary and important matter, especially in the case of large numbers, and from here the process of managing queues becomes at the forefront of these needs. From this standpoint, this project appeared to organize queues before the Social Services Committee at the University of El-Oued to relieve pressure on the Social Services Office and ensure a reduction in waiting times for service beneficiaries. This project allows organizing waiting queues in an easy and simple way for both beneficiaries and employees, ensuring that beneficiaries do not lose their place in line until they receive the required service. In addition, this project ensures that paper waste is avoided, reflecting ethical, economic and environmental considerations.

Key words : Queue Management System , DevelopmentImplementation of Queue Management SystemUniversity Social ServicesQueue Management in UniversitiesSocial Services ,ImprovementQueue System for University Services

Résumé

Dans le monde d'aujourd'hui, la qualité du service ne se mesure plus uniquement par le respect des exigences, mais également par l'exécution des tâches avec une grande efficacité et dans les plus brefs délais. Par conséquent, l'organisation des services pour les travailleurs est devenue une question nécessaire et importante, surtout dans le cas d'un grand nombre de travailleurs, et à partir de là, le processus de gestion des files d'attente devient au premier plan de ces besoins.

De ce point de vue, ce projet semble organiser des files d'attente devant le Comité des Services Sociaux de l'Université d'El-Oued pour soulager la pression sur le Bureau des Services Sociaux et assurer une réduction des délais d'attente pour les bénéficiaires des services. Ce projet permet d'organiser les files d'attente de manière simple et facile, tant pour les bénéficiaires que pour les employés, en garantissant que les bénéficiaires ne perdent pas leur place dans la file d'attente jusqu'à ce qu'ils reçoivent le service requis. De plus, ce projet garantit que le gaspillage de papier est évité, reflétant des considérations éthiques, économiques et environnementales.

Les mots clés : Mise en œuvre d'un système de gestion des files d'attente. Développement d'un système de gestion des files d'attente. Système de gestion des files d'attente. Services sociaux universitaires. Gestion des files d'attente à l'université. Amélioration des services sociaux

ملخص

الآن ونحن في عصر السرعة في كل جوانب الحياة وأهمها جانب الخدمات ، لم يعد تقديم الخدمة الجيدة يقاس بمعيار انجاز المطلوب وبكفاءة عالية فقط وانما انجاز المطلوب بكفاءة عالية و في أقل وقت ممكن. لذا فان تنظيم الخدمات أصبح أمر مطلوب ومهم و تنظيم صفوف الانتظار يأتي في مقدمة ذلك، من هذا المنطلق جاء هذا المشروع لتنظيم صفوف الانتظار للهيئة الاجتماعية الجامعية لتخفيف الضغط على الموظفين خلف الشبائيك، و ضمان تقليل وقت الانتظار للمستفيدين من الخدمات الاجتماعية حيث يتيح هذا المشروع تنظيم صفوف الانتظار، و بطريقة سهلة وبسيطة لكل من المستفيدين والموظفين، و ضمان عدم ضياع دور المستفيد حتى يتلقى الخدمة التي يريدها ، كذلك يضمن هذا المشروع عدم التبذير في الورق وهي لفئة أخلاقية دينية وبيئية.

الكلمات المفتاحية : تطوير نظام إدارة الطوابير، تنفيذ نظام إدارة الطوابير، الخدمات الاجتماعية الجامعية، نظام إدارة الطوابير، إدارة الطوابير في الجامعات، تحسين الخدمات الاجتماعية.

Table of Contents

Acknowledgements	I
Abstract in English	II
Abstract in French	III
Abstract in Arabic	V
Table of Contents	VII
List of Figures	IX
List of Tables	X
General Introduction	1
Chapter 1: Basic Concepts and Study of the Existing	2
1.1 Introduction	2
1.2 Queue management definition:	2
1.3 Types of queue management	4
1.4 Examples queue management system	8
1.5 The problem statement	18
1.5.1 The university Social Services Committee	18
1.5.2 Fields of social services	18
1.5.3 Description of the problem and some previous solutions	18
1.6 Project overview	19
1.7 Conclusion	20
Chapter 2: System Modelling and Design	21
2.1 Introduction	21
2.1.1 The importance of the Unified Modeling Language and its significance	21

2.1.2	The benefits of Unified Modeling Language	21
2.1.3	Types of Unified Modeling Language Diagrams	22
2.1.4	The Unified Modeling Language diagrams used in our project study	22
2.2	Analysis	23
2.2.1	Actors	23
2.2.2	Task Specification for Each Actor in the Queue Manager System .	23
2.3	Unified Modeling Language diagrams	26
2.3.1	Use Case Diagram	26
2.3.2	Sequence Diagrams	27
2.3.3	Activity Diagrams	31
2.3.4	Class Diagrams	32
2.3.5	The data dictionary tables	33
2.4	Conclusion	35
Chapter 3: System Development		36
3.1	Introduction	36
3.2	Deployment environment	36
3.3	Development environment	37
3.3.1	Visual Studio Code:	37
3.3.2	MySQL:	37
3.4	The programming languages	38
3.4.1	PHP Programming Language:	38
3.4.2	CSS Programming Language and HTML Technology:	39
3.4.3	java script Programming Language:	40
3.5	the frameworks:	42
3.5.1	Laravel:	42
3.6	Some of the interfaces:	43
3.6.1	Login page:	43
3.6.2	Committee worker page:	44
3.6.3	Service Session page:	46

Table of Contents

3.6.4	Display queue information page:	47
3.6.5	Display next ticket:	48
3.6.6	Get your ticket page:	49
3.6.7	Administrator page:	51
3.6.8	Add employee page:	52
3.6.9	Delete employee page:	53
3.7	Conclusion	53
	General Conclusion	54

List of Figures

I.1	a single channel, single-phase model	4
I.2	a single channel, the multi-phase model	5
I.3	multi-channel, single phase	6
I.4	multi-channel, multi-phase	7
I.5	ATLAS Wavetec Queue management system	9
I.6	Qmatic Queue management system	11
I.7	Qtrac Queue management system	12
I.8	Qminder Queue management system	14
I.9	Qwaiting Queue management system	15
I.10	Verint Queue management system	16
I.11	Qless Queue management system	17
II.1	Use Case Diagram for all system	26
II.2	Log in sequence diagram	27
II.3	Administrative sequence diagram	28
II.4	calling customer sequence diagram	29
II.5	Get ticket sequence diagram	30
II.6	Activity diagram of our project	31
II.7	Class diagram of our project	32
III.1	Visual Studio Code logo	37
III.2	MySQL logo	37
III.3	PHP logo	38
III.4	HTML logo	39

III.5 CSS logo	39
III.6 JavaScript logo	40
III.7 Laravel logo	42
III.8 Login page	43
III.9 Committee worker page(No customer available.)	44
III.10Committee worker page(Committee Worker 1)	45
III.11Service Session page	46
III.12Display queue information page	47
III.13Display next ticket in queue information page as model	48
III.14Get your ticket page	49
III.15Ticket to print (ticket 18)	50
III.16Administrator page	51
III.17Add employee page	52
III.18Delete employee page	53

List of Tables

II.1	Schema of the employees table	33
II.2	Schema of the login table	33
II.3	Schema of the tickets table	34
II.4	Schema of the service_session table	34

General Introduction

In the modern era, public and service organizations face great challenges in organizing customers efficiently due to the increasing diversity and abundance of services. Among these institutions, social service committees stand out for their efforts to provide social support through the distribution of aid and resources and the provision of solidarity services to employees at the level of each institution. As the number of patrons increases, the need for an automated queue management system becomes essential to improve efficiency and reduce chaos. Automated systems help manage customer flow, reduce wait times, and improve the overall user experience.

Problem Statement:

The Social Services Committee at the University of El-Oued faces organizational challenges during two major events: the distribution of “Ramadan food baskets” and “school bags.” During these two events, a large number of beneficiaries go to the social services office to obtain receipts, resulting in long lines and uncomfortable waiting times. Accordingly, this overcrowding leads to beneficiary dissatisfaction and difficulties in managing operations effectively. Therefore, there is an urgent need to develop an automated queue management system to improve efficiency, reduce congestion, and ensure fair and rapid distribution of aid.

Study Objectives:

This study aims to develop an automated queue management system for the Social Services Committee at the University of El-Oued to improve management of the flow of beneficiaries and reduce overcrowding during major events such as the distribution of Ramadan food baskets and school bags. The system that we are seeking to develop aims to reduce waiting times, enhance the user experience, and ensure that aid is distributed fairly and quickly. In addition, the project aims to enhance the committee’s operational efficiency and improve beneficiary satisfaction by adopting modern queue management techniques.

Methodology:

The project will be accomplished through several stages (chapters), including:

- **Basic Concepts and Study of the Existing:** presenting some basic concepts about queue management systems. Then, studying the specific needs of the Social Services Committee and analyzing current problems in queue management.
- **System Modelling and Design:** Creating a comprehensive UML diagrams for the proposed automated system, including user interface, database, and queue management mechanisms.
- **System Development:** Creating a web system using suitable programming languages and integrating different components.

Chapter 1

Basic Concepts and Study of the Existing

1.1 Introduction

The increase in the variety and abundance of services across all aspects of life, along with the growing number of customers seeking these services, has compelled companies and institutions to organize these customers in queues. With the advancement of automation, the organization of these queues has shifted to automated systems. In this chapter, we will discuss the definition of queue management systems, their types, and provide examples. This will offer a comprehensive understanding of how different queuing models operate and the benefits they bring to enhancing customer service efficiency.

1.2 Queue management definition:

Before looking to know how to manage waiting lines, we must talk about some basic concepts related to it:

1. **Queue:** A queue is a line or sequence of people, objects, or tasks awaiting their turn for service, processing, or completion. In economic terms, a queue typically represents demand exceeding supply, resulting in individuals or entities waiting for their needs to be fulfilled. Queues can manifest in various contexts, such as retail checkout lines, customer service counters, or processing systems, and are often managed through structured systems or protocols to optimize efficiency and minimize wait times.
2. **Desk:** In the context of queue management systems, a desk refers to a specific point of interaction where customers or individuals address their needs or concerns with service providers. It is typically a designated area within a service facility, such as a counter or reception desk, where customers receive assistance, information, or transactions are conducted. Each desk is usually identified by a unique identifier (e.g., Desk 1) and may be associated with a specific staff member to facilitate efficient service delivery. Desks play a crucial role in organizing and directing customer flow within the queue management system, ensuring that individuals are served promptly and effectively.

3. **Ticket:** In the context of queue management systems, a ticket is a numbered document or token provided to individuals to signify their place in line and their turn for service. Each ticket is typically associated with a specific queue and service session, providing essential information such as the name of the queue, the type of service requested, and a unique identifier. Tickets may be dispensed electronically or manually through ticket dispensers, and they serve to organize and manage the flow of customers or individuals awaiting service. Additionally, tickets may include features such as estimated wait times and notifications to keep customers informed about their position in the queue and when their turn is approaching.
4. **Service session:** In the context of queue management systems, a service session refers to the designated period during which a ticket remains valid for receiving service. This time frame typically aligns with the operating hours of the service facility or the availability of service providers. Tickets obtained during a service session are considered valid for that session only, ensuring fairness and efficiency in queue management. Service sessions help regulate customer flow by providing clear boundaries for when tickets are valid, preventing backlog and ensuring that customers are served within reasonable time frames. Additionally, service sessions may be extended or adjusted by service administrators as needed to accommodate fluctuations in demand or operational constraints.

Therefore we can have different definitions and meaning of the queue management system or waiting lines system, where the most common of them can be summarized as follows:

1. A queue management system is the organization of queues of people within a retail or public sector department. It can be either reactive through a system that can organize the existing queue or proactive through queue management statistics gathering system, so that the trends can be identified and anticipated [1] .
2. From another point of view, the queue management system considers as a mathematical models for the operations research and this prototype aims to visualize the objective reality of a particular situation and summarize its dimensions to study and analyze it and then take the right decision [2] .
3. The queue management theory depends on units arrival such as clients, machines...etc to the service stations, where the arrival operation can be with a fixed rate in a known period of time, or with a randomly [3].
4. Queuing theory is the systematic analysis of standing in line and is a whole field of operations management discipline. The theory of queuing uses statistical models and performance metrics to analyze and ideally enhance customer flow through a queuing framework[4] .
5. A Queue Management System is a software-based solution designed to efficiently manage customer flow and reduce waiting times in service-oriented environments. It typically includes features such as queue monitoring, ticketing, appointment scheduling, staff management, and real-time analytics to optimize resource allocation and enhance the overall customer experience[5].

1.3 Types of queue management

Let's explore different types of queue management systems based on two key criteria: the number of channels (or servers) and the number of operation phases. Channels represent the stations accessing the service, while stages indicate the steps required to complete the service. These criteria can each take single or multi values, resulting in four distinct forms of queue management when combined.

1. Single channel, single-phase:

A single-channel, single-phase queueing system refers to a specific type of queuing model characterized by having only one service channel and one phase of service. In other words, there is a single server attending to customers, and each customer undergoes a single phase of service before leaving the system.

Key characteristics of a single-channel, single-phase queueing system include:

- (a) Single Server: There is only one service point or server available to assist customers.
- (b) One Phase of Service: Each customer receives service for a single task or transaction before exiting the system. There are no multiple stages of service or additional processing steps.
- (c) Waiting Line: Customers may queue up in a waiting line if the server is busy attending to other customers. The order in which customers are served typically follows a first-come, first-served (FCFS) discipline.
- (d) Arrival and Service Rates: The system operates based on the rates at which customers arrive and are serviced. These rates influence the overall performance and efficiency of the queueing system.

Single-channel, single-phase queueing systems are commonly encountered in various service environments, such as retail checkout counters, bank teller windows, and customer service desks. Analyzing and understanding the behavior of such systems is essential for optimizing service levels, minimizing wait times, and improving customer satisfaction[6]. The following figure clarifies this model:

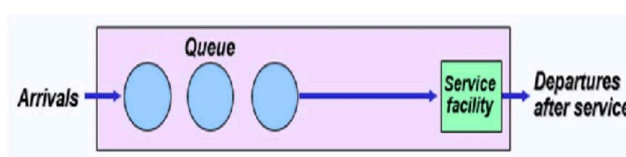


Figure I.1: a single channel, single-phase model

2. Single Channel, Multi-Phase:

A single-channel, multi-phase queueing system is a queuing model where customers go through multiple service phases or stages before completing their service and exiting the system. Despite having only one service channel (or server), customers undergo distinct phases of service, each addressing different aspects of their needs or tasks.

Key features of a single-channel, multi-phase queueing system include:

- (a) Single Server: There is only one service point or server available to assist customers, which means that only one customer can be served at a time.
- (b) Multi-Phase Service: Each customer goes through multiple phases of service, addressing different aspects of their request or transaction. For example, in a bank setting, a customer may first receive assistance with a deposit, followed by a withdrawal, and then account inquiries.
- (c) Sequential Processing: Customers move through the different service phases in a sequential order, typically following a predetermined process flow.
- (d) Waiting Line: If the server is busy assisting another customer, incoming customers may have to wait in a queue until they can begin their service process.
- (e) Service Time Variation: The duration of each service phase may vary depending on the complexity of the task or transaction involved, as well as the specific needs of each customer.

Single-channel, multi-phase queueing systems are commonly found in service environments where customers require comprehensive assistance or where transactions involve multiple steps. Examples include healthcare facilities with triage processes, government offices handling permit applications, and customer support centers addressing technical issues. Optimizing such systems involves managing both the efficiency of individual service phases and the overall flow of customers through the system[7]. the following figure gives us more details:

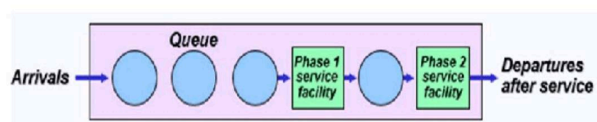


Figure I.2: a single channel, the multi-phase model

3. Multi-Channel, Single Phase:

A multi-channel, single-phase queueing system is a queueing model characterized by having multiple service channels (servers) available to assist customers, with each customer undergoing a single phase of service before leaving the system.

Key features of a multi-channel, single-phase queueing system include:

- (a) Multiple Service Channels: There are multiple service points or servers available to assist customers simultaneously. This means that several customers can be served concurrently, reducing wait times and increasing system capacity.
- (b) Single Phase of Service: Each customer undergoes a single phase of service, completing their transaction or task before exiting the system. There are no multiple stages of service or additional processing steps.
- (c) Parallel Processing: Customers are served independently at each service channel, allowing for parallel processing of service requests. This helps distribute the workload evenly across multiple servers and speeds up overall service delivery.

- (d) **Waiting Line Management:** If all service channels are occupied when a new customer arrives, they may have to wait in a queue until a channel becomes available. The order in which customers are served typically follows a first-come, first-served (FCFS) discipline.
- (e) **Service Time Variation:** The duration of service for each customer may vary depending on factors such as the complexity of the task, the skills of the service personnel, and the specific needs of the customer.

Multi-channel, single-phase queueing systems are commonly found in various service environments, such as retail checkout counters with multiple cashiers, call centers with multiple phone lines, and self-service kiosks at airports or train stations. Optimizing such systems involves managing queue lengths, balancing workload across channels, and ensuring efficient utilization of available resources to enhance customer satisfaction[8].

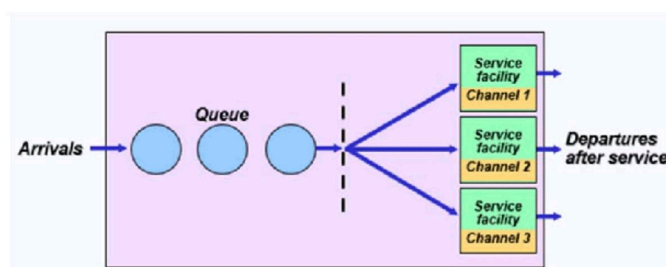


Figure I.3: multi-channel, single phase

4. Multi-Channel, Multi-Phase:

A multi-channel, multi-phase queueing system is a queuing model characterized by having multiple service channels (servers) available to assist customers, with each customer undergoing multiple phases of service before completing their transaction and exiting the system. This type of system offers both parallel processing through multiple service channels and sequential processing through multiple service phases.

Key features of a multi-channel, multi-phase queueing system include:

- (a) **Multiple Service Channels:** There are multiple service points or servers available to assist customers concurrently, allowing for parallel processing of service requests. This helps distribute the workload evenly and reduces overall wait times.
- (b) **Multiple Phases of Service:** Each customer undergoes multiple phases or stages of service, addressing different aspects of their request or transaction. These phases may include initial assessment, data collection, processing, verification, and completion.
- (c) **Parallel and Sequential Processing:** Customers may be served simultaneously at different service channels, while each customer progresses through the various service phases sequentially. This combination of parallel and sequential processing enhances efficiency and flexibility in service delivery.

- (d) Waiting Line Management: If all service channels are occupied or if a service phase requires more time, customers may have to wait in a queue until a channel or phase becomes available. Effective queue management strategies are essential to minimize wait times and optimize customer flow.
- (e) Service Time Variation: The duration of each service phase may vary depending on factors such as the complexity of the task, the skills of the service personnel, and the specific needs of the customer. Balancing service times across channels and phases is crucial for maintaining overall system efficiency.

Multi-channel, multi-phase queueing systems are commonly found in complex service environments such as healthcare facilities, government agencies, and financial institutions. Optimizing such systems requires careful planning, resource allocation, and performance monitoring to ensure smooth operation and enhance customer satisfaction[9].

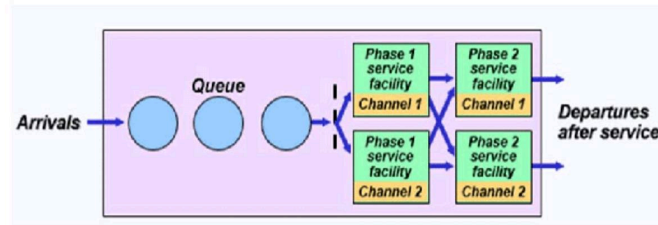


Figure I.4: multi-channel, multi-phase

1.4 Examples queue management system

There are several solution providers offering queue management systems. However, some of the solution providers stand out due to the high-quality standards of their products and exceptional service quality.

Here is a comprehensive list of the best queuing system providers[10]:

1. Wavetec

Since its establishment in 1986, Wavetec has established a strong reputation as a dependable provider of solutions for global businesses. Today, it stands as a prominent leader in queue management systems, aiding numerous businesses and service providers in optimizing customer flow and elevating customer satisfaction.

Wavetec's queue management solution is intricately crafted to enable businesses to handle customer queues with precision and efficacy. It encompasses a holistic package comprising hardware, software, and support services to ensure seamless queue management operations.

Top Features:

Here is a comprehensive list of features offered by Wavetec's Queuing System:

- **Self-Service Check-in:** Visitors can check in through the take a number machine kiosk or via QR code scanning.
- **Automated Notifications:** Third-party integration with WhatsApp enable customers to receive notifications.
- **Appointment Booking:** Appointment booking solutions enable customers to schedule their appointments remotely.
- **Hardware Support:** Customers can print their own tickets. Moreover, the LCD screens in the waiting room show approximate wait times for customers.
- **Dashboards:** Data can be accessed by the managers, enabling them to monitor queues and manage branch performance remotely.
- **Analytical Tools:** Data can be analyzed to understand the customer experience, optimize resource allocation and improve branch performance.
- **Modern Design:** The sleek and modern hardware design complements the aesthetics of the service centers.
- **Customer Feedback:** Track and manage customer journeys by receiving feedback from customers in real time.

Wavetec's hardware is engineered for durability, capable of enduring heavy usage over time. Their queuing software is user-friendly and adaptable, allowing customization to accommodate the specific requirements of individual businesses.

Main Disadvantages:

- **High Initial Cost:** The comprehensive package including hardware and software can be expensive for small businesses.
- **Complex Setup:** Initial installation and customization may require specialized technical support.
- **Maintenance:** Regular maintenance of hardware components can be costly and time-consuming.
- **Dependence on Internet:** Cloud-based features and remote monitoring require a stable internet connection, which may be a limitation in areas with poor connectivity.

To summarize, Wavetec offers a comprehensive, self-contained, and adaptable solution that caters to businesses of varying scales. Their premier queue system's outstanding quality, along with exceptional customer support, distinguishes Wavetec in the market.

WAVETEC



Figure I.5: ATLAS Wavetec Queue management system

2. Qmatic

With a track record of delivering exceptional queue management software and hardware over the years, this provider has consistently enhanced queue flow and elevated customer experiences. As a result, Qmatic secures the second spot on our list of top solution providers.

Top Features

Here are some of the most remarkable features of Qmatic's queue management systems:

- **Hardware solutions:** Qmatic's solutions come with custom hardware including printers, self-service queuing system software-kiosks, and LCD's.
- **Real-Time Reporting:** Data is recorded and reported in real-time, enabling service centers to ensure an efficient service.
- **Automated Notification System:** Customers can receive updates on the wait times via SMS, display LCD, or audio announcements.
- **Third-party integrations:** This system can get integrated with tools such as Salesforce, Oracle, or WhatsApp.

Main Disadvantages:

- **High Cost:** The custom hardware solutions can be expensive to purchase and maintain.
- **Integration Complexity:** Integrating with third-party tools like Salesforce and Oracle can be complex and may require technical expertise.
- **Scalability Issues:** The system may face challenges when scaling up for larger enterprises with very high customer flow.
- **User Training:** Employees may require extensive training to efficiently use the system, which can incur additional costs.

Qmatic's queue management system empowers businesses to enhance customer service by enabling customers to book appointments and join queues virtually using mobile devices. This innovative approach reduces wait times and enhances customer journeys.

QMATIC



Figure I.6: Qmatic Queue management system

3. Qtrac

Qtrac has emerged as a notable provider of queue management solutions, specializing in a virtual approach. Their cloud-based virtual queue management system caters to both small businesses and large enterprises, facilitating timely customer interactions.

Top Features:

Qtrac's top features include the following:

- **Dashboard:** Customer data is available on the dashboard, facilitating data-driven decision-making.
- **Virtual Queuing:** Customers can join virtual queues by scanning a QR code or simply by sending an SMS from their mobile devices.
- **Schedule Appointments:** Customers can book their appointments remotely.
- **Notification system:** Customers can receive notifications that update them regarding their turn in the queue.

Main Disadvantages:

- **Limited Hardware Support:** As a primarily virtual solution, Qtrac might not provide sufficient hardware options for businesses needing physical queue management tools.
- **Internet Dependence:** The cloud-based system relies heavily on internet connectivity, which can be a drawback in locations with unreliable internet service.
- **Customization Limitations:** While flexible, Qtrac's customization options might not meet the specific needs of all businesses.
- **Initial Setup Complexity:** Setting up the virtual queue system and ensuring it integrates well with existing processes can be complex and time-consuming.

In essence, Qtrac's queue management solutions equip businesses with a comprehensive set of tools to effectively handle queues, monitor customer wait times, streamline customer flow, and elevate the overall customer experience.

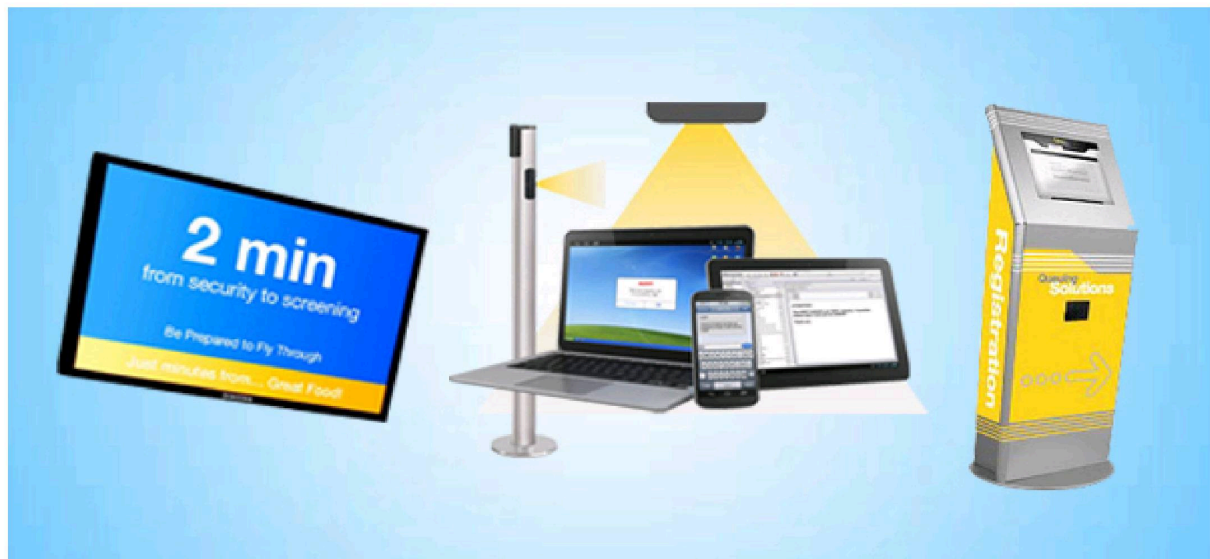


Figure I.7: Qtrac Queue management system

4. Qminder

Committed to efficiently managing foot traffic, Qminder has created and implemented effective solutions for businesses to manage customer flow. This standalone, cloud-based system optimizes customer flows, reducing lengthy waiting lines.

Top Features:

Here are some of the most prominent features of Qminder's queue management solution:

- **Visitor Check-In:** The customers that enter the service center can join queues by using the self-service kiosk or by scanning a QR code.
- **Notification System:** Updates can be sent to customers to update them regarding their turn while they wait in line.
- **Dashboard:** Integration with the dashboard enables businesses to store customer data and thus improve their decision-making abilities.

Main Disadvantages:

- **Limited Physical Queue Management:** Qminder is primarily cloud-based, which might not cater well to businesses needing robust physical queuing solutions.
- **Internet Reliance:** As a cloud-based solution, it requires a stable internet connection for optimal performance.
- **Cost for Advanced Features:** Accessing advanced features and analytics might require additional costs, which can be a burden for small businesses.
- **Technical Support Dependence:** Users may require ongoing technical support to handle any issues, which can be an additional expense.

Qminder's user-centric solutions have benefited numerous businesses over time, providing a comprehensive approach to effectively managing queues.



Figure I.8: Qminder Queue management system

5. Qwaiting

Qwaiting stands out as another renowned solution provider, catering to both small businesses and large enterprises with their queue management solutions. Their cloud-based software effectively streamlines the customer journey.

Top Features:

- **Dashboard:** Customer data is tracked and stored on a dashboard.
- **Appointment Booking:** Customers can book their appointments remotely, conveniently.
- **Visitor Check-In:** users can generate their own tickets and join a queue.
- **Simple User Interface:** This solution has a user-friendly interface, making it an easy-to-use option for all.

Main Disadvantages:

- **Internet Dependence:** The cloud-based system's functionality is highly dependent on stable internet connectivity.
- **Scalability Issues:** The system may face difficulties in scaling up to accommodate larger businesses or enterprises.
- **Integration Complexity:** Integrating Qwaiting with existing systems and tools can be complex and may require technical assistance.
- **Cost of Customization:** Customizing the system to fit specific business needs can incur additional costs.

Qwaiting offers highly customizable solutions that seamlessly integrate with various systems, providing businesses with versatility to enhance their queue management and customer experience.

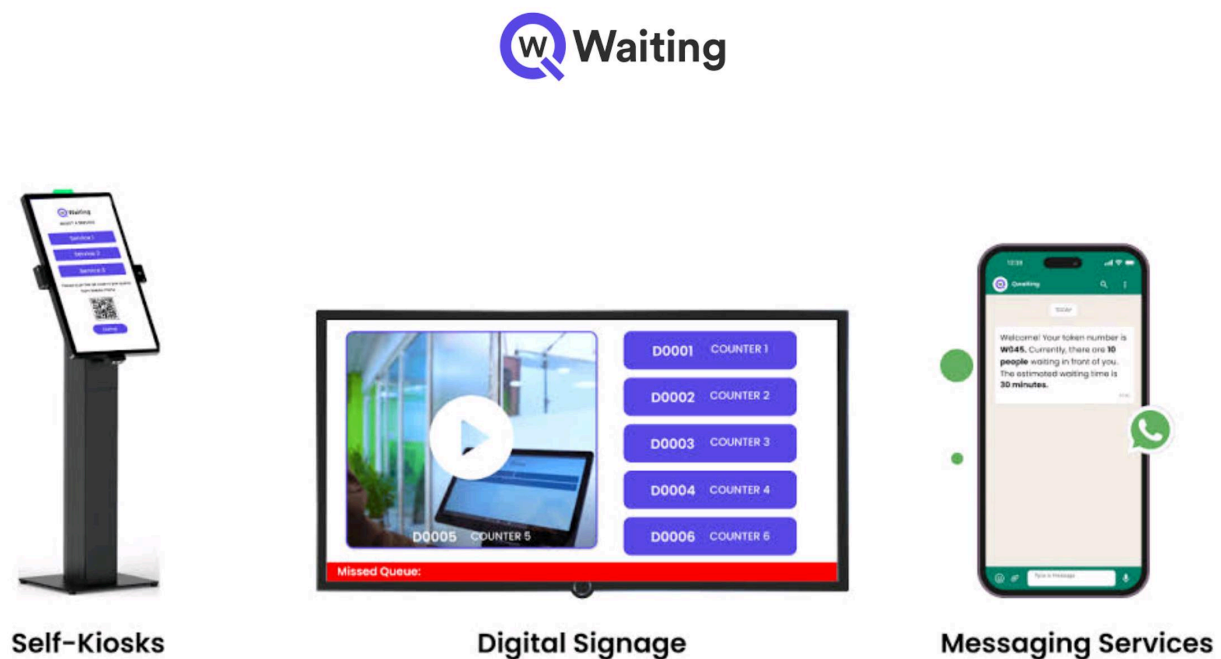


Figure I.9: Qwaiting Queue management system

6. Verint

For those seeking efficient customer flow management solutions, Verint's queue management systems present a compelling option. With a comprehensive range of features, their software simplifies service line management for service providers.

Top Features:

Here is a list of Verint's top features:

- **Virtual Check in:** Visitors can join queues either from the service center or remotely.
- **Schedule Appointments:** Customers can book their appointments in advance.
- **Analytics and Reporting:** Customer data can be monitored and stored to inform data-driven decision-making.

Main Disadvantages:

- **High Implementation Cost:** The comprehensive solution can be costly to implement, especially for small businesses.
- **Complex Integration:** Integrating with existing business processes and systems can be complicated and may require significant technical support.
- **Maintenance Requirements:** Regular maintenance and updates can be resource-intensive.
- **Learning Curve:** Employees may require significant training to effectively use the system, leading to additional costs and downtime.

In summary, Verint provides a customizable solution suitable for businesses and service providers to streamline the customer service process.

VERINT.

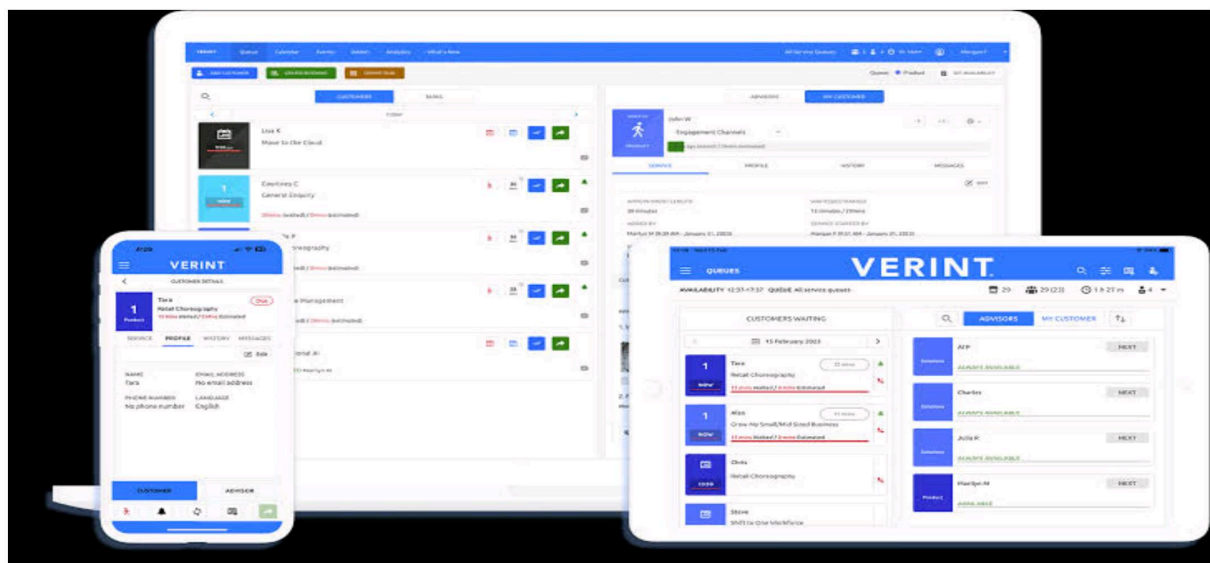


Figure I.10: Verint Queue management system

7. Qless

Qless has empowered businesses of various sizes with its robust queue management solutions.

Top Features:

This solution comes with multiple amazing features. The following are the top features:

- **Virtual Check-In:** Customers can join queues remotely.
- **Appointment Scheduling:** Customers can schedule appointments beforehand.
- **Data Analytics:** The solution provides real-time analytics and reporting tools.

Main Disadvantages:

- **Internet Dependency:** As a virtual solution, it requires a stable internet connection for seamless operation.
- **High Costs for Full Features:** Accessing the full range of features and real-time analytics can be expensive.
- **Customization Complexity:** Customizing the system to fit specific business requirements can be complex and require technical expertise.
- **User Training:** Extensive training may be needed for employees to use the system effectively, which can incur additional costs.

Qless offers a highly adaptable and scalable queue management solution, enabling businesses to enhance their customer experience and foster growth.

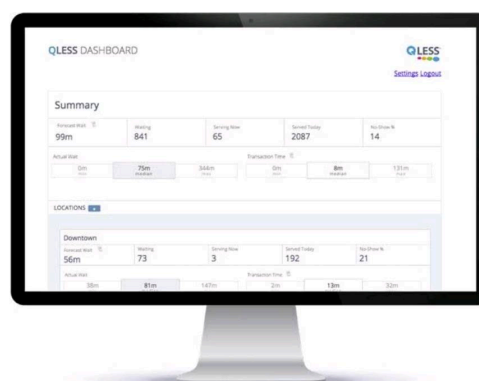


Figure I.11: Qless Queue management system

1.5 The problem statement

1.5.1 The university Social Services Committee

The Social Services Committee is a university-level body with a president and a vice president who succeeds them. The committee is responsible for managing social service activities, which aim to improve and enhance the livelihoods of professors, employees, and workers both materially and spiritually through supplementing their income. It provides services in the fields of health, housing, basic necessities, culture, entertainment, and generally all social measures targeting the daily life of the employee or worker and their families to enable them to improve their productivity or job performance[11][12].

1.5.2 Fields of social services

The social services provided by the employing authority and the integrated entities of the state, local communities, and specialized agencies, in accordance with applicable legislation and regulations, are directed towards the following areas funded by social services:

1. Social assistance.
2. Healthcare services.
3. Maternity and childcare centers.
4. Public sports facilities.
5. Entertainment and cultural activities.
6. Activities aimed at developing popular tourism.
7. Family recreation centers.
8. Leisure centers.
9. Holiday resorts.
10. Financing of the pre-retirement pension system[11][12].

1.5.3 Description of the problem and some previous solutions

Due to the large number of tasks undertaken by the Social Services Committee in all areas, as well as the high number of beneficiaries of these services, there is always a waiting queue. The size of this queue is often related to seasonal events and services, such as the back-to-school backpack distribution at the beginning of the academic year and the Ramadan food basket distribution during Ramadan.

Several solutions have been proposed by the committee:

1. The first solution was to implement a ticketing system at the entrance, where clients would take a ticket upon arrival. When a social service employee calls out the ticket number, the client would approach to receive the desired service. However, there was a problem with this solution, as clients could obtain multiple tickets and

distribute them to others. Consequently, a client could arrive late, yet still obtain an advanced position in the queue by acquiring tickets from earlier clients. This unfairly disadvantaged those waiting longer.

2. The second solution involved dividing beneficiaries of social services according to their faculties and establishing an office for each group of faculties. This was aimed at reducing the pressure on the main office. While this alleviated pressure during regular circumstances and services, during seasonal events like the back-to-school backpack distribution and Ramadan food basket distribution, the problem persisted, and the committee faced significant pressure from beneficiaries.

1.6 Project overview

Our project proposes an automated queue management solution designed to streamline and optimize the customer service process. Based on the sixth example, the Verint system, our queue management system will be implemented as a web-based application with multiple interfaces accessible through any browser. This approach minimizes both development and deployment costs, ensuring compatibility across all operating systems with a browser.

The system operates with a central server that hosts the website's source code and database. This server distributes the various pages of the system to connected computers via a local area network (LAN). Each webpage serves as an interface within the queue management system, providing predefined functions to users.

Advantages of Our Queue Management System

Compared to existing queue management systems, our solution offers several notable advantages:

- **Validation of Customer ID:** Tickets can only be printed if the customer ID is correct, reducing the risk of overbooking and ensuring only legitimate customers are served.
- **One Ticket per Customer:** The system ensures that a customer can only hold one ticket at a time in the queue, preventing duplicate tickets and streamlining the queue process.
- **Service Session Recording:** Every interaction between a customer and a committee worker is logged, providing a detailed record of service sessions for better tracking and accountability.

These features enhance the efficiency and effectiveness of the queue management process, offering a reliable and user-friendly system that addresses common issues faced by traditional queue management solutions.

1.7 Conclusion

In summary, queue management systems have become essential in efficiently managing the flow of customers across various service environments. By understanding the basic concepts of queuing, including the roles of queues, desks, tickets, and service sessions, organizations can better appreciate the different types of queue management systems available. From single-channel, single-phase models to multi-channel, multi-phase systems, each type offers distinct advantages and applications. Additionally, examples of leading queue management system providers highlight the practical implementations and benefits of these systems. As technology continues to advance, the adoption of sophisticated queue management solutions will likely become even more integral to optimizing customer service and satisfaction.

Chapter 2

System Modelling and Design

After thoroughly studying the current system, we have gathered extensive information and conducted a comprehensive analysis. This process has equipped us with the necessary insights to envision the future system. We now transition to the next phase, where we will design an information system based on the following steps:

2.1 Introduction

2.1.1 The importance of the Unified Modeling Language and its significance

The Unified Modeling Language (UML) was introduced in the 1990s by software engineers Grady Booch, Ivar Jacobson, and James Rumbaugh. Their goal was to create a standardized method for representing the complexities of software development, distinguishing the methodology from the process. Today, UML remains the industry standard for developers, project managers, business owners, technical stakeholders, and professionals across various fields, providing a universal language for modeling software systems.

2.1.2 The benefits of Unified Modeling Language

- **Simplifying Complexities:** UML helps break down complex systems into manageable components, making it easier to understand and design intricate software architectures.
- **Maintaining Open Lines of Communication:** By providing a common visual language, UML ensures that developers, project managers, and stakeholders can communicate effectively and avoid misunderstandings.
- **Automating Software and Process Production:** UML supports automation in the development process, enabling the generation of code, documentation, and other artifacts directly from UML diagrams, thus streamlining the workflow.
- **Assisting in Continuous Architectural Problem-Solving:** UML facilitates ongoing evaluation and refinement of software architecture, allowing teams to identify and address issues early in the development cycle.

- **Increasing the Quality of Work:** By standardizing the design process and promoting best practices, UML contributes to higher quality software, with fewer errors and more robust functionality.
- **Reducing Costs and Time to Market:** Efficient communication, automation, and improved quality all contribute to faster development cycles and lower costs, enabling quicker delivery of products to the market.

2.1.3 Types of Unified Modeling Language Diagrams

In the realm of Unified Modeling Language (UML), diagrams serve as powerful tools for conceptualizing and communicating complex systems. The two primary categories of UML diagrams are behavioral and structural, each catering to distinct aspects of system analysis and design.

Behavioral diagrams delve into the dynamic aspects of a system, depicting the interactions and behaviors among its components. These diagrams include use case diagrams, activity diagrams, state machine diagrams, and sequence diagrams, among others. They are instrumental in illustrating the functionality and operational flow of the system, enabling stakeholders to grasp its behavior under various scenarios.

On the other hand, structural diagrams focus on the static structure of a system, elucidating its components, relationships, and organization. Examples of structural diagrams encompass class diagrams, object diagrams, component diagrams, and deployment diagrams. These diagrams provide insights into the architecture and composition of the system, facilitating discussions on its design and implementation.

Tailored to meet the diverse needs of stakeholders ranging from clients and project managers to developers and quality assurance personnel, each UML diagram serves a unique purpose, offering a specialized perspective on the system. Through clear and concise visual representations, UML diagrams strive to foster understanding and collaboration across multidisciplinary teams, ensuring that all stakeholders share a common understanding of the system's architecture and behavior.

2.1.4 The Unified Modeling Language diagrams used in our project study

- **Behavioral Diagrams:** These diagrams spotlight the dynamic aspects of a software system or process, showcasing its functionalities and emphasizing the expected behavior in the designed system. The diagrams we will utilize include:
 - **Use Case Diagram:** Describes the system's functionalities and actors' interactions. It visually represents sequences of events to fulfill specific processes within the system.
 - **Sequence Diagram:** Illustrates the sequence of messages and interactions between actors and objects in chronological order. It is effective for depicting multi-tasking scenarios.
 - **Activity diagram:** is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.

- **Structural Diagrams:** These diagrams represent different levels of abstraction and implementation within a system, such as a database or application. They visualize the hierarchical arrangement of components and their interactions. Structural diagrams help ensure that all system parts function together cohesively, providing a clear guide for the system's architecture and integration.

- **Class Diagram:** Depicts the logical and actual design of the system by displaying its classes, attributes, and methods. It provides a visual representation of class interconnections and structures.

2.2 Analysis

2.2.1 Actors

The involved parties in this system are as follows:

- **Administrative:** he control the database, he add to the database new employee (customer), or delete a employee.
- **committee worker:** he can call customer in the queue, if the customer absent he can hold it , if he completes his session with with customer he can call next customer.
- **customer:** he can just get his ticket in the queue.

2.2.2 Task Specification for Each Actor in the Queue Manager System

Customer

- **Get Ticket:**
 - **Description:** The customer requests a ticket from the system.
 - **Operations:**
 1. Request ticket page.
 2. Provide ID.
 3. Receive verification message (success or failure).
 4. Get a new ticket added to the queue if the ID is correct.

Committee Worker

- **Login:**
 - **Description:** The committee worker logs into the system.
 - **Operations:**
 1. Request login page.
 2. Provide ID and password.
 3. If validation is true, return to the page of the role.

4. If validation is false, refuse access.

- **Calling Customer:**

- **Description:** The committee worker calls a customer for service.

- **Operations:**

1. Call the customer.
2. Wait for the customer to come.
3. Serve the customer.
4. Hold the number if the customer is not coming.
5. Insert the information of service session to the database.
6. Call the next customer.

- **Store Service Session:**

- **Description:** The committee worker stores service session information for the customer.

- **Operations:**

1. The committee worker enters the customer's service session information.
2. Service session information includes the type of service and the amount benefited (additional information can be written with it).

Administrative

- **Add Employee:**

- **Description:** The administrative adds a new employee to the system.

- **Operations:**

1. Add a new employee to the database.
2. Receive result message.

- **Delete Employee:**

- **Description:** The administrative deletes an existing employee from the system.

- **Operations:**

1. Delete an employee from the database.
2. Receive result message.

- **Login:**

- **Description:** The administrative logs into the system.

- **Operations:**

1. Request login page.
2. Provide ID and password.
3. Receive login validation (true or false).
4. If validation is true, return to the page of the role.
5. If validation is false, refuse access.

System Operations

- **Login (Verify ID):**
 - **Description:** Verifies the ID provided by the customer.
 - **Operations:**
 1. Verify the ID during the ticket request process.
 2. Return success or failure message based on ID verification.
 3. if the ID is in the employees table and not in the queue verification is acceptable, otherwise verification is not acceptable
- **Calling Next Customer:**
 - **Description:** Automatically calls the next customer in line.
 - **Operations:**
 1. Extend from the calling customer operation.
 2. Call the next customer when the current one is served(Optional for committee worker).

2.3 Unified Modeling Language diagrams

2.3.1 Use Case Diagram

- Shows interactions between actors and the system's functionalities.
- Each operation and interaction is mapped to specific actors.

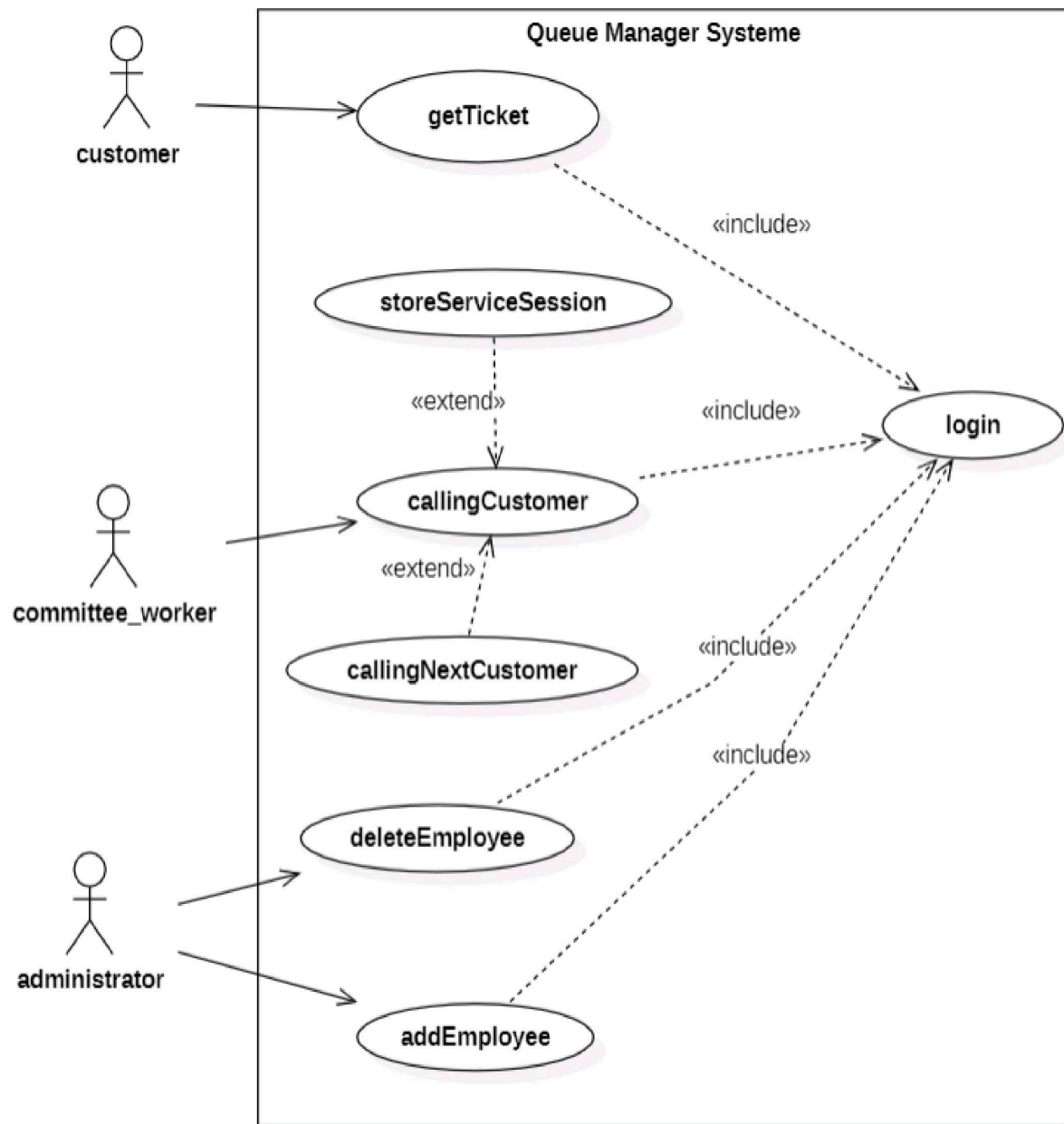


Figure II.1: Use Case Diagram for all system

2.3.2 Sequence Diagrams

- Illustrate the chronological flow of messages between objects or components in a system.
- Essential for visualizing and designing communication aspects of software systems.

Log in sequence diagram

The diagram below shows the login sequence for a committee worker or admin. It involves requesting the login page, submitting credentials, verifying these credentials against the database, and then either refusing access if the validation fails or granting access and returning the appropriate role page if the validation succeeds.

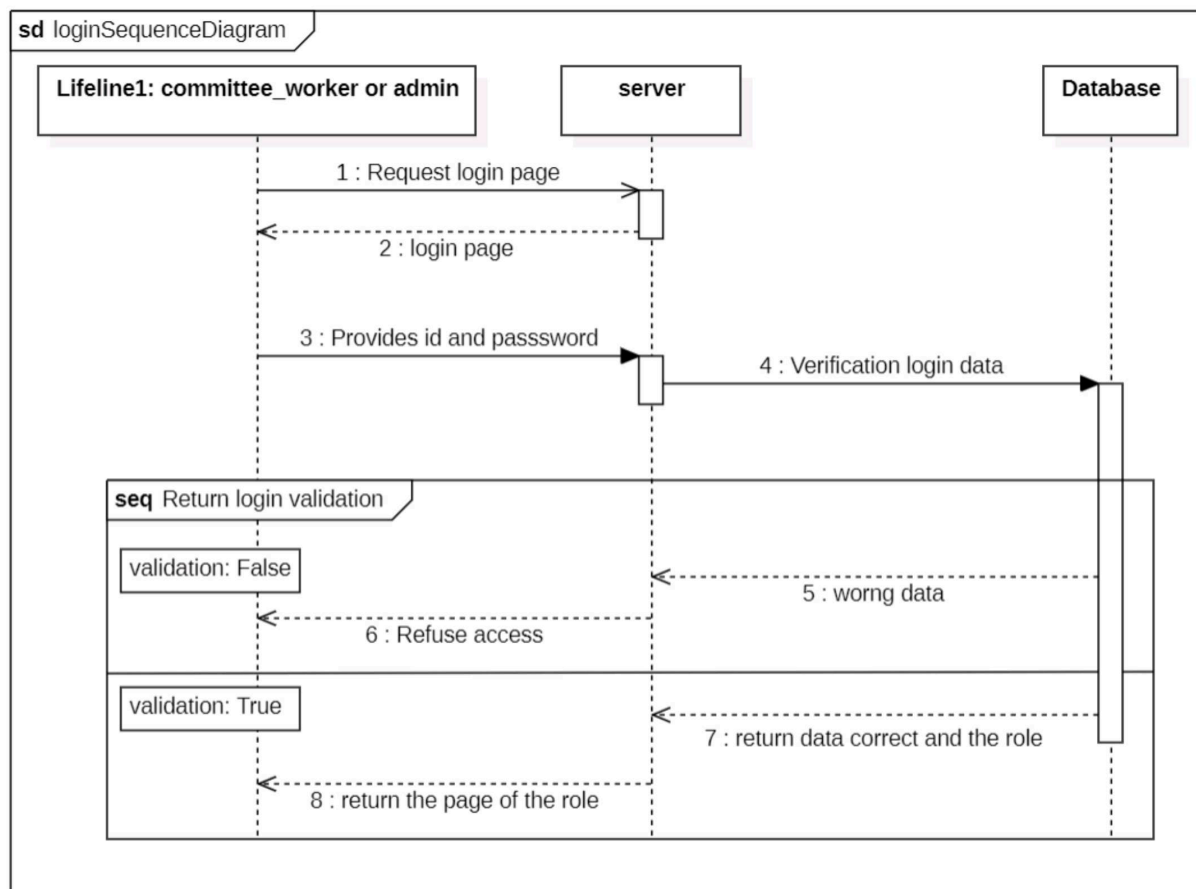


Figure II.2: Log in sequence diagram

Administrative sequence diagram

The diagram below represents the sequence of administrative operations in a system, where after a valid login, an administrator can add or delete employees. The process involves updating the server and database with the corresponding result messages after each operation.

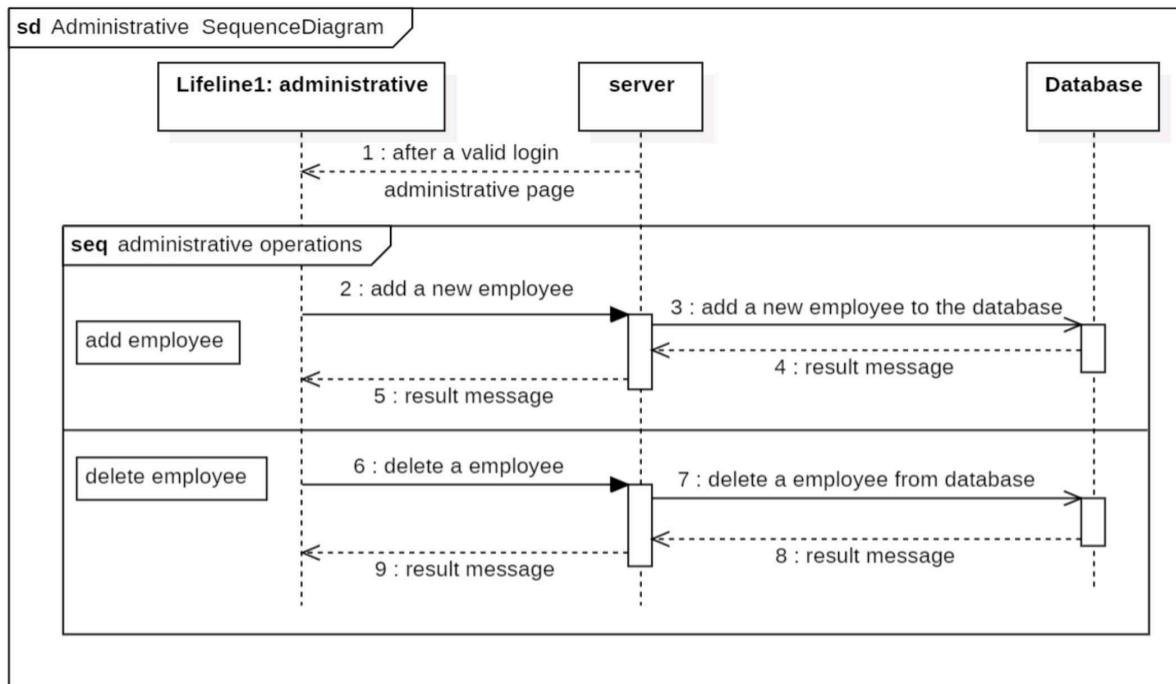


Figure II.3: Administrative sequence diagram

calling customer sequence diagram

The diagram below illustrates a sequence where a customer is called and served by a committee worker. The process involves calling the customer, waiting for their arrival, serving them, storing the service session, and then calling the next customer if the worker can serve.

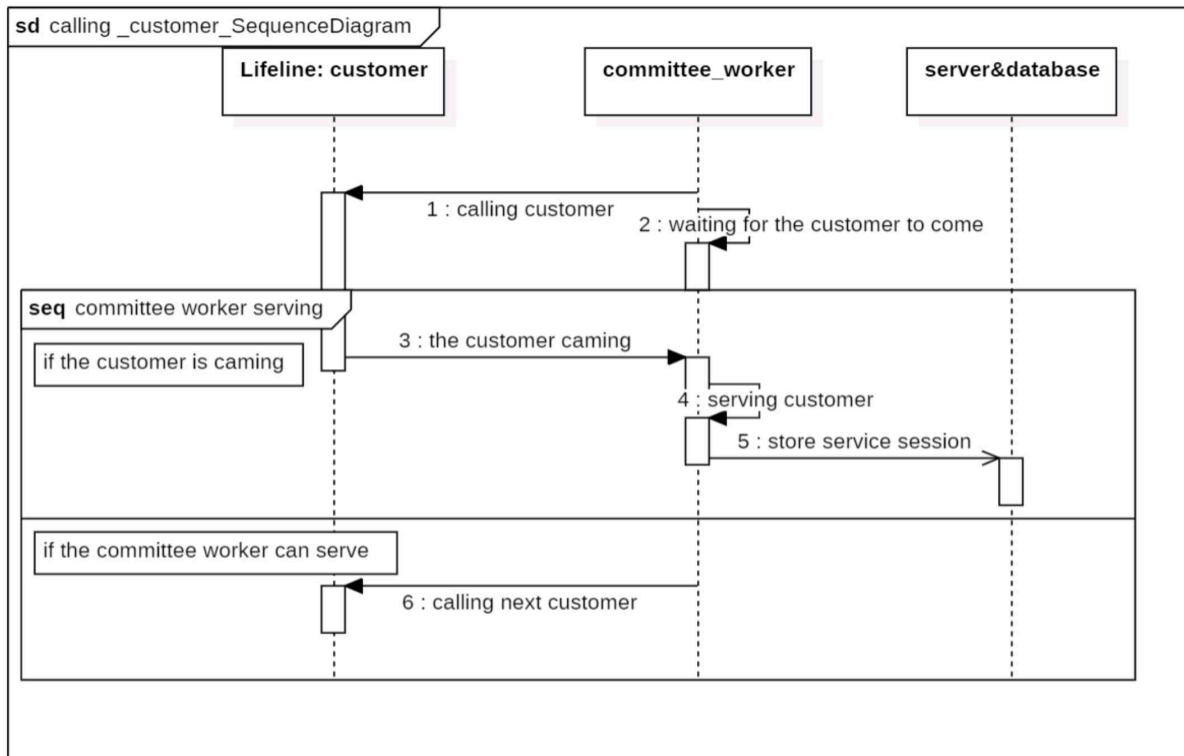


Figure II.4: calling customer sequence diagram

Get ticket sequence diagram

The diagram below illustrates the sequence for a customer to get a ticket. The customer requests the ticket page, provides their ID, which is then verified by the server and database. If the ID is correct and not already in the queue, a new ticket is added and printed; otherwise, an error message is returned.

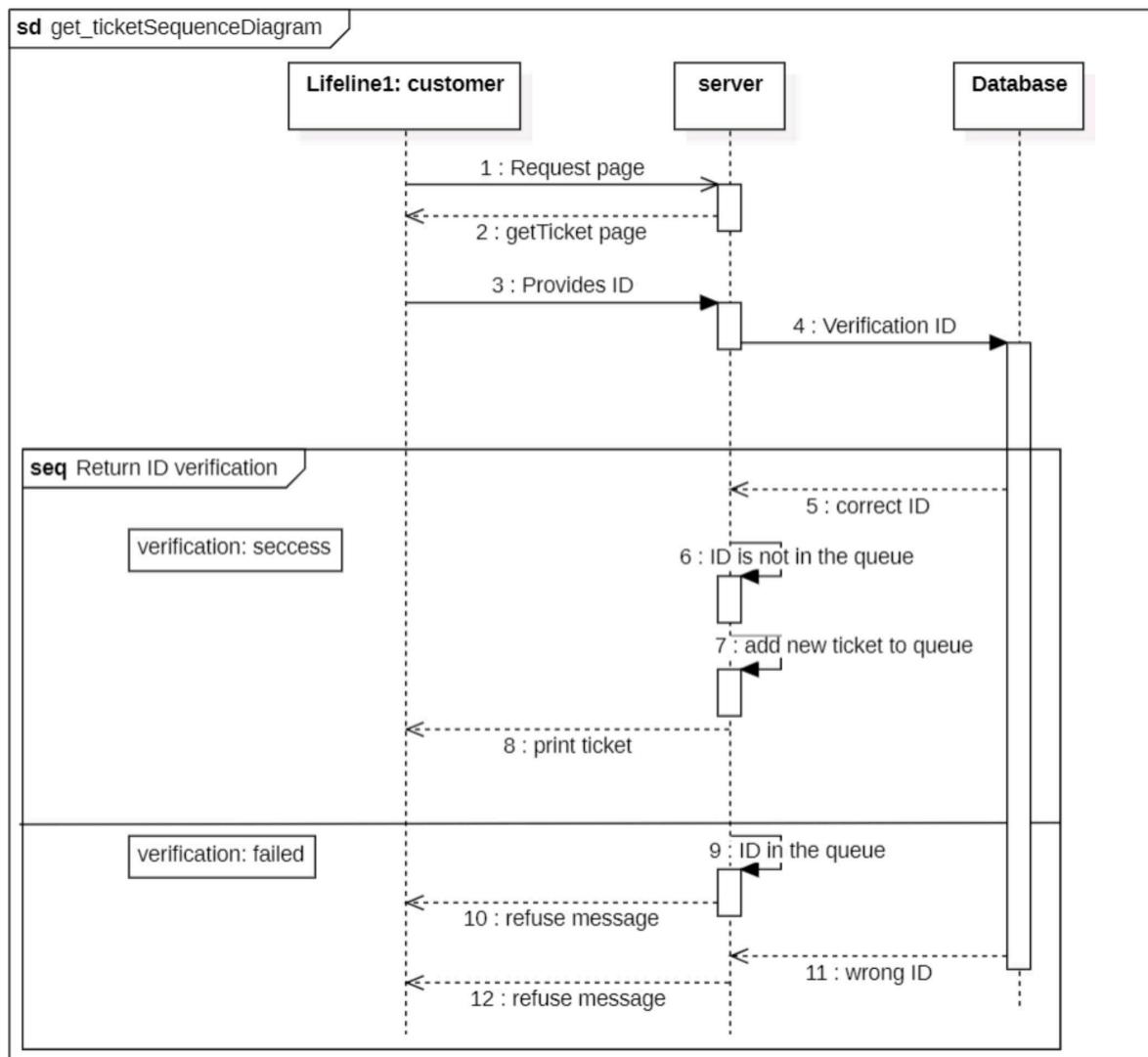


Figure II.5: Get ticket sequence diagram

2.3.3 Activity Diagrams

- Identify candidate use cases, through the examination of business workflows
- Model in detail complex activities in a high level activity Diagram.

The diagram below illustrates the activity flow for a customer obtaining a ticket and receiving service. It begins with the employee's arrival, entering an ID to get a ticket, verifying the ID, checking if the ID is in the queue, adding the customer to the queue if not present, waiting to be called, and then proceeding to the service session. If the customer does not show up, the ticket is deleted from the queue.

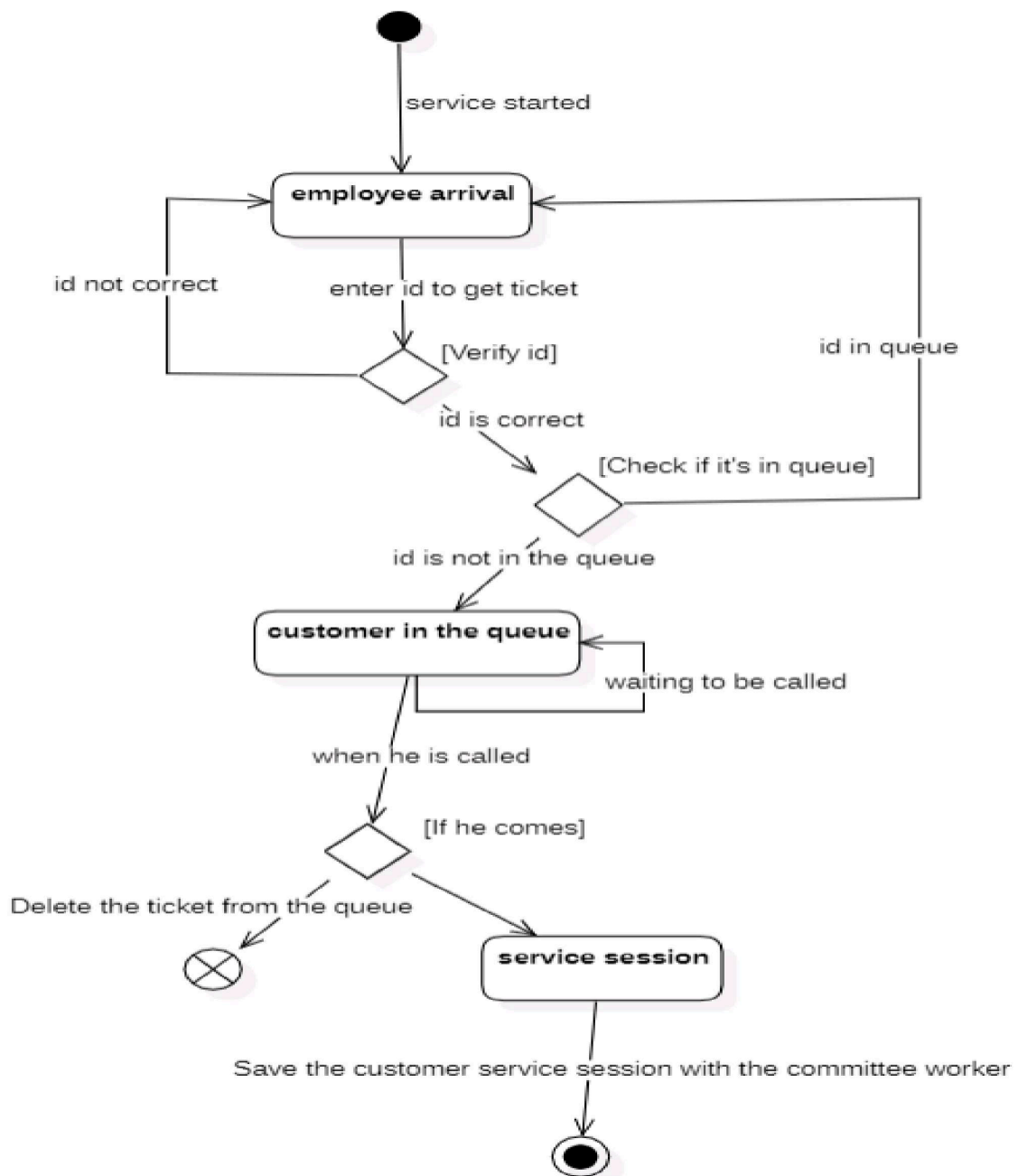


Figure II.6: Activity diagram of our project

2.3.4 Class Diagrams

- Defines the classes involved in the system and their attributes and methods.
- Shows relationships between different classes (Employee, Customer, Ticket, Administrative).

The diagram below is a class diagram for a queue management system, depicting the relationships and attributes of entities like Employee, Customer, Administrative, Committee Worker, Ticket, and Service Session. Each class has specific attributes and methods, illustrating the system's structure and interactions.

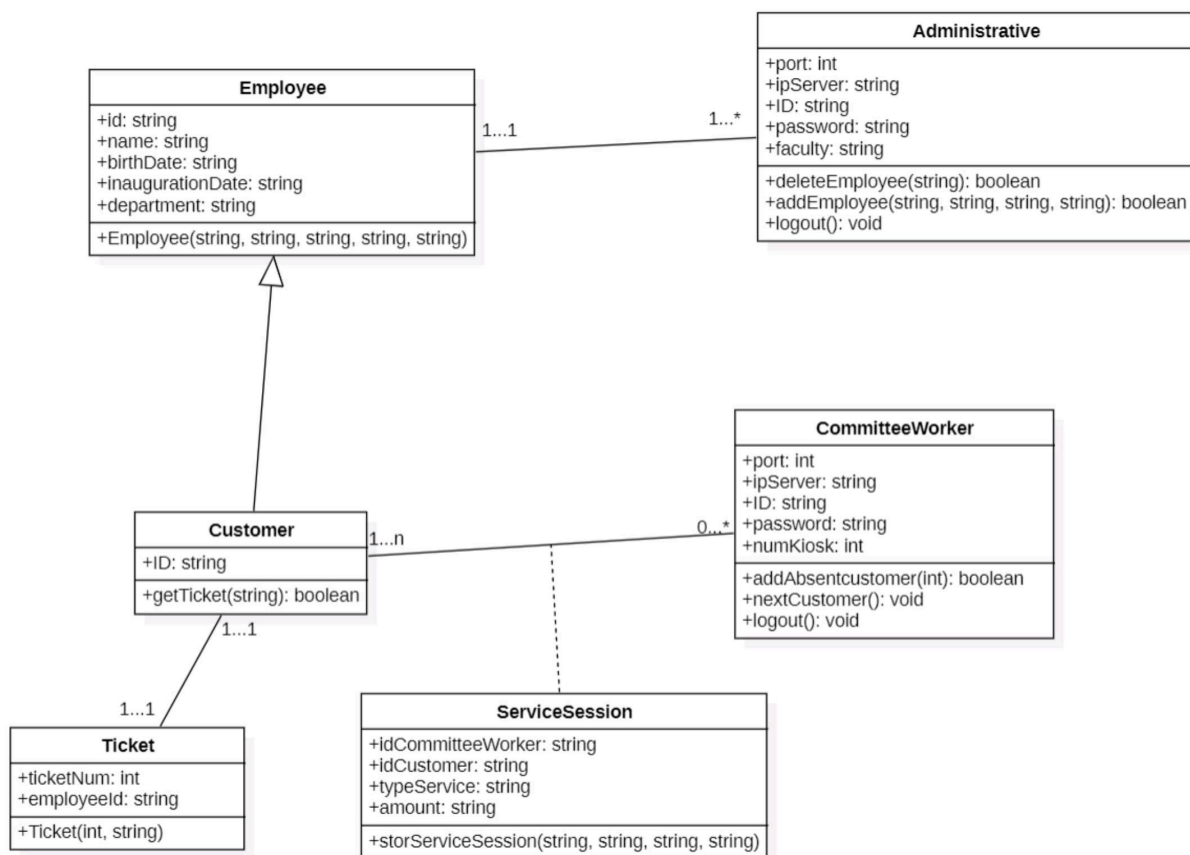


Figure II.7: Class diagram of our project

Each actor in the system has specific roles and operations they can perform, ensuring a smooth and efficient queue management process.

2.3.5 The data dictionary tables

Employees Table Description

Column Name	Data Type	Description
id	varchar(255)	Unique identifier for each employee.
first_last_name	varchar(255)	The first and last name of the employee.
date_of_birth	varchar(255)	The date of birth of the employee.
inauguration_data	varchar(255)	The date when the employee was inaugurated or joined the company.
department	varchar(255)	The department to which the employee belongs.
created_at	timestamp	The timestamp when the employee record was created.
updated_at	timestamp	The timestamp when the employee record was last updated.

Table II.1: Schema of the employees table

Login Table Description

Column Name	Data Type	Description
login_id	varchar(255)	Primary key and foreign key linking to the 'id' column in the 'employees' table.
password	varchar(255)	The password for the login.
role	varchar(255)	The role assigned to the login.
created_at	timestamp	The timestamp when the login record was created.
updated_at	timestamp	The timestamp when the login record was last updated.

Table II.2: Schema of the login table

Tickets Table Description

Column Name	Data Type	Description
id	bigint(20) UN-SIGNED	Primary key for the ticket, automatically incremented.
employee_id	varchar(255)	Foreign key linking to the 'id' column in the 'employees' table.
status	varchar(255)	The status of the ticket.
time_issued	timestamp	The timestamp when the ticket was issued.
time_served	timestamp	The timestamp when the ticket was served.
service_desk	varchar(255)	Identifier for the service desk associated with the ticket.
created_at	timestamp	The timestamp when the ticket record was created.
updated_at	timestamp	The timestamp when the ticket record was last updated.

Table II.3: Schema of the tickets table

Service Session Table Description

Column Name	Data Type	Description
id	bigint(20)	Primary key for the Service Session record.
id_customer	varchar(255)	Foreign key linking to the 'id' column in the 'employees' table, part of the composite key.
id_committee_worker	varchar(255)	Foreign key linking to the 'login_id' column in the 'login' table, part of the composite key.
date_serve	varchar(255)	The date of the service.
amount	varchar(255)	The amount associated with the service.
created_at	timestamp	The timestamp when the service session record was created.
updated_at	timestamp	The timestamp when the service session record was last updated.

Table II.4: Schema of the service_session table

2.4 Conclusion

In conclusion, adopting the Unified Modeling Language (UML) proved to be helpful in developing and analyzing our queue management system. By using both behavioral and structural diagrams, we effectively mapped the interactions, processes, and architecture of the system. The use of UML facilitated clear communication between stakeholders, simplified the design of complex system components, and ensured a coherent understanding of system functionality. As evidenced by the specific diagrams and data dictionary tables provided, the UML modeling language not only enhanced the quality of the system design, but also streamlined the development process, ultimately contributing to a more efficient and reliable queue management solution.

Chapter 3

System Development

3.1 Introduction

This chapter explores the crucial components of system development, focusing on the deployment environment, development tools, programming languages, frameworks, and user interfaces. These elements lay the groundwork for creating a robust and user-friendly system.

3.2 Deployment environment

The system setup includes several key components:

- **Server Computer:** This computer hosts the database and the Laravel project. It is managed by the system administrator.
- **Entrance Computer:** Located at the entrance of the waiting area, this computer is connected to a ticket printer. It displays a page for customers to obtain their waiting tickets.
- **Integrated Display Screen:** Positioned in the middle of the waiting area, this screen, which has its own processor, shows information about the queue. It displays the number of the next customer and the corresponding desk number.
- **Committee Staff Computers:** One or two computers are available for the committee staff to manage queue operations, such as calling the next customer or recording service session details.
- **Administrator Computer:** This computer is used by the administrator to add or remove employees from the system.

3.3 Development environment

3.3.1 Visual Studio Code:

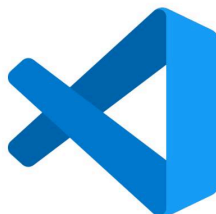


Figure III.1: Visual Studio Code logo

- Visual Studio Code is an open-source, cross-platform text editor utilized in web and software development.
- It offers a user-friendly interface tailored to developers' needs and supports multiple programming languages and extensions, enhancing productivity and simplifying code writing and debugging.
- Notable for its integration with popular development tools and version control systems like Git, it facilitates collaboration and streamlines source code management.
- Additionally, it provides an integrated runtime and debugging environment for web and software applications, aiding in swift error identification and resolution.

3.3.2 MySQL:



Figure III.2: MySQL logo

- MySQL stands out as a popular open-source database management system, revered for its ease of use and seamless integration with Structured Query Language (SQL).
- Renowned for its powerful tools, MySQL facilitates efficient data management and analysis.
- Widely employed for storing and retrieving various project-related data types, including user information, reservation details, comments, and more.
- MySQL equips users with comprehensive database management tools, enabling tasks such as table creation and modification, query execution, and storage, update, and retrieval operations tailored to application needs.

3.4 The programming languages

3.4.1 PHP Programming Language:

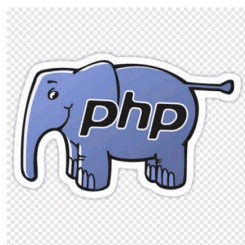


Figure III.3: PHP logo

Reasons for Choosing PHP for Project Implementation:

- PHP is among the most widely used and popular programming languages for web application development.
- The availability of PHP as a free and open-source language simplifies its adoption, extension, and enhancement by developers.
- PHP's seamless interaction with databases makes it well-suited for implementing and managing the project's associated database.
- PHP boasts support for numerous frameworks and libraries, facilitating the development of robust and scalable applications.

Capabilities and Features of PHP in Web Application Development:

- PHP provides advanced capabilities in text processing and validation, ideal for creating login systems, session management, and text processing.
- Strong support in PHP for connecting and interacting with other web services enables smooth integration between applications and external data sources.
- PHP supports modern web technologies like AJAX and JSON, empowering developers to build interactive and dynamic applications.

Utilizing PHP in Creating and Managing the Database and Interacting with Other Project Elements:

- With PHP, establishing a connection to the project's database and executing queries to retrieve, update, and delete data is achievable.
- PHP facilitates integration with other project elements such as JavaScript libraries or external APIs.
- The PHP language enables the execution of data processing operations and conversion to different formats such as JSON or XML.

3.4.2 CSS Programming Language and HTML Technology:

HTML (Hypertext Markup Language):



Figure III.4: HTML logo

- HTML serves as the standard markup language for creating the structure and content of web pages.
- It comprises a set of tags and elements defining various parts of a web page, including headings, paragraphs, images, links, tables, forms, and more.
- HTML forms the foundation for displaying content on the web and acts as the structural backbone for web documents.

CSS (Cascading Style Sheets):



Figure III.5: CSS logo

- CSS is a style sheet language utilized to describe the presentation and visual appearance of HTML and XML documents.
- It empowers web developers to manage the layout, design, and styling of web pages by associating style rules with HTML elements.
- CSS defines properties such as colors, fonts, sizes, margins, padding, positioning, and more, enabling precise control over the visual aspects of a website.
- By separating content and presentation, CSS facilitates easier maintenance and updates to the visual elements of a website.

Capabilities and Features of CSS and HTML in Project Implementation:

- HTML defines the structure of web pages, including elements like headings, paragraphs, lists, links, and images.
- CSS is responsible for styling elements on web pages, covering properties such as colors, fonts, margins, borders, and effects.
- CSS offers comprehensive formatting concepts like transitions, transformations, and responsive layouts, enhancing user experience across devices.

Usage of CSS and HTML in Project Implementation and Design:

- HTML establishes page structure and basic element formatting.
- CSS defines visual styles, contributing to desired design and smooth formatting.
- Advanced CSS features like flexible grids and media queries enable responsive user experiences.
- Utilizing CSS libraries such as Bootstrap and Foundation streamlines the design process and yields impressive visual outcomes.

3.4.3 java script Programming Language:



Figure III.6: JavaScript logo

JavaScript is a high-level, dynamic, untyped, and interpreted programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. JavaScript is an essential part of web applications; it enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it.

- **Client-Side Execution:**

- **Performance:** JavaScript is executed on the client-side, which reduces the demand on the server and the need for server-side processing. This leads to faster and more responsive interactions for the user.
- **Interactivity:** It allows for the creation of highly interactive websites. Users can engage with features such as animations, form validations, and dynamic content updates without needing to reload the page.

- **Versatility:**

- **Web Development:** JavaScript is used for both front-end and back-end development (with Node.js), making it a versatile language for full-stack development.
- **Libraries and Frameworks:** A plethora of libraries (like jQuery) and frameworks (like React, Angular, and Vue.js) extend JavaScript's capabilities, allowing developers to build complex applications more efficiently.
- **Ease of Learning and Use:**
 - **Syntax:** JavaScript has a relatively simple and flexible syntax, which is easy to learn for beginners and versatile enough for experienced developers.
 - **Community Support:** It has a large and active community, providing ample resources for learning and troubleshooting, including extensive documentation, forums, and tutorials.
- **Compatibility:**
 - **Browser Support:** JavaScript is natively supported by all modern web browsers, ensuring that scripts run consistently across different platforms and devices.
 - **Integration:** It integrates seamlessly with HTML and CSS, enabling the enhancement of web pages without the need for additional plugins or tools.
- **Asynchronous Processing:**
 - **AJAX:** JavaScript supports asynchronous operations through AJAX (Asynchronous JavaScript and XML), allowing web pages to fetch data from servers without refreshing the entire page. This results in smoother and faster user experiences.
 - **Promises and Async/Await:** Modern JavaScript provides constructs like Promises and `async/await` for handling asynchronous operations more effectively, making code easier to read and maintain.
- **Rich Interfaces:**
 - **User Experience:** JavaScript enables the creation of rich user interfaces, including drag-and-drop components, sliders, and other interactive elements, enhancing the overall user experience.
 - **Data Visualization:** Libraries such as D3.js and Chart.js allow for the creation of sophisticated data visualizations, making it easier to present complex data in an understandable and visually appealing manner.
- **Development Tools:**
 - **Debugging:** Modern browsers come with powerful JavaScript debugging tools that help developers identify and fix issues efficiently.
 - **Development Environments:** Numerous development environments and tools, such as Visual Studio Code, WebStorm, and others, offer features specifically designed for JavaScript development, including code completion, linting, and version control.

3.5 the frameworks:

3.5.1 Laravel:



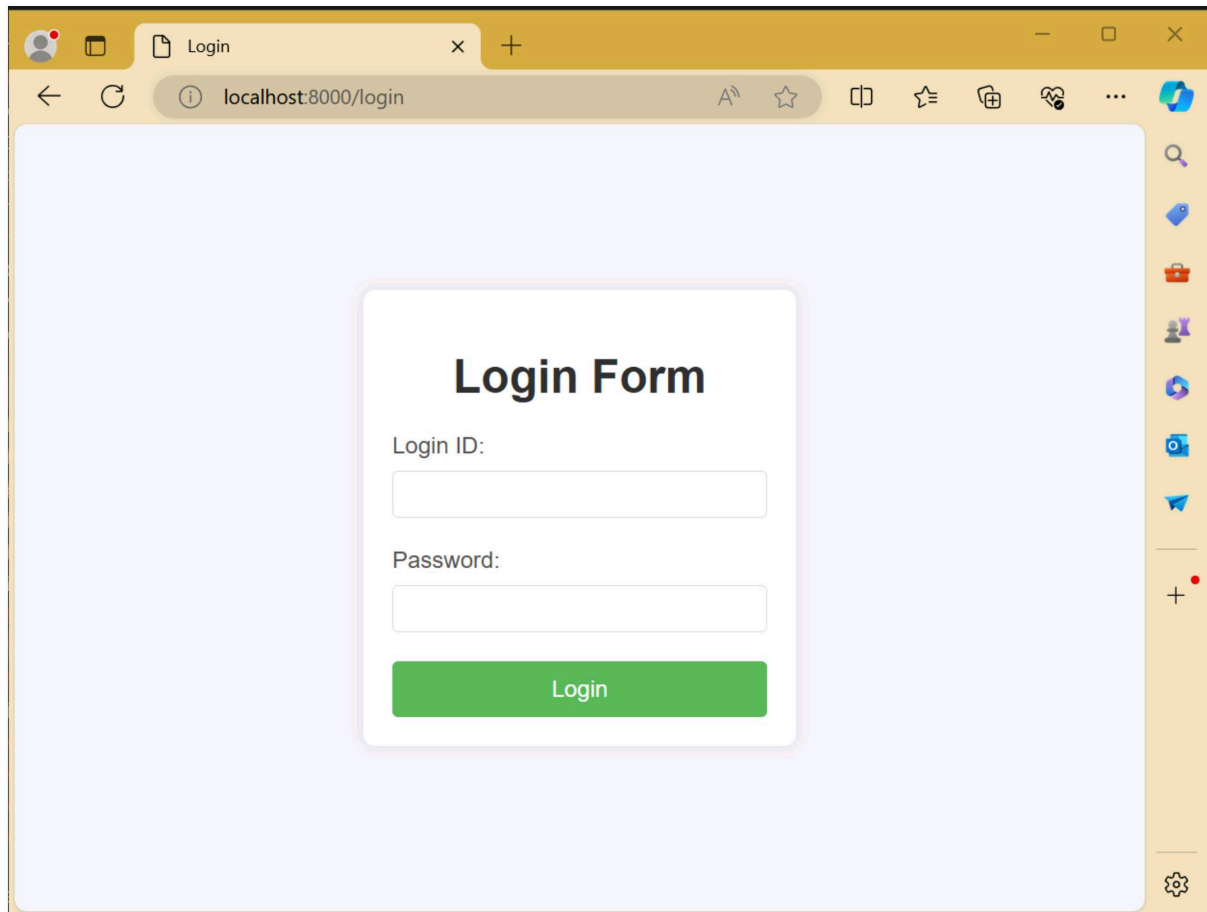
Figure III.7: Laravel logo

Laravel is an open-source web framework based on PHP, renowned for its elegant syntax and developer-friendly features. It follows the Model-View-Controller (MVC) architectural pattern and offers a comprehensive suite of tools and libraries for constructing web applications. Laravel simplifies common tasks such as routing, caching, database management, and authentication, enabling developers to concentrate on crafting clean and maintainable code. Additionally, it includes Blade, a templating engine for generating dynamic views.

3.6 Some of the interfaces:

3.6.1 Login page:

In this page, the committee worker or administrator puts his/her ID and password and clicks on Login to get his/her role page if the ID and password are correct, otherwise he/she will not be logged in and will get an error message.



The screenshot shows a web browser window with a single tab titled 'Login'. The address bar displays 'localhost:8000/login'. The page content is a simple login interface centered on a light blue background. The form is white with a thin border and contains the following elements:

- Login Form**: A heading in bold black text.
- Login ID:**: A label followed by a text input field.
- Password:**: A label followed by a text input field.
- Login**: A green rectangular button with white text.

The browser's sidebar on the right shows various application icons, and the bottom right corner of the page has a small gear icon for settings.

Figure III.8: Login page

3.6.2 Committee worker page:

This is the committee worker's page where he/she performs the functions of calling the next customer and performing the service session for the current customer.

"No customer data available" is written when the committee worker has already entered this page or if the next customer button is pressed when the queue is empty.

The committee worker can also log out of this page and return to the login page.

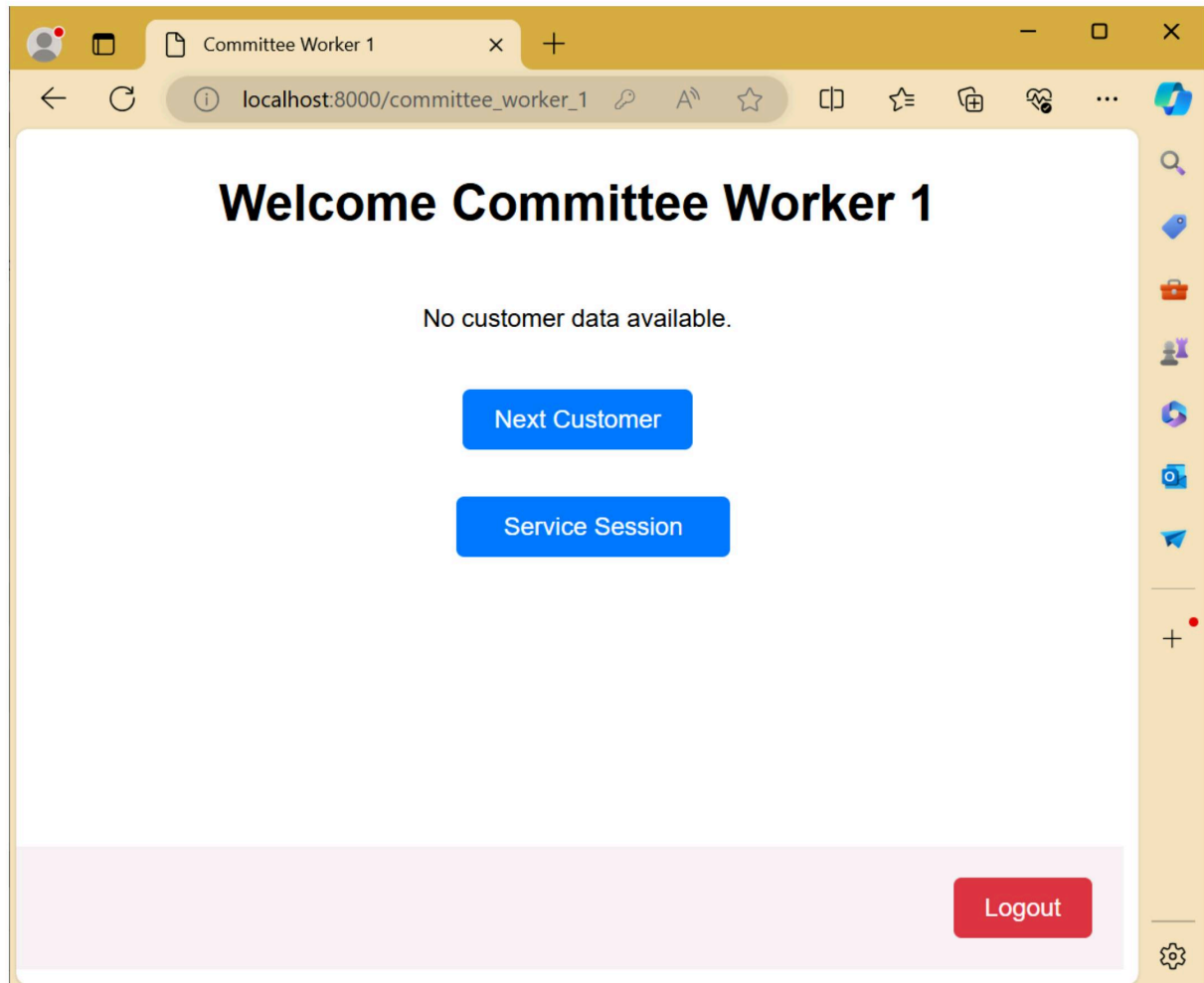


Figure III.9: Committee worker page(No customer available.)

When customers are available in the queue and the next customer button is pressed, the customer's information (ticket number and customer name) is displayed on the page of the committee worker who called them in the current customer field.

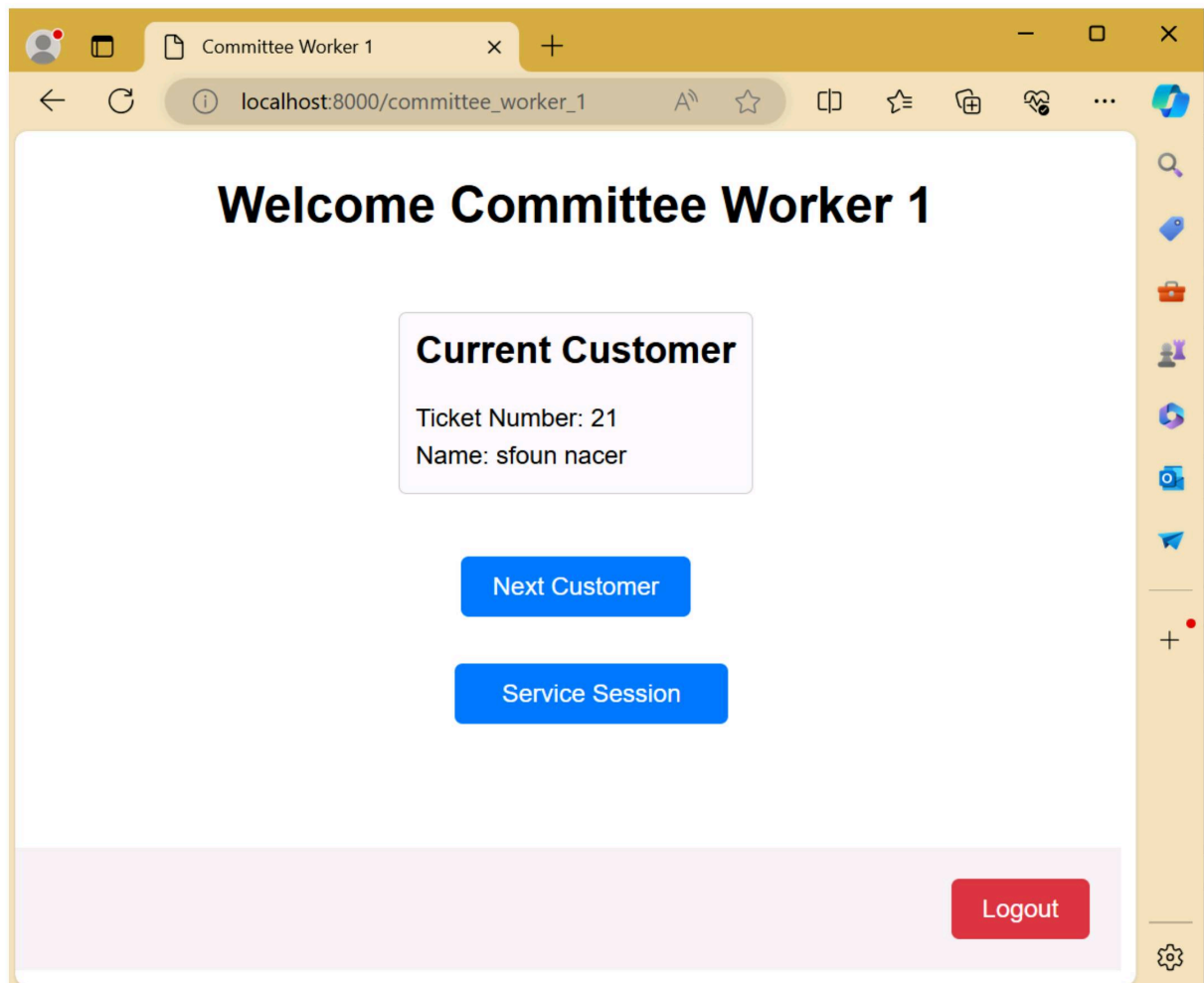


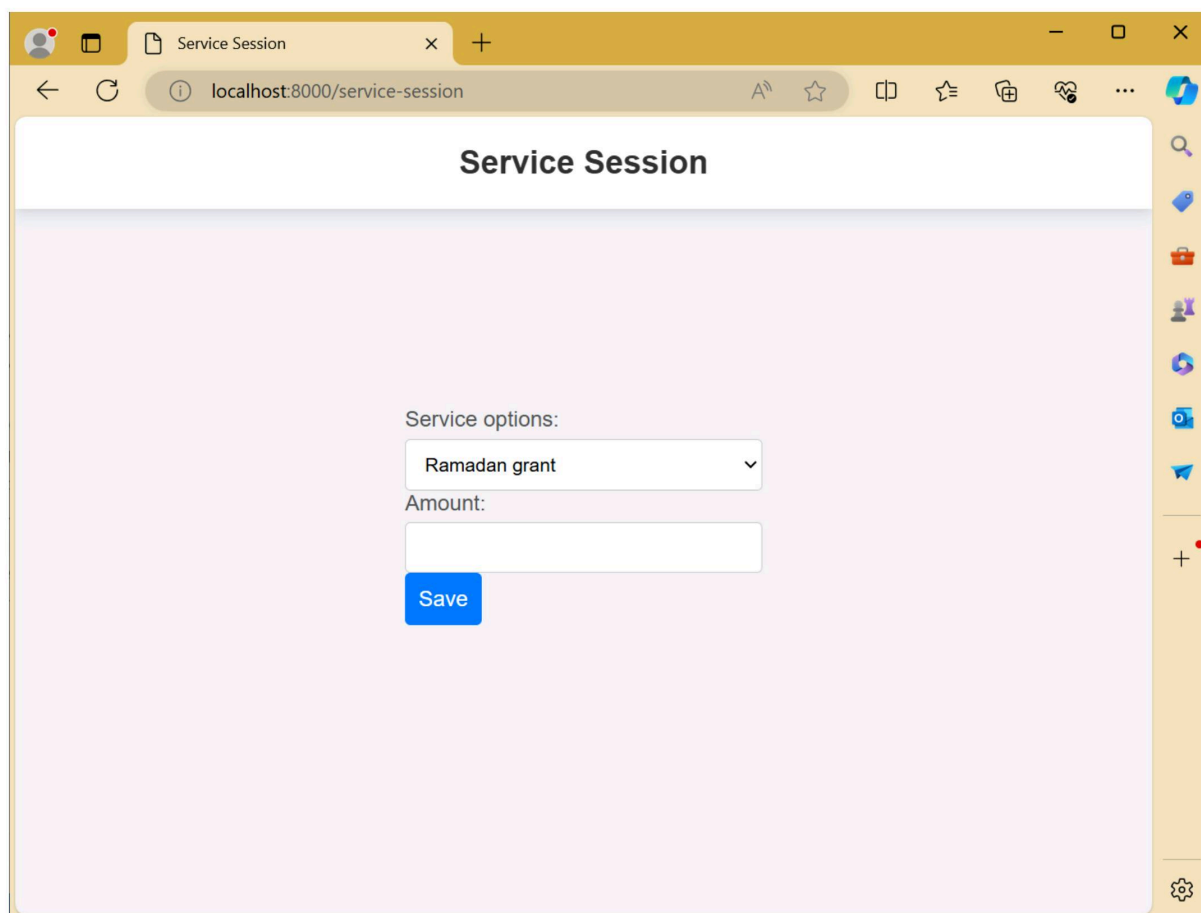
Figure III.10: Committee worker page(Committee Worker 1)

3.6.3 Service Session page:

On this page, the committee worker saves the service session for the current customer, specifying the type of service he provided to the customer and the amount he gave to the customer.

The committee worker can type other information in the amount field if they wish.

After clicking Save, the service session information, along with the committee worker's ID and the customer's (employee's) ID, is entered into the service session table in the database.



The screenshot shows a web browser window with the title 'Service Session'. The address bar displays 'localhost:8000/service-session'. The page content is titled 'Service Session' and features a form with the following elements:

- Service options:** A dropdown menu with 'Ramadan grant' selected.
- Amount:** A text input field.
- Save:** A blue button.

The browser's right sidebar contains various icons, including a search icon, a home icon, a shopping cart icon, a user profile icon, a camera icon, a mail icon, a plus icon, and a settings icon.

Figure III.11: Service Session page

3.6.4 Display queue information page:

In this page, queue information is displayed, such as the number of the last ticket printed, the number of customers waiting, and the number of the customer being served now.

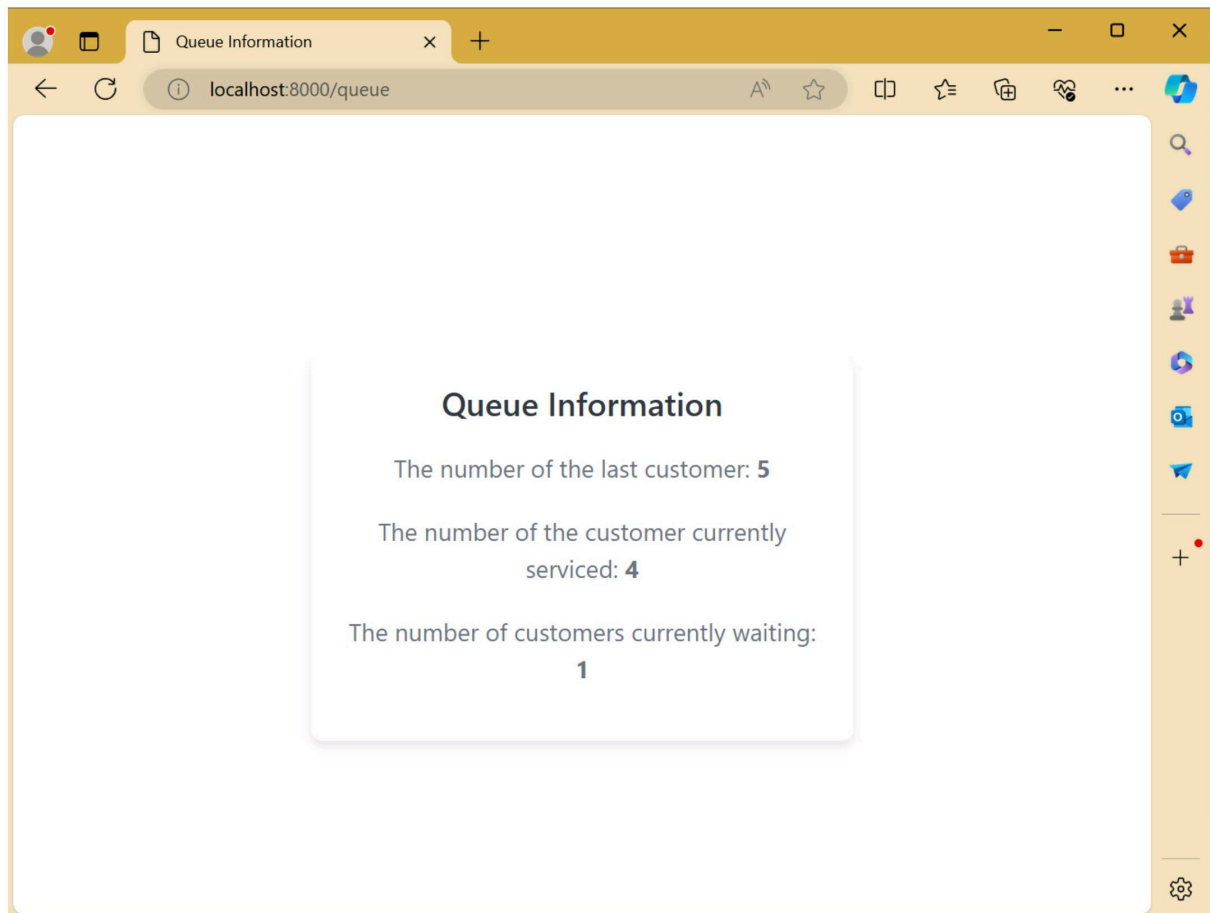


Figure III.12: Display queue information page

3.6.5 Display next ticket:

When the committee worker presses the Next Customer button and the queue is not empty, a model is displayed on the queue information page containing the next ticket number and the number of the office he or she is requesting.

This model disappears ten seconds after it appears and the page returns to displaying the queue information.

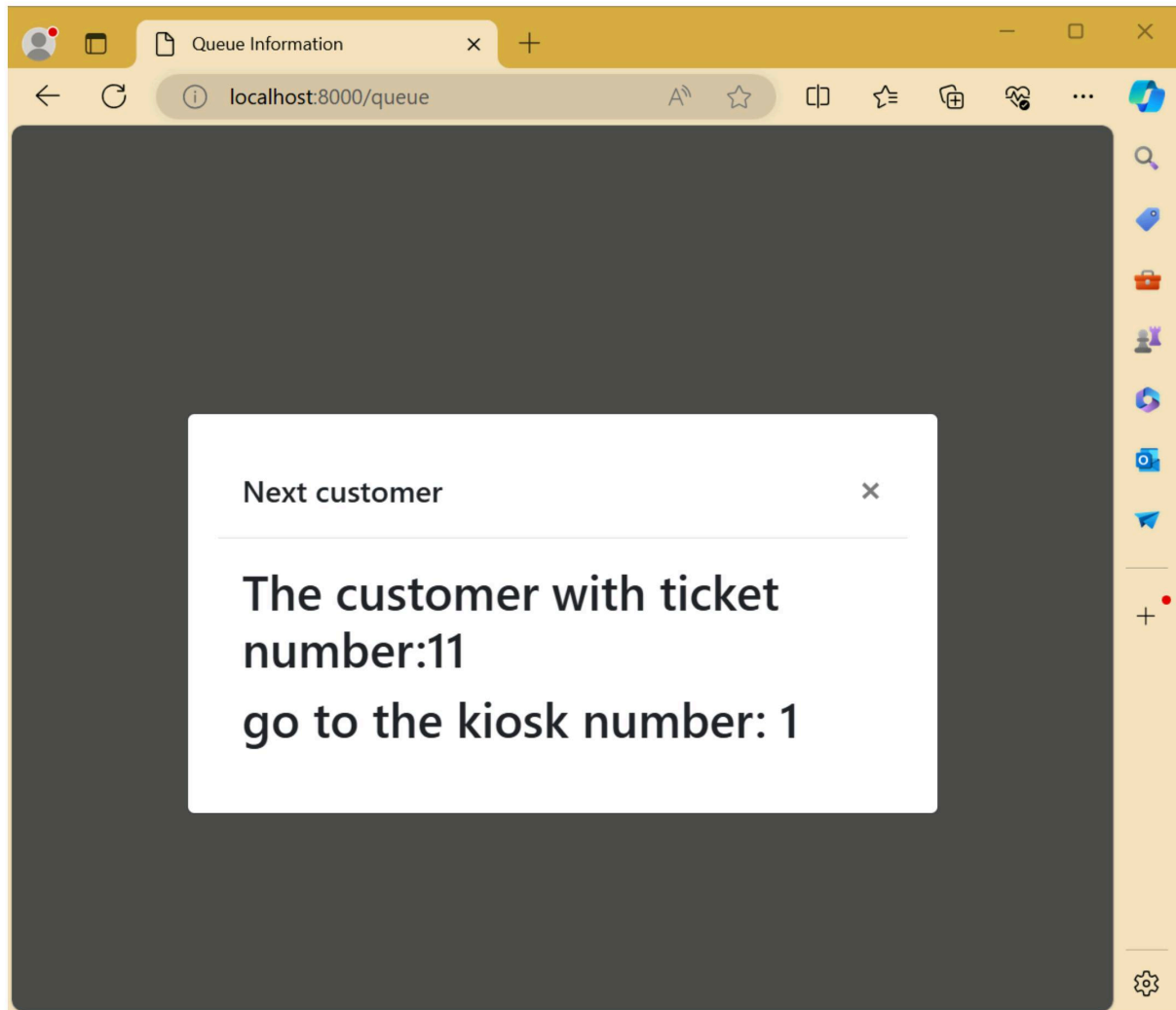


Figure III.13: Display next ticket in queue information page as model

3.6.6 Get your ticket page:

On this page, the employee enters their ID to receive their ticket and become a customer with a number in the queue.

If the ID is wrong or is already in line, the customer will not get their ticket and will be shown the reason message.

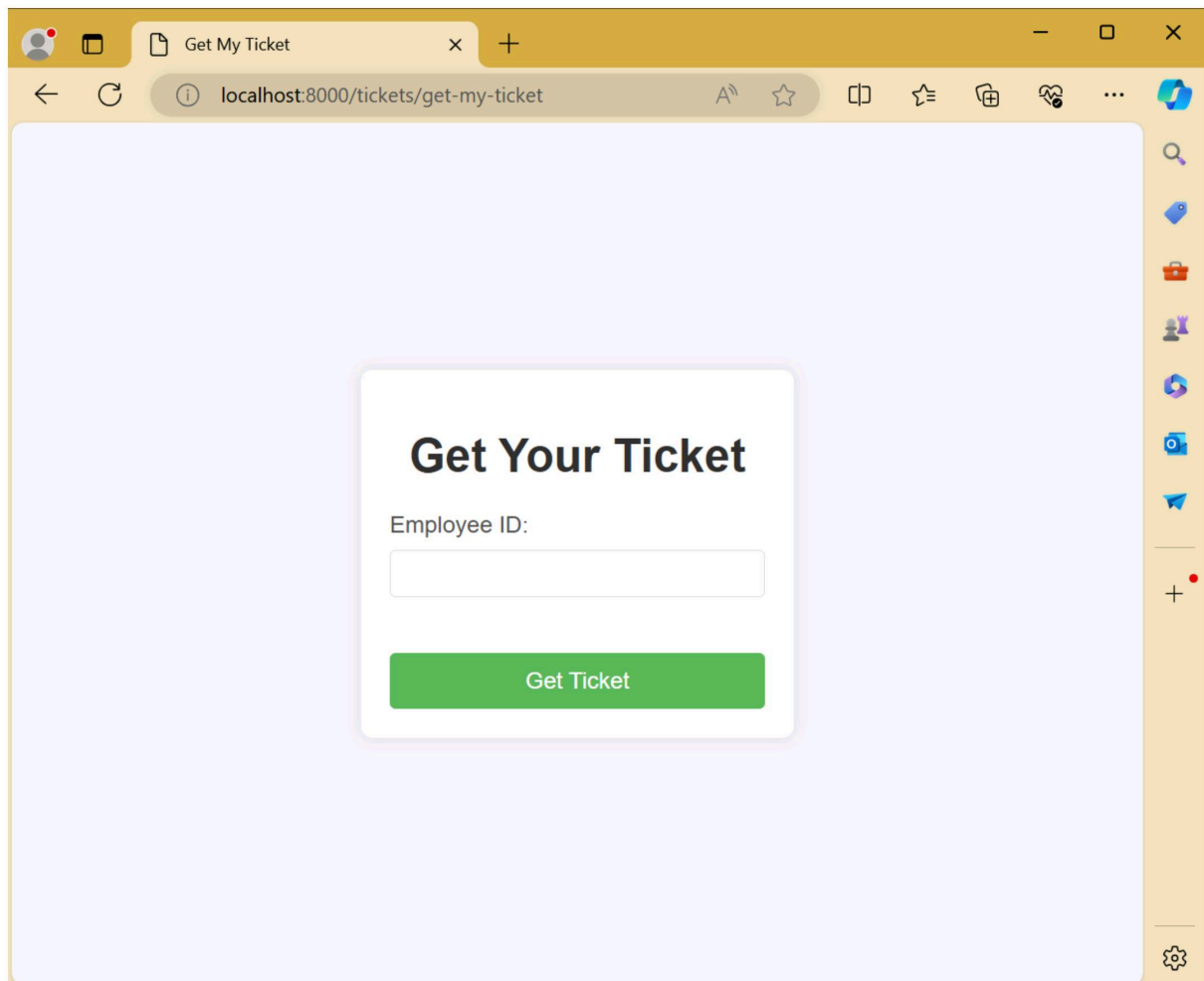


Figure III.14: Get your ticket page

The ticket can be printed or taken as a pdf.

The ticket format:

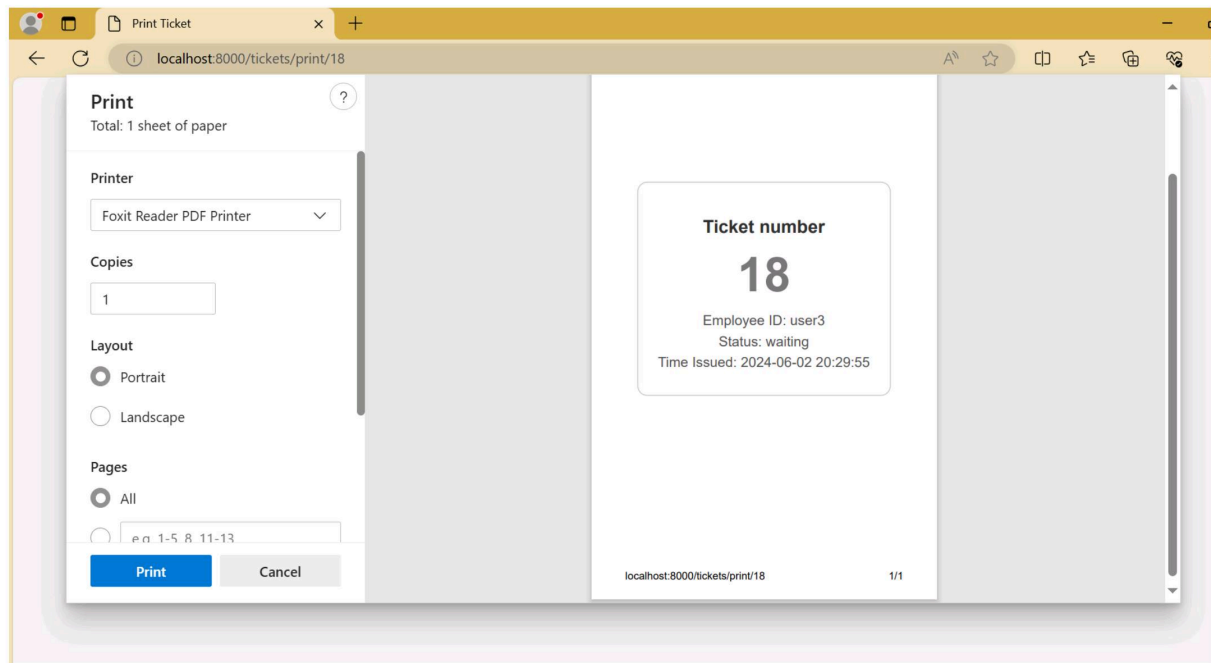


Figure III.15: Ticket to print (ticket 18)

3.6.7 Administrator page:

The administrator page can add or delete employees.

The use of this page is very limited because employees are usually imported from an Excel file directly into the database's employee table.

The administrator can also log out of this page and return to the login page.

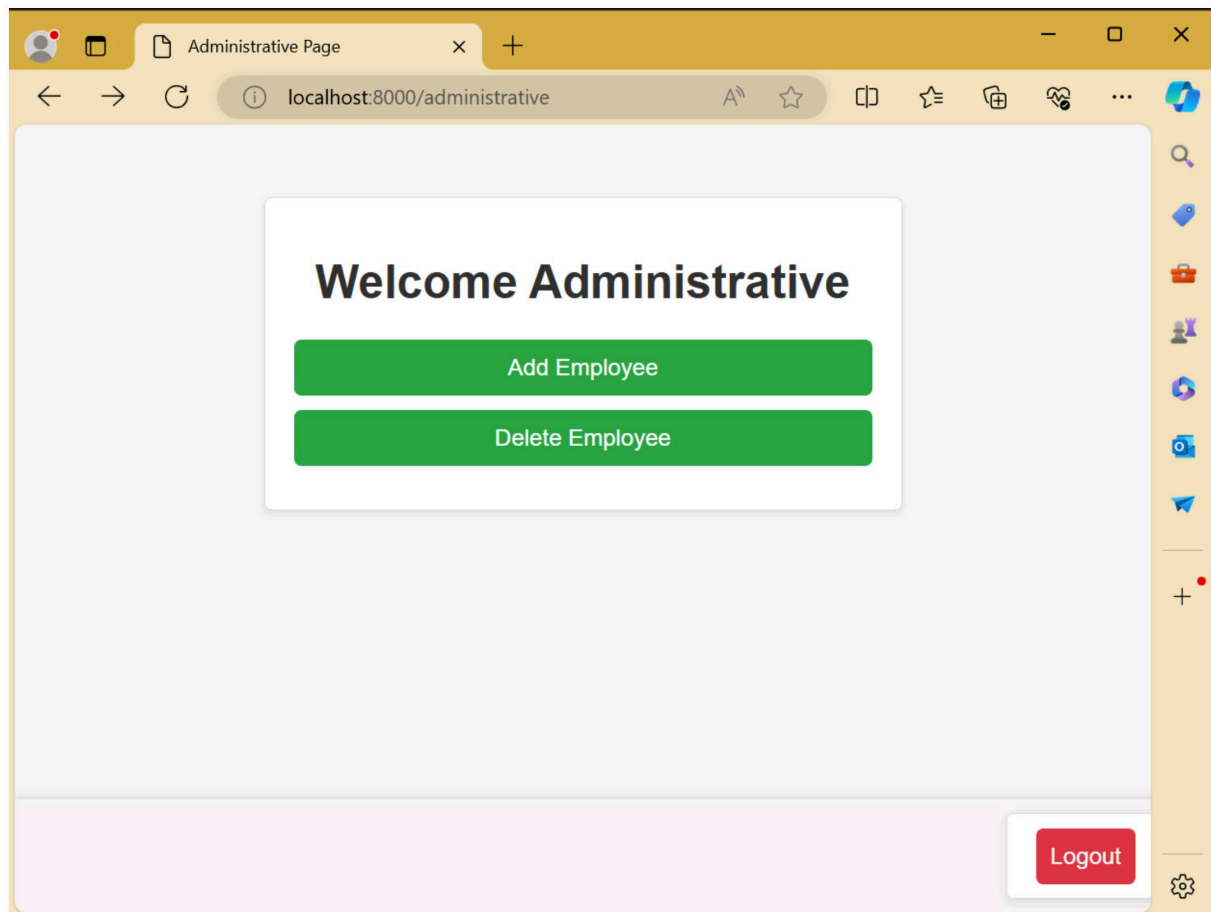


Figure III.16: Administrator page

3.6.8 Add employee page:

On this page, a new employee is added. The fields are Employee ID, Employee first name and last name, date of birth, date of Inauguration, and the department to which the employee belongs with the appropriate information.

When you click the Add Employee button, the information is entered into the database's employees table and a success message is displayed if the employee entry is successful or a failure message if the employee entry fails.

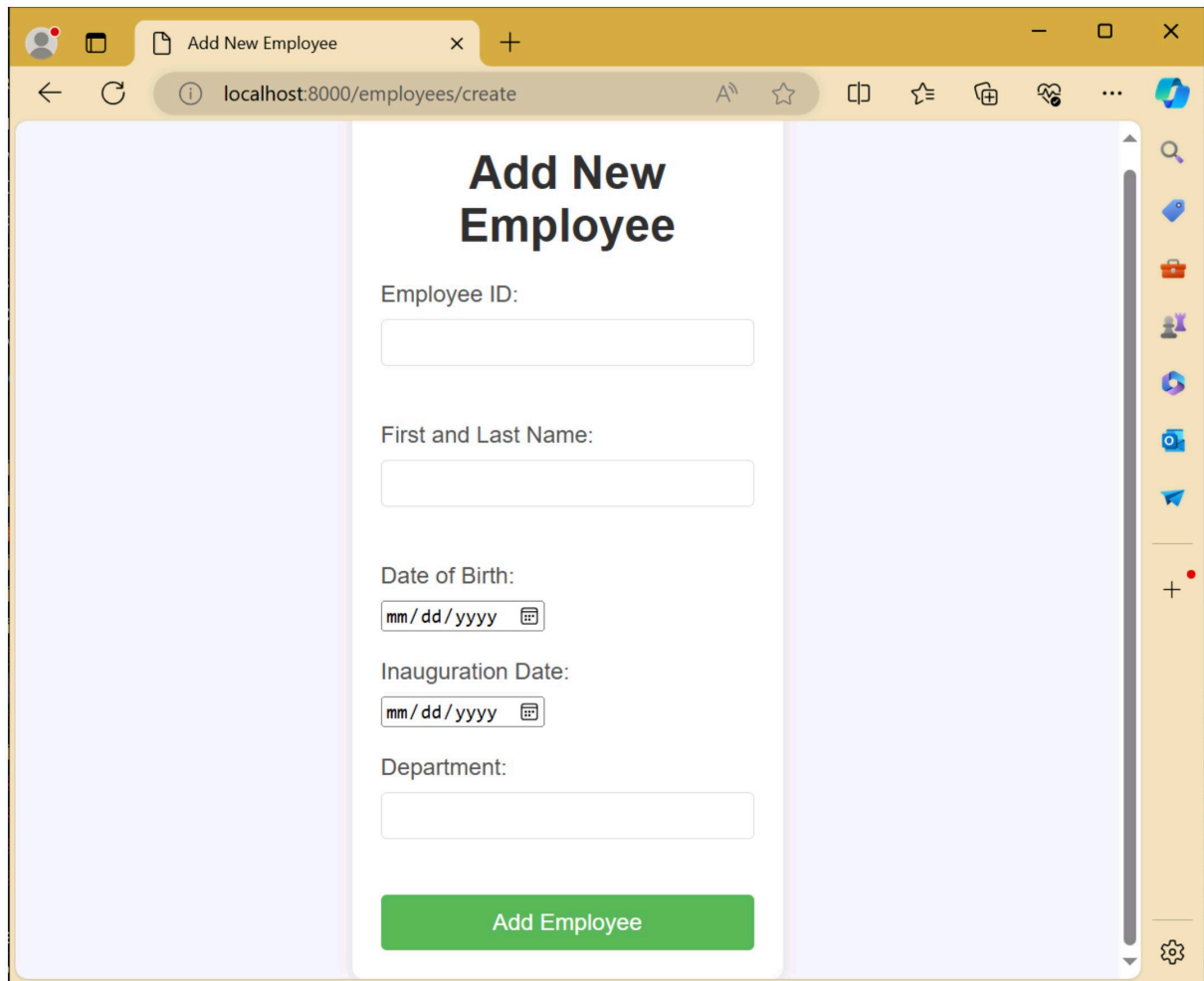
A screenshot of a web browser displaying a form titled "Add New Employee". The browser's address bar shows "localhost:8000/employees/create". The form is centered on a light purple background. It contains five input fields: "Employee ID:", "First and Last Name:", "Date of Birth:" (with a date picker icon), "Inauguration Date:" (with a date picker icon), and "Department:". At the bottom of the form is a green button labeled "Add Employee". The browser's sidebar on the right shows various application icons.

Figure III.17: Add employee page

3.6.9 Delete employee page:

On this page, the administrator puts the ID of the employee he wants to delete, and when he clicks on the Delete button, the employee is deleted from the employees' table in the database and gets a success message or a failure message if the employee is not deleted

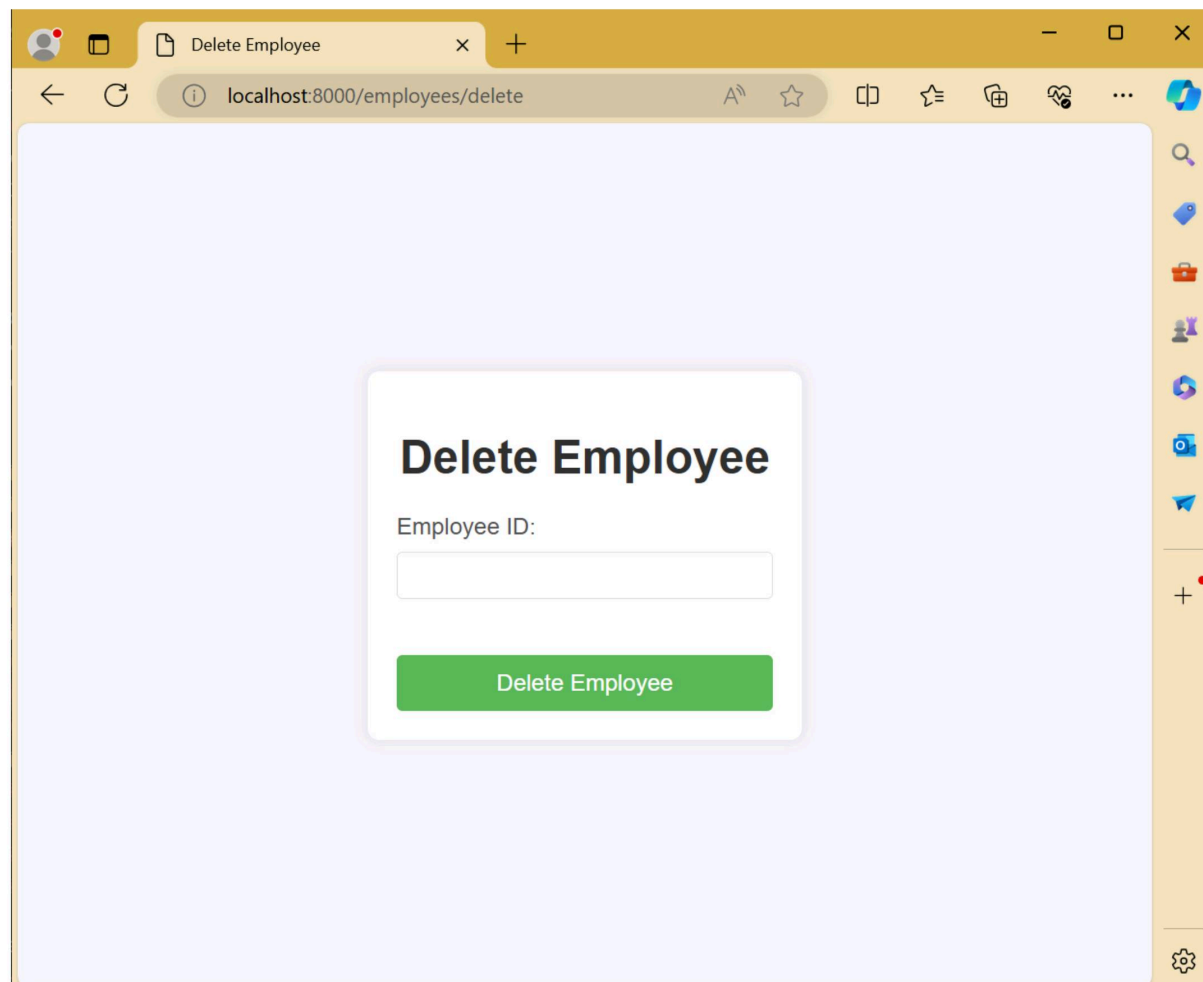


Figure III.18: Delete employee page

3.7 Conclusion

In summary, the deployment environment, development tools, programming languages, frameworks, and user interfaces collectively shape the system's functionality and user experience. By carefully selecting and utilizing these components, developers can create efficient, scalable, and intuitive solutions that meet the needs of users and administrators.

General Conclusion

In summary, this memorandum details the development and implementation of an automated Queue Manager System for the Social Services Committee at the University of El-Oued. The primary problem addressed is the significant organizational challenges faced during key events such as the distribution of Ramadan food baskets and school bags, which previously resulted in long wait times and beneficiary dissatisfaction.

The system developed aims to streamline queue management, reducing waiting times and improving user satisfaction by efficiently managing the flow of beneficiaries. Key results include the creation of a user-friendly web system, robust backend infrastructure, and seamless integration of various components. The system employs modern programming languages and frameworks, such as PHP, JavaScript, HTML, CSS, and Laravel, ensuring scalability and maintainability.

The introduction of the Queue Manager System marks a significant improvement in operational efficiency for the Social Services Committee. Beneficiaries can now receive their services in a timely and organized manner, reflecting positively on the committee's commitment to social support.

Looking forward, the project can be enhanced by integrating additional features. Future work could include the installation of a scanner to allow ticket retrieval through card swipes, transitioning to electronic tickets sent directly to beneficiaries' phones or via WhatsApp, and implementing status updates through this electronic tickets to inform beneficiaries of their exact turn. These enhancements will further reduce paper waste and enhance the overall user experience, aligning with ethical, economic, and environmental considerations.

Overall, this project sets a new standard for queue management in social service environments, offering a scalable and efficient solution that can be adapted and improved upon in future iterations.

Bibliography

- [1] HABBACHE Fares and MAIZA Messaoud Amir. Queue management system (qms) as a recent trend in improving service's quality. *Economic Researcher Review*, 9(1):71, 2021.
- [2] HABBACHE Fares and MAIZA Messaoud Amir. Queue management system (qms) as a recent trend in improving service's quality. *Economic Researcher Review*, 9(1):71, 2021.
- [3] HABBACHE Fares and MAIZA Messaoud Amir. Queue management system (qms) as a recent trend in improving service's quality. *Economic Researcher Review*, 9(1):72, 2021.
- [4] HABBACHE Fares and MAIZA Messaoud Amir. Queue management system (qms) as a recent trend in improving service's quality. *Economic Researcher Review*, 9(1):72, 2021.
- [5] OpenAI. Chatgpt: A large-scale generative pre-trained transformer. 2019.
- [6] OpenAI. Chatgpt: A large-scale generative pre-trained transformer. 2019.
- [7] OpenAI. Gpt-3: Language models are few-shot learners. 2020.
- [8] OpenAI. The chatgpt family: Advancements in conversational ai. 2021.
- [9] OpenAI. Gpt-4: Pushing the boundaries of natural language understanding. 2022.
- [10] WaveTec. Best queue management system review. <https://www.wavetec.com/blog/queue-management/best-queue-management-system-review/>.
- [11] Batna university website. https://doubatnac.univ-batna.dz/index.php?option=com_content&view=article&id=25&Itemid=173&lang=ar, 2023.
- [12] The internal regulations of the social services committee of ibn khaldoun university, tiaret for the year 2020, 2020.