Software Requirements Specification

Hotel Management System (HMS)

Version 1.0

Prepared by:
Software Engineering Team

Date: November 07, 2025

Team Members:

Abanob Thabet Fathy	(202320257)
Youssef Gerges Milad	(202320018)
Samer Yousry Labeb	(202320317)

Contents

Revisions 3

1 Introduction 4
1.1 Document Purpose 4
1.2 Product Scope 4
1.3 Intended Audience and Document Overview 4
1.4 Definitions, Acronyms and Abbreviations 4
1.5 References and Acknowledgments 5

2 Overall Description 6
2.1 Product Overview 6
2.2 Product Functionality 6
2.3 User Characteristics 6
2.4 Design and Implementation Constraints 7
2.5 Assumptions and Dependencies 7

3 Specific Requirements 8
3.1 External Interface Requirements 8
3.1.1 User Interfaces 8
3.1.2 Hardware Interfaces 9
3.1.3 Software Interfaces 9
3.2 Functional Requirements 10
3.2.1 User Registration and Authentication 10
3.2.2 Room Search and Booking 10
3.2.3 Check-in and Check-out Management 11
3.2.4 Billing and Payment Processing 12
3.2.5 Housekeeping and Maintenance 12
3.2.6 Reporting and Analytics 13
3.2.7 Administration and Configuration 13

4 Non-Functional Requirements 14
4.1 Performance Requirements 14
4.2 Safety and Security Requirements 14
4.3 Software Quality Attributes 15
4.3.1 Reliability 15
4.3.2 Scalability 15
4.3.3 Usability 15
4.3.4 Maintainability 15
4.3.5 Portability 16

5 Other Requirements 17
5.1 Logical Database Requirements 17
5.2 Legal and Regulatory Requirements 17

5.3 Environmental Requirements 17
5.4 Business Rules 18

Revisions

Version	**Primary Author(s)**	**Description**	**Date**
1.0	Software Engineering Team	Initial Release	Nov 07, 2025

Chapter 1

Introduction

1.1 Document Purpose

This document defines the requirements for the Hotel Management System (HMS), enabling hotel staff to manage room bookings, guest check-ins/check-outs, housekeeping operations, and billing processes. This SRS serves as a reference for development, testing, and maintenance teams.

1.2 Product Scope

HMS is a web-based platform designed to streamline hotel operations by:

- Enabling guests to search and book rooms online with secure payment
- Providing staff tools for efficient check-in/check-out management
- Facilitating housekeeping operations and maintenance tracking
- Automating billing and payment processing
- Offering management comprehensive reports and analytics

The system improves operational efficiency, reduces manual errors, enhances guest satisfaction, and provides real-time visibility into hotel operations.

1.3 Intended Audience and Document Overview

Intended for: Development team, project managers, system administrators, hotel stakeholders, and QA team.

Document structure: Functional and non-functional requirements, external interfaces, system features, design constraints, and quality attributes.

1.4 Definitions, Acronyms and Abbreviations

- **HMS:** Hotel Management System
- **SRS:** Software Requirements Specification
- **API:** Application Programming Interface
- **PCI-DSS:** Payment Card Industry Data Security Standard

- **GDPR:** General Data Protection Regulation
- **ADR:** Average Daily Rate
- **RevPAR:** Revenue Per Available Room
- **FOLIO:** Guest Billing Record
- **OTA:** Online Travel Agency
- **Guest:** Registered user booking and staying at the hotel
- **Receptionist:** Front desk staff managing check-ins/check-outs
- **Admin:** System administrator with full access

1.5 References and Acknowledgments

- IEEE Std 830-1998: Software Requirements Specifications
- Payment Card Industry Data Security Standard (PCI-DSS)
- ISO/IEC 25010: Software Product Quality Model
- GDPR (EU 2016/679)
- Egyptian Data Protection Law
- WCAG 2.1: Web Content Accessibility Guidelines

Chapter 2

Overall Description

2.1 Product Overview

HMS is a comprehensive web application integrating with:

- Third-party payment gateways (Stripe, PayPal, Visa)
- Email and SMS services for notifications
- Cloud storage for document management
- Reporting and analytics tools

The system provides role-based interfaces for guests, receptionists, housekeeping staff, managers, and administrators with real-time synchronization across all interfaces.

2.2 Product Functionality

- **User Management:** Registration, authentication, profile management, role-based access control
- **Room Search and Booking:** Real-time availability, advanced filtering, secure online booking, booking management
- **Check-in/Check-out Management:** Guest check-in with ID verification, automated room assignment, express check-out, walk-in registration
- **Billing Management:** Automatic folio generation, charge posting, multiple payment methods, tax-compliant invoices
- **Housekeeping Operations:** Real-time room status tracking, automated task assignment, maintenance reporting
- **Reporting and Analytics:** Occupancy reports, revenue analysis, performance metrics (ADR, RevPAR), guest history

2.3 User Characteristics

- **Guests:** General users with basic internet knowledge seeking convenient online booking
- **Receptionists:** Moderate computer skills, require quick access to bookings and room status

- **Housekeeping Staff:** Basic computer familiarity, need simple status update interfaces
- **Managers:** Experienced professionals requiring comprehensive reports and analytics
- **Administrators:** Technical personnel managing system configuration and maintenance

2.4 Design and Implementation Constraints

- **DC1:** Comply with GDPR and Egyptian Data Protection Law
- **DC2:** Use HTTPS and secure authentication (OAuth 2.0 or JWT)
- **DC3:** Payment processing must conform to PCI-DSS Level 1
- **DC4:** Support English and Arabic with RTL layout for Arabic
- **DC5:** Support browsers (Chrome, Firefox, Safari, Edge)
- **DC6:** Use responsive web design principles
- **DC7:** Implement role-based access control
- **DC8:** Maintain audit logs for all transactions

2.5 Assumptions and Dependencies

- Internet connection required for system access
- Third-party payment gateways and APIs available
- Hotels provide accurate room inventory and pricing
- Staff have basic computer literacy
- Cloud infrastructure maintains service-level agreements
- Users have valid email addresses and phone numbers

Chapter 3

Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

HMS provides responsive interfaces for web browsers (Chrome, Firefox, Safari, Edge), following WCAG 2.1 Level AA accessibility standards.

Guest Portal:

- Clean booking interface
- Room search with calendar
- Secure payment forms
- Booking history dashboard

Front Desk Dashboard:

- Real-time room status grid
- Quick guest search
- One-click check-in/check-out
- Folio management

Housekeeping Interface:

- Task list view
- Status update functionality
- Maintenance reporting with photo upload capability

Management Dashboard:

- Interactive analytics charts
- Report generation
- Configuration interfaces
- Performance metrics

Admin Console:

- System configuration

- Room inventory management
- Rate configuration
- Audit logs
- Backup management

All interfaces support English and Arabic with proper RTL layout, optimized icons, colors, and typography.

3.1.2 Hardware Interfaces

Desktop Systems:

- Standard keyboard/mouse
- Minimum 1280x720 resolution

Peripherals:

- Network printers for invoices/registration cards
- Card readers
- Receipt printers

Server Infrastructure:

- Scalable cloud infrastructure
- Load balancers
- Redundant storage

3.1.3 Software Interfaces

Payment Gateways:

- Stripe, PayPal, Visa Direct APIs with OAuth 2.0
- JSON format
- PCI-DSS compliant tokenization

Email Service:

- SMTP (TLS 1.2+) or SendGrid API
- HTML templates
- Delivery tracking

SMS Gateway:

- Twilio API for notifications and OTP

- International delivery support

Cloud Storage:

- AWS S3 or Azure Blob Storage
- Encrypted document storage

Maps API:

- Google Maps for location services (optional)

Database:

- MySQL 8.0+ or PostgreSQL 13+
- Redis for caching

All APIs use RESTful design with JSON over HTTPS, proper authentication, and error handling with retry logic.

3.2 Functional Requirements

3.2.1 User Registration and Authentication

- **FR1:** Allow guest registration with email/phone and secure password (min 8 chars, mixed case, numbers)
- **FR2:** Verify accounts via OTP sent by email/SMS within 2 minutes
- **FR3:** Support login with credentials or social accounts (Google/Apple)
- **FR4:** Provide password reset via email/SMS
- **FR5:** Allow profile updates (name, contact, preferences)
- **FR6:** Enable admin creation of staff accounts with role assignment
- **FR7:** Implement 15-minute session timeout
- **FR8:** Lock accounts after 5 failed login attempts
- **FR9:** Maintain login history with timestamp, IP, and device

3.2.2 Room Search and Booking

- **FR10:** Display available rooms based on check-in/check-out dates
- **FR11:** Allow filtering by room type, price, amenities
- **FR12:** Show room details with images, descriptions, pricing, occupancy
- **FR13:** Support dynamic pricing (weekend, seasonal, promotional)
- **FR14:** Allow multiple room selection in single booking

- **FR15:** Calculate total cost including taxes and fees
- **FR16:** Support promotional codes and discounts
- **FR17:** Enable guest preferences (bed type, floor, smoking status)
- **FR18:** Process secure online payments via payment gateways
- **FR19:** Generate unique booking confirmation number
- **FR20:** Send confirmation via email and SMS
- **FR21:** Allow viewing, modifying, and canceling bookings
- **FR22:** Prevent overbooking with real-time inventory control
- **FR23:** Support group bookings

3.2.3 Check-in and Check-out Management

- **FR24:** Search bookings by confirmation number, name, or phone
- **FR25:** Display upcoming arrivals filtered by date
- **FR26:** Enable check-in with ID verification and digital signature
- **FR27:** Auto-assign rooms based on preferences and availability
- **FR28:** Allow manual room assignment with reason logging
- **FR29:** Generate and print registration cards
- **FR30:** Update room status to ”Occupied” upon check-in
- **FR31:** Support early check-in and late check-out with charges
- **FR32:** Process check-out with folio review and payment
- **FR33:** Update room status to ”Dirty” upon check-out
- **FR34:** Handle walk-in guests with immediate booking
- **FR35:** Support express check-out for pre-authorized payments
- **FR36:** Maintain guest history (previous stays, preferences)

3.2.4 Billing and Payment Processing

- **FR37:** Auto-generate folios upon check-in
- **FR38:** Allow posting additional charges (minibar, services)
- **FR39:** Calculate and post daily room charges, taxes, fees
- **FR40:** Support multiple payment methods (cash, cards, transfers)
- **FR41:** Process credit cards using tokenization (PCI-DSS compliant)
- **FR42:** Generate tax-compliant invoices in PDF format
- **FR43:** Support split billing for multiple payers
- **FR44:** Allow manager-approved refunds with documentation
- **FR45:** Maintain payment history with all transactions
- **FR46:** Support corporate billing with direct invoicing
- **FR47:** Handle partial payments and balance tracking
- **FR48:** Auto-email invoices upon check-out
- **FR49:** Support void/correction transactions with authorization

3.2.5 Housekeeping and Maintenance

- **FR50:** Track room status: Vacant-Dirty, Occupied-Dirty, Vacant-Clean, Occupied-Clean, Cleaning in Progress, Inspected, Out of Order
- **FR51:** Auto-generate daily cleaning task lists
- **FR52:** Assign tasks to specific staff with workload balancing
- **FR53:** Allow status updates through web interface with timestamp
- **FR54:** Require supervisor inspection before "Vacant-Clean" status
- **FR55:** Enable maintenance reporting with photos, priority, location
- **FR56:** Notify front desk when rooms are clean
- **FR57:** Track maintenance requests from submission to completion
- **FR58:** Provide housekeeping dashboard with task completion metrics
- **FR59:** Allow marking rooms "Out of Order" with reason
- **FR60:** Generate housekeeping performance reports

3.2.6 Reporting and Analytics

- **FR61:** Generate daily occupancy reports with percentage
- **FR62:** Calculate and display ADR for date ranges
- **FR63:** Calculate and display RevPAR with trends
- **FR64:** Provide revenue reports by category (rooms, services, taxes)
- **FR65:** Generate guest demographics reports
- **FR66:** Maintain guest history with booking patterns
- **FR67:** Allow custom reports with user-defined parameters
- **FR68:** Support export in PDF, Excel, CSV formats
- **FR69:** Provide graphical dashboards with interactive charts
- **FR70:** Generate forecast reports based on trends
- **FR71:** Analyze booking sources (direct, OTA, walk-ins)
- **FR72:** Support custom date range filtering

3.2.7 Administration and Configuration

- **FR73:** Allow adding, editing, deactivating rooms
- **FR74:** Configure room types with descriptions, rates, amenities
- **FR75:** Support seasonal pricing with date ranges
- **FR76:** Configure tax rates and service charges
- **FR77:** Provide comprehensive user management
- **FR78:** Implement role-based access control with custom permissions
- **FR79:** Maintain detailed audit logs of all actions
- **FR80:** Support automated daily backups with retention policy
- **FR81:** Allow data restoration from specific backup points
- **FR82:** Configure hotel settings (name, contact, check-in times, policies)
- **FR83:** Configure email/SMS notification templates
- **FR84:** Configure payment gateway settings
- **FR85:** Provide system health monitoring with alerts

Chapter 4

Non-Functional Requirements

4.1 Performance Requirements

- **PR1:** Support 200 concurrent users without degradation
- **PR2:** Load web pages in under 2 seconds for 90% of users (10 Mbps+)
- **PR3:** Average response time 2 seconds for user actions
- **PR4:** Process 95% of transactions within 3 seconds at peak
- **PR5:** Database queries return results within 500ms (95% of requests)
- **PR6:** Handle 5x normal traffic during peak seasons
- **PR7:** Support horizontal scaling for increased load
- **PR8:** API throughput minimum 100 requests/second

4.2 Safety and Security Requirements

- **SR1:** Encrypt sensitive data with AES-256 at rest
- **SR2:** Use HTTPS with TLS 1.3+ for all transmission
- **SR3:** Hash passwords with bcrypt algorithm
- **SR4:** Comply with PCI-DSS Level 1 using tokenization
- **SR5:** Implement role-based access control (RBAC)
- **SR6:** Auto-logout after 15 minutes inactivity
- **SR7:** Comply with GDPR and Egyptian Data Protection Law
- **SR8:** Lock accounts after 5 failed login attempts
- **SR9:** Implement MFA for administrator accounts
- **SR10:** Log all activities with timestamp, user, IP, action
- **SR11:** Retain audit logs minimum 90 days in tamper-proof format
- **SR12:** Perform automated encrypted backups daily with offsite storage
- **SR13:** Prevent SQL injection via parameterized queries

- **SR14:** Prevent XSS attacks through output encoding
- **SR15:** Implement CSRF protection tokens

4.3 Software Quality Attributes

4.3.1 Reliability

- Maintain 99.9% uptime annually (max 8.76 hours downtime/year)
- Auto-recover from failures within 5 minutes
- MTBF exceeds 720 hours (30 days)
- Implement graceful degradation
- Maintain 99.99% data integrity

4.3.2 Scalability

- Support 50-500 rooms without architectural changes
- Scale horizontally to handle 10x traffic increase
- Database scales linearly with read replicas
- Support multi-property management

4.3.3 Usability

- Intuitive interface requiring ¡ 1 hour training
- Task completion rates ¿ 95% for common workflows
- Comply with WCAG 2.1 Level AA accessibility
- Provide context-sensitive help and tooltips
- Clear, actionable error messages
- Support English and Arabic with proper RTL

4.3.4 Maintainability

- Modular codebase with clear separation of concerns
- Technical debt ¡ 5%
- Support zero-downtime deployments
- Code documentation coverage ¿ 70%
- Use version control with comprehensive commit messages
- Automated test coverage ¿ 80% for critical logic

4.3.5 Portability

- Deploy on AWS, Azure, DigitalOcean
- Run on Linux (Ubuntu 20.04+, CentOS 8+, Debian 11+)
- Use Docker containerization
- Support MySQL and PostgreSQL via abstraction layer
- Externalize configuration using environment variables

Chapter 5

Other Requirements

5.1 Logical Database Requirements

- **LDB1:** Maintain normalized relational database
- **LDB2:** Include core tables: Users, Guests, Rooms, RoomTypes, Bookings, Folios, Payments, HousekeepingTasks, MaintenanceRequests
- **LDB3:** Use foreign key constraints for referential integrity
- **LDB4:** Implement indexes on frequently queried columns
- **LDB5:** Use database transactions for atomic operations
- **LDB6:** Enforce data type constraints at schema level
- **LDB7:** Implement soft deletion for audit trail records
- **LDB8:** Support timestamp fields (created_at, updated_at)
- **LDB9:** Maintain lookup tables for standardized values
- **LDB10:** Support full-text search for guest names and bookings

5.2 Legal and Regulatory Requirements

- Comply with Egyptian Data Protection Law and GDPR
- Implement data subject rights (access, rectification, erasure, portability)
- Adhere to local tax laws and regulations
- Generate tax-compliant invoices meeting authority requirements
- Maintain records for minimum 7 years
- Require verified business licenses for hotel registration

5.3 Environmental Requirements

- Operate with minimum 10 Mbps internet bandwidth
- Server hardware operates in 15°C to 35°C ambient temperature
- Provide informative error messages during network outages

5.4 Business Rules

1. **No Overbooking:** Prevent booking beyond available inventory
2. **Cancellation Policy:** Free cancellation up to 24 hours before check-in; charges apply within 24 hours
3. **Check-in/Check-out Times:** Standard check-in 15:00, check-out 12:00; early/late requests incur charges
4. **Payment Policy:** Full prepayment for non-refundable rates; credit card guarantee for standard rates
5. **Minimum Stay:** One night minimum; seasonal minimums configurable
6. **Child Policy:** Children under specified age stay free; different rates for older children
7. **Commission:** OTA bookings incur commission fees per platform agreement