Akdeniz University
Department of Engineering

Software Engineering Project

Project short name: Hotel Management System

Software Requirements Specification

İbrahim Utku Adanur, Seher Oğuz, Ozan Ergüleç

Team Leader: Ozan Ergüleç
Product Owner: İbrahim Utku Adanur

Instructor: Prof. Ümit Deniz ULUŞAR

16.03.2025

This report is submitted to the Department of Computer Engineering of Akdeniz University of the Software Engineering course CSE332.

Abbreviations	
IP	Internet Protocol
GUI	Graphical User Interface
HTTPS	Secure Hypertext Transfer Protocol
KVKK	Kişisel Verilerin Korunması Kanunu (Turkish Personal Data Protection Law)
SRS	Software Requirements Specification
UI	User Interface
RN	React Native

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to provide a detailed description of the requirements for the Hotel Management System. This Software Requirements Specification (SRS) document outlines the scope, functionality, and constraints of the system, which is designed to streamline and centralize various hotel operations, including reservation management, customer relations, financial management, room organization, and staff operations.
This SRS covers the entire system, which is intended to improve the efficiency of hotel management, enhance customer satisfaction, and provide better financial oversight. The document is intended to serve as a reference for the development team to build the first version of the system and to be proposed to stakeholders for approval. The system will be developed as a comprehensive solution to manage all aspects of hotel operations, ensuring smooth and efficient management of resources, staff, and customer interactions.

1.2 Product Scope

The Hotel Management System is a comprehensive software solution designed to streamline and centralize various hotel operations, including reservation management, customer relations, financial management, room organization, and staff operations. The purpose of this system is to improve the efficiency of hotel management, enhance customer satisfaction, and provide better financial oversight.
The system will offer a range of features to support hotel staff in managing daily operations, including:

Reservation Management: Enabling online and group bookings, manual entries, and real-time room availability tracking.
Customer Management: Tracking guest profiles and maintaining detailed customer histories.
Hotel Units Management: Categorizing rooms, tracking amenities, managing bed types, and floor plans.
Financial Management: Supporting income-expense tracking, financial reporting, bank account management, and flexible pricing models.
Personnel Management: Managing employee records, shift planning, attendance tracking, and performance evaluations.
Customer Relationship Management: Maintaining customer profiles, contact details, and interaction histories.

The Hotel Management System will be accessible via a web and mobil based interface, allowing hotel staff to manage operations from any device with an internet connection.
The primary objectives of the system are to:
Improve Operational Efficiency: By automating routine tasks and providing real-time data, the system will reduce manual effort and minimize errors.
Enhance Customer Satisfaction: By providing a seamless booking experience and personalized service, the system will help hotels build stronger relationships with their guests.
Provide Financial Oversight: By tracking income and expenses in real-time, the system will help hotel managers make informed financial decisions.
Support Scalability: The system will be designed to accommodate the needs of hotels of various sizes, from small boutique hotels to large chains.
The Hotel Management System aligns with the broader business strategy of improving customer satisfaction and operational efficiency, which are critical for the success of any hotel. By providing a centralized platform for managing all aspects of hotel operations, the system will help hotels achieve their business goals and maintain a competitive edge in the hospitality industry.

1.3 References

1. Software Requirement Report Template for HW1.docx – Akdeniz University, Department of Computer Engineering.
2. Software Engineering Project Reportpdf – Group Project Instructions

2 Overall Description

2.1 Product Perspective

The Hotel Management System is a new, self-contained software product designed to streamline and centralize various hotel operations, including reservation management, customer relations, financial management, room organization, and staff operations. The system is developed to address the growing need for efficient and automated hotel management solutions in the hospitality industry. It is not a replacement for any existing system but rather a modern, integrated solution that can be adopted by hotels of various sizes, from small boutique hotels to large chains.
The system will consist of two main components, both designed exclusively for hotel staff (customers will not have access to these interfaces):
Web-Based Management Portal: This will be the primary interface for hotel staff to manage all aspects of hotel operations, including reservations, customer profiles, room management, financial tracking, and employee management. The web portal will be accessible from any device with an internet connection, providing flexibility and ease of use for hotel administrators.
Mobile Application for Staff: This mobile application will provide the same functionality as the web-based portal but will be optimized for on-the-go access. It will allow hotel staff to manage operations from their mobile devices, ensuring that they can perform tasks such as checking room availability, managing reservations, and updating customer information even when they are not at a desk.
Both the web-based portal and the mobile application will serve the same purpose and provide the same features, ensuring consistency and flexibility for hotel staff. The mobile application is designed to complement the web portal, offering a more convenient way for staff to access the system while maintaining the same level of functionality.

System Context and Interfaces:
Database: The system will rely on a centralized database to store all relevant data, including customer information, reservation details, room inventory, financial records, and employee data. Both the web-based management portal and the mobile application will interact with this database:
Both interfaces will have full access to the database for adding, modifying, and retrieving data.
The database will ensure real-time synchronization between the web portal and the mobile application, allowing staff to access up-to-date information regardless of the interface they are using.
Resource Allocation: The mobile application is designed to be resource-efficient, ensuring smooth operation even on devices with limited hardware capabilities. It will be optimized for performance, with a focus on minimizing memory and storage usage while maintaining high responsiveness.

2.2 Product Functions

The Hotel Management System is designed to provide a comprehensive set of features to streamline hotel operations. Below is a high-level summary of the major functions that the system must perform or enable the user (hotel staff) to perform:
1. Reservation Management
Online and Group Bookings: Allow staff to manage reservations made through online platforms or group bookings.
Manual Reservation Entries: Enable staff to manually enter reservations for walk-in customers or phone bookings.
Real-Time Room Availability Tracking: Provide real-time updates on room availability to avoid overbooking.
Reservation Calendar: Display a calendar view of all reservations for efficient schedule management.
2. Customer Management
Guest Profile Tracking: Maintain detailed profiles of guests, including personal information, preferences, and contact details.
Customer History: Keep records of previous stays, including room preferences.
Contact Information Management: Store and manage guests' phone numbers, email addresses, and other contact details.
3. Hotel Units Management
Room Categorization: Categorize rooms based on type (e.g., single, double, suite) and features (e.g., sea view, balcony).

Amenities Tracking: Track amenities available in each room (e.g., mini-bar, Wi-Fi, air conditioning).
Bed Types and Floor Plans: Manage bed configurations (e.g., king-size, twin) and floor plans for efficient room allocation.
4. Financial Management
Income and Expense Tracking: Monitor daily income and expenses, including room revenue, additional services, and operational costs.
Financial Reporting: Generate financial reports for management, including profit and loss statements, occupancy rates, and revenue analysis.
Bank Account Management: Manage and track transactions related to the hotel's bank accounts.
Flexible Pricing Models: Implement dynamic pricing strategies.
5. Personnel Management
Employee Records: Maintain detailed records of all employees, including personal information, job roles, and employment history.
Shift Planning: Create and manage employee shifts to ensure adequate staffing at all times.
Attendance Tracking: Track employee attendance and working hours.
6. Mobile Application for Staff
On-the-Go Access: Provide hotel staff with access to the system via a mobile application, allowing them to manage reservations, check room availability, and update customer information from anywhere.
Real-Time Synchronization: Ensure that data entered via the mobile app is instantly synchronized with the web-based portal and central database.
Task Management: Allow staff to view and manage their tasks and responsibilities directly from the mobile app.
7. System Administration
User Management: Manage user accounts and permissions for hotel staff.
System Updates: Notify staff of system updates and new features.

2.3 User Types and Characteristics

1. Hotel Manager (Müdür)
Role: Oversees all operations.
Access: Full access to hotel system.
Responsibilities: Monitors hotel performance, generates reports, and manages configurations.
2. Receptionist (Resepsiyonist)
Role: Manages guest interactions.
Access: Limited to check-ins, check-outs, and room reservations.
Responsibilities: Handles guest check-ins/check-outs, updates room availability, and manages guest profiles.
3. Accounting Manager (Muhasebe Müdürü)
Role: Manages finances.
Access: Financial data only (income, expenses, bank transactions).
Responsibilities: Tracks revenue, generates financial reports, and manages accounts.

2.4 Operating Environment

Mobile: Android 8.0+ via React Native + Expo
Web: Modern browsers (Chrome, Firefox,Safari, Edge) via React
Backend: .NET (version 6 or higher) hosted on a cloud or on-premise server
Database: PostgreSQL

2.5 Design and Implementation Constraints

- Must comply with KVKK (Turkish Data Protection Law) and general privacy regulations.
- Must ensure security (HTTPS)
- Since the application fetches data from the database over the Internet, it is crucial that there is an Internet connection for the application to function.

2.6 User Documentation

FAQ/Help Section accessible from within the application.

2.7 Assumptions and Dependencies

•Stable Internet Connection:
The system assumes that the hotel will have a reliable and high-speed internet connection for real-time synchronization and access to cloud-based services.
•Hardware Compatibility:
The system assumes that the hotel will have compatible hardware (e.g., computers, tablets, mobile devices) to run the software efficiently.

3 External Interface Requirements

3.1 User Interfaces

- Mobile UI (React Native):
  - Primary screens: Home ,Check-in, Check-out,Accounting,Staff Management, Financial Reports, Room Status, Customer Info
  - Login and Registration screens, where users can sign up or log in to their accounts.
  - Navigation bars, icons, and menu options appropriate for mobile UX.
  -
- Web UI (React):
  - Similar functionality to the mobile app, adapted for desktop or laptop screens
  - Responsive design ensuring ease of use on different browser window sizes.
  -

Final Interface: https://www.figma.com/design/vIpcNTqMyrOCqhFNTqN6a5/Untitled?node-id=0-1&t=yGtSgcAhdae8TTYk-1

3.2 Hardware Interfaces

End-User Devices:
Mobile Devices:
Android 8.0+ or iOS 12+ for the React Native mobile app.
Used by staff for on-the-go management (e.g., room status updates, check-ins).
Desktop/Laptop Computers:
For the React Native Web version, accessible via modern web browsers (Chrome, Firefox, Safari, Edge).
Used by managers and receptionists for detailed operations (e.g., reservations, financial tracking).

Server Hardware:
Hosting Environment:
Cloud or on-premise servers capable of running the .NET backend and PostgreSQL database.

Additional Notes:
No specialized hardware (e.g., custom POS terminals or sensors) is required.
The system relies on standard hardware and internet connectivity for seamless operation.

3.3 Software Interfaces

1. Databases:
PostgreSQL (Version 12+):
Purpose: Stores all system data, including reservations, guest profiles, financial records, and employee information.
Data Flow: Incoming data (e.g., new reservations, guest updates) and outgoing data (e.g., reports, room availability).
2. Backend Framework:
.NET Core (Version 6.0+):
Purpose: Handles business logic, API endpoints, and data processing.
Communication: RESTful APIs for communication between the frontend (web/mobile) and the database.
3. Frontend Frameworks:
React Native (for Mobile App):
Purpose: Provides the user interface for mobile devices.
Communication: Fetches data from the backend via RESTful APIs.

React (for Web Portal):
Purpose: Provides the user interface for desktop/laptop users.
Communication: Fetches data from the backend via RESTful APIs.
4. Libraries and Tools:
Entity Framework Core:
Purpose: ORM for database interactions.

3.4 Communications Interfaces

1. HTTPS (SSL/TLS): Purpose: Encrypts all communication between clients (web/mobile) and the server to protect sensitive data.
Usage: Protects user credentials and guest personal data during transmission. Ensures secure communication for all API requests and responses.
2. WebSocket / SignalR: Purpose: Enables real-time communication for instant updates.
Usage: Staff receive instant updates for new reservations, check-ins, or maintenance requests

4 System Features / Requirements

This section details the functional requirements of the system, organized by key features.

4.1 Receptionist

4.1.1 Check-in

4.1.1.1. Description and Priority

The Check-in and Check-out management system enables receptionists to efficiently handle guest arrivals and departures. This feature ensures seamless guest registration, room assignment, and billing processes while maintaining accurate records of hotel occupancy. Priority is **high,** check-in and check-out are fundamental hotel processes that occur daily. Without an efficient system, hotel operations would be severely disrupted.

4.1.1.2. Stimulus/Response Sequences

Receptionist selects "Check-in" option in the system. System displays a list of upcoming reservations and available rooms. Receptionist searches for an existing reservation or creates a new booking. System retrieves reservation details or prompts for guest information entry. Receptionist verifies guest identification and required documents. System confirms identity and validates document details. Receptionist assigns a room based on availability and guest preferences. Receptionist confirms check-in and finalizes the process.

4.1.1.3. Functional Requirements

ID	Title	Description	Rationale	Dependencies
FR1	Search for Reservation	The system should allow receptionists to search for an existing reservation	To enable quick retrieval of reservation details	None
FR2	Create a New Booking	If no reservation exists, the receptionist should be able to manually create a new booking	To allow walk-in guests or last-minute bookings	None
FR3	Assign Room	The system should allow the receptionist to assign an available room to the guest based on preferences and availability	To manage room allocation efficiently	FR1, FR2
FR4	Display List of Checked-in Guests	The system should provide a list of all guests currently checked in	To facilitate guest management	None

			and quick lookups	

4.1.2. Check out

4.1.2.1. Description and Priority

The Check-out Management system allows receptionists to efficiently handle guest departures by finalizing payments, updating room availability, and maintaining accurate billing records. This feature ensures a smooth guest departure process and real-time hotel occupancy updates. Priority is **high,** Check-out is a **critical hotel operation** that occurs daily.

4.1.2.2. Stimulus/Response sequences

Receptionist selects "Check-out" option in the system. Receptionist searches for and selects the guest who is checking out. Receptionist confirms check-out and finalizes the process.

4.1.2.3 Functional requirements

ID	Title	Description	Rationale	Dependencies
FR5	Retrieve Checked-in Guest List	The system should provide a list of all currently checked-in guests	To enable quick selection of the guest for check-out.	None
FR6	Search for Guest	The system should allow receptionists to search for a guest by name, room number, or reservation ID	To facilitate quick retrieval of guest records.	FR5
FR7	Update Room Status	After check-out, the system should mark the room as "Available" for future reservations.	To ensure real-time room availability updates.	None

4.2 Accountant

4.2.1. Accounting Operations

4.2.1.1. Description and Priority

The Accounting Management System allows accountants to efficiently handle hotel financial transactions, including payment processing, expense tracking, and financial reporting. This feature ensures accurate financial records, seamless revenue management, and compliance with financial regulations. Priority is high, accounting operations are essential for maintaining the hotel's financial health. Accurate tracking of payments, revenue, and expenses is **critical for decision-making, auditing, and profitability analysis**. Any errors in financial management could lead to **losses, incorrect billing, or compliance issues**, making this feature a **high-priority requirement**.

4.2.1.2. Stimulus/Response Sequences

Accountant selects "Financial Reports" in the system. System displays expenses, and financial summaries. Accountant selects "Process Payments" option. System retrieves outstanding balances and pending transactions. Accountant generates an end-of-day, weekly, or monthly financial report. System compiles revenue, expenses, and outstanding balances into a structured report.

4.2.1.3 Functional requirements

ID	Title	Description	Rationale	Dependencies
FR8	View Financial Reports	The system should allow accountants to access daily, weekly, and monthly financial reports.	To ensure accountants can monitor financial health and make informed decisions.	None
FR9	Retrieve Expense Records	The system should provide a detailed list of all recorded expenses, categorized by type and date.	To track operational costs and maintain financial transparency.	FR8
FR10	Process Payments	The system should allow accountants to process payments related to guest stays, services, and vendor transactions.	To ensure accurate revenue collection and maintain financial records.	None
FR11	Generate Financial Reports	The system should compile and generate structured reports, including revenue, expenses, and outstanding balances, for daily, weekly, and monthly periods.	To facilitate financial analysis and regulatory compliance.	FR8, FR9, FR10

4.3 Administrators

4.3.1. Staff & Hotel Management

4.3.1.1. Description and Priority

The **Admin Management System** allows administrators to oversee and manage hotel operations, including user roles, system settings, and access permissions. This feature ensures secure system management, proper role assignments, and smooth operational workflows. Priority is high, Administrators control user access, system configurations, and operational security. Without a well-defined admin system, unauthorized access, data inconsistencies, and operational inefficiencies could occur, making this a high-priority requirement.

4.3.1.2. Stimulus/Response Sequences

Admin selects "Staff Management" in the system. System displays a list of all hotel staff, including receptionists, accountants, housekeepers, and other employees. Admin adds, updates, or removes staff members. Admin selects "Financial Reports" to view financial data. Admin selects "Room Management" to check room statuses. Admin selects "Customer Information" to view guest details.

4.3.1.3. Functional Requirements

ID	Title	Description	Rationale	Dependencies
FR12	Manage Staff Accounts	The system should allow admins to add, update, or remove hotel staff.	To ensure proper staff organization and role assignments.	None
FR13	Assign Staff Roles & Permissions	The system should allow admins to assign roles (e.g., receptionist, accountant,	To control staff responsibilities and system access.	FR12

		housekeeping) with specific access levels.		
FR14	View Financial Reports	The system should allow admins to access revenue, expenses, and outstanding balances.	To monitor hotel finances and ensure financial stability.	None
FR15	View Room Status	The system should display real-time room availability, occupancy status, and housekeeping updates.	To ensure smooth hotel operations and room management.	None
FR16	View Customer Information	The system should allow admins to access guest profiles, stay history, and contact details.	To assist with customer service and record-keeping.	None

5 Use Cases

5.1 Creating a new account

The purpose of this use case is to describe the procedure of creating an account in the system.
Pre-conditions:

- None

Post-conditions**:**

- An account is created for the user.

Basic Flow:

1. The user is on the homepage.
2. The user clicks on the "New Account" link and is redirected to the account creation page.
3. The user enters all required information and clicks the "Create" button.
4. If the username already exists, an error message appears prompting the user to choose another username. If the username does not exist, a confirmation message appears indicating the account has been created successfully.

5.2 Deleting an Account

The purpose of this use case is to describe the procedure of deleting an account in the system.
Pre-conditions:

- The user must be logged into the system.

Post-conditions:

- The user's account is permanently deleted from the system.

Basic Flow:

1. The user navigates to the account settings page.
2. The user selects the "Delete Account" option.
3. A confirmation dialog appears, asking the user to confirm the deletion.
4. The user confirms the deletion by clicking the "Confirm" button.
5. The account is removed from the system, and a confirmation message is displayed.

5.3 Making a Reservation

The purpose of this use case is to describe the procedure for making a reservation in the hotel management system by a receptionist.
Pre-conditions:

- The receptionist must have an active account and be logged in.
- The system must have available rooms for booking.

Post-conditions:

- A reservation is created in the system.

Basic Flow:

1. The receptionist logs into the system.
2. The receptionist navigates to the reservation page.
3. The receptionist enters the guest's details.

4. The receptionist selects check-in and check-out dates.
5. The system displays available rooms.
6. The receptionist selects a room and confirms the reservation.
7. The system saves the reservation and displays a confirmation message.

Alternate Flow:

- If no rooms are available, the system displays a message

5.4 Canceling a Reservation

The purpose of this use case is to describe the procedure for canceling an existing reservation by a receptionist.

Pre-conditions:

- The receptionist must be logged into the system.
- The reservation must exist in the system.

Post-conditions:

- The reservation is removed from the system.

Basic Flow:

1. The receptionist logs into the system.
2. The receptionist navigates to the reservation management page.
3. The receptionist selects the reservation to be canceled.
4. The system prompts for confirmation.
5. The receptionist confirms the cancellation.
6. The reservation is removed, and a confirmation message is displayed.

5.5 Managing Employee Shifts

The purpose of this use case is to describe the procedure for managing employee shift schedules.

Pre-conditions:

- The user must be an admin or manager.

Post-conditions:

- The shift schedule is updated in the system.

Basic Flow:

1. The admin logs into the system.
2. The admin navigates to the shift management page.
3. The admin selects an employee and assigns or updates their shift schedule.
4. The system saves the changes and notifies the employee.

Alternate Flow:

- If there is a scheduling conflict, the system warns the admin before saving the changes.

5.6 Managing Financial Records

The purpose of this use case is to describe the procedure for managing financial transactions and reports in the system by an accountant.

Pre-conditions:

- The accountant must have an active account and be logged in.
- The system must have financial records available.

Post-conditions:

- Financial transactions and reports are updated in the system.

Basic Flow:

1. The accountant logs into the system.
2. The accountant navigates to the financial management section.
3. The accountant reviews income and expense records.
4. If needed, the accountant updates financial entries or corrects errors.
5. The system saves the changes and updates financial reports accordingly.

Alternate Flow:

- If there are missing or inconsistent records, the system alerts the accountant to review and correct them before proceeding.

5.7 Checking Room Availability

The purpose of this use case is to describe the procedure for checking room availability by a receptionist.

Pre-conditions:

- The receptionist must be logged into the system.

Post-conditions:

- The system displays available rooms.

Basic Flow:

1. The receptionist logs into the system.
2. The receptionist navigates to the room availability section.
3. The receptionist enters the check-in and check-out dates.
4. The system displays available rooms.
5. The receptionist informs the customer of available options.

5.8 Managing Customer Profiles

The purpose of this use case is to describe the procedure for adding, updating, or deleting customer profiles by a receptionist.

Pre-conditions:

- The receptionist must be logged into the system.

Post-conditions:

- The customer profile is updated in the system.

Basic Flow:

1. The receptionist logs into the system.
2. The receptionist navigates to the customer management section.
3. The receptionist adds a new customer or selects an existing customer to update information.
4. The system saves the changes and updates the profile.

Alternate Flow:

- If required fields are missing, the system prompts the receptionist to enter the necessary details.

6 Nonfunctional System Requirements

6.1Performance Requirements

ID	TAG	GIST	SCALE	METER	MUST	WISH
NF1	Response Time < ID: FR6 TITLE: Mobile Application – Search >	The speed of the search functionality.	The response time for a typical search operation from the moment the user presses "Search" to the display of results.	Measurements obtained from 500 search operations during testing.	No more than 5 seconds for 100% of searches.	No more than 1 second for 100% of searches.
NF2	Response Time < ID: FR4 TITLE: User Login and Register– Mobile/Web Application >	The speed of the login/authentication process.	The response time from the moment a user submits valid credentials until the	Measurements obtained from 1,000 login attempts during testing.	No more than 10 seconds for 100% of logins.	No more than 1 second for 100% of logins.

			main dashboard or home screen is displayed.			

6.2Safety Requirements

ID	TAG	GIST	SCALE	METER	MUST	WISH
NF3	Data backup	Prevent data loss	Automated daily backup	System logs for backup completion	No data loss in case of failure	Redundant backup storage
NF4	Error handling	Prevent service disruptions	Logging and recovery system	Error logs and system recovery tests	Automatic recovery without restart	Self-healing system

6.3Security Requirements

ID	TAG	GIST	SCALE	METER	MUST	WISH
NF5	Data Encryption	Encryption coverage and algorithms used	Encryption standards.	Security audit ensuring 100% of relevant fields are never stored in plaintext.	Hashing for passwords.	When storing passwords, use a secure hashing method with a randomly generated salt for each user.
NF6	User Authentication	Secure login	Unique username & password	Authentication success/failure logs	Only authorized users can log in	Multi-factor authentication

6.4 Software Quality Attributes

ID	TAG	GIST	SCALE	METER	MUST	WISH
NF7	Reliability	Ensure system availability	Uptime percentage	System uptime logs	90% uptime per month	95% uptime or higher with a robust failover setup.

6.5 Business Rules

ID	TAG	GIST	SCALE	METER	MUST	WISH
NF8	Legal Compliance < ID: FR9 TITLE: Manage User Accounts – Administrator Panel >	Ensure all data handling complies with KVKK (Turkish Data Protection Law) and related regulations.	Compliance check of data handling and user consent flows.	Legal audit verifying 100% alignment with KVKK guidelines for data collection, retention, and user consent.	All required user-consent forms and disclaimers are presented before data collection.	Automatic compliance updates and user notifications for any data policy changes.

7 Other Requirements

7.1 Database Requirements

7.1.1 Relational Storage

- The system relies on **PostgreSQL** to store **customer information, reservations, room availability, employee records, and financial transactions**.
- The database schema must be **normalized** to ensure data integrity and minimize redundancy.
- Relationships between **guests, reservations, and payments** should be properly maintained to prevent inconsistencies.

7.1.2 Scalability and Concurrency

- The database should **efficiently handle concurrent read/write requests**, ensuring smooth operation during peak hours.
- Support for **growing hotel chains** should be considered, allowing multiple branches to be managed within the same system.
- Indexing and caching strategies should be implemented to optimize performance for frequent queries, such as room availability checks.

7.2 Data Protection and Compliance

- The system must comply with **KVKK (Turkish Personal Data Protection Law) and GDPR** for handling customer and employee personal data.
- Users should be able to **request account deletion** and **opt-out of marketing communications**.
- Financial records and invoices must be **stored securely** and retained for the legally required duration.
- Role-based access control (RBAC) must be enforced to **restrict access to sensitive data** (e.g., only accountants can view financial reports).

8 References

8.1 Software Requirement Report Template for HW1.docx – Akdeniz University, Department of Computer Engineering.

8.2 Software Engineering Project Report.pdf – Group Project Instructions

9 Appendix A: Glossary

API (Application Programming Interface): A set of rules and protocols for building and interacting with software applications.

Backend: The server-side part of the application that handles business logic, database interactions, and API endpoints.

CRUD (Create, Read, Update, Delete): The four basic operations performed on a database.

FR (Functional Requirement): A requirement that describes a function or feature that the system must implement.

GUI (Graphical User Interface): A user interface that includes graphical elements, such as windows, icons, and buttons.

HTTPS (Hypertext Transfer Protocol Secure): An encrypted version of HTTP for secure communication over a network.

JWT (JSON Web Token): A compact, URL-safe means of representing claims between two parties for authentication and authorization.

.NET: A software development framework by Microsoft, used in this project for backend services and APIs.

PostgreSQL: A powerful, open-source relational database used for storing customer data, reservations, and financial records.

React: A JavaScript library for building user interfaces, primarily for web applications.

React Native: A JavaScript framework used to develop mobile applications for iOS and Android using a shared codebase.

RBAC (Role-Based Access Control): A security model where users are assigned specific roles that define their access permissions.

SRS (Software Requirements Specification): A comprehensive document describing the functionalities, constraints, interfaces, and design of a software product.

UI (User Interface): The visual part of the application that users interact with.

UX (User Experience): The overall experience and satisfaction a user has when interacting with the system.