**Software Requirements Specification (SRS)**

**Conference Room Booking Application**

**Version History**

| Version | Date | Author | Description |
| --- | --- | --- | --- |
| 1.1 | 2024-12-19 | AI Assistant | Updated with Functional Descriptions and Data Validation |

**Table of Contents**

1. Introduction
2. Overall Description
3. System Features and Requirements
4. External Interface Requirements
5. Non-Functional Requirements
6. Other Requirements
7. Data Validation Rules

**1. Introduction**

**1.1 Purpose**

This document describes the software requirements for the Conference Room Booking Application. It is intended for stakeholders, project managers, developers, and testers involved in the development process.

**1.2 Scope**

The Conference Room Booking Application is a web-based system that enables employees to search for, view availability of, and book conference rooms within their organization. The system will prevent double-booking, manage user accounts, and provide administrative controls for room management.

**1.3 Definitions, Acronyms, and Abbreviations**

* **SRS**: Software Requirements Specification
* **MVP**: Minimum Viable Product
* **UI**: User Interface
* **API**: Application Programming Interface
* **Admin**: System Administrator

**1.4 References**

* Agile Development Methodology
* Previous Epic and User Story documentation

**1.5 Overview**

This document is organized into sections covering overall description, specific requirements, interface requirements, and non-functional requirements.

**2. Overall Description**

**2.1 Product Perspective**

The Conference Room Booking Application is a new, self-contained system that may eventually integrate with existing corporate systems such as:

* Email systems for notifications
* Calendar systems (Google Calendar, Outlook)
* Corporate Active Directory for authentication

**2.2 Product Functions**

* Room discovery and availability checking
* User registration and authentication
* Room booking and management
* Administrative room and system management
* Notifications and reporting

**2.3 User Characteristics**

| User Type | Characteristics | Technical Proficiency |
| --- | --- | --- |
| **Regular User** | Employees needing to book meeting spaces | Basic computer literacy |
| **Administrator** | Facilities managers, IT staff | Moderate to high technical proficiency |

**2.4 Constraints**

* Must be web-accessible
* Must support concurrent users
* Must prevent double-booking
* Must be responsive for tablet/mobile use

**2.5 Assumptions and Dependencies**

* Users have valid corporate email addresses
* Reliable network connectivity is available
* Modern web browsers are used (Chrome, Firefox, Safari, Edge)

**3. System Features and Requirements**

**3.1 Epic 1: Core Room Discovery & Basic Booking**

**3.1.1 Description**

Users can find available rooms and make basic bookings.

**3.1.2 Functional Requirements**

**FR-1.1: Room List Display**

* **Description**: The system shall display a list of all conference rooms
* **Functional Description**:
  + System retrieves active rooms from database
  + Displays rooms in a responsive grid or list layout
  + Shows room name, capacity, and location for each room
  + Handles empty states when no rooms are available
  + Supports pagination if room count exceeds display threshold
* **Data Validation**:
  + Room names must be non-empty strings (1-50 characters)
  + Capacity must be positive integers (1-500)
  + Location must be non-empty strings (1-100 characters)

**FR-1.2: Room Details View**

* **Description**: The system shall display detailed information for a selected room
* **Functional Description**:
  + User selects a room from the list to view details
  + System displays comprehensive room information
  + Shows availability status in real-time
  + Provides booking initiation interface
* **Data Validation**:
  + Room ID must be valid and exist in database
  + Feature lists must contain only predefined feature codes
  + Image URLs must be properly formatted

**FR-1.3: Availability Checking**

* **Description**: The system shall allow users to check room availability for specific dates and times
* **Functional Description**:
  + User selects date and time range for availability check
  + System queries bookings for conflicting time slots
  + Displays visual indicators (green/red) for availability status
  + Updates availability in real-time as user modifies time selection
* **Data Validation**:
  + Date must be current or future date (not in past)
  + Start time must be before end time
  + Time range must respect business hours (8 AM - 6 PM)
  + Booking duration minimum 15 minutes, maximum 4 hours

**FR-1.4: Room Booking**

* **Description**: The system shall allow users to book available rooms
* **Functional Description**:
  + User completes booking form with meeting details
  + System validates room availability at submission time
  + Creates booking record in database
  + Sends confirmation to user
  + Updates room availability status
* **Data Validation**:
  + Meeting title: required, 1-100 characters
  + Organizer name: required, 2-50 characters
  + Participant count: optional, positive integer
  + Time slot: must not conflict with existing bookings

**FR-1.5: Double-Booking Prevention**

* **Description**: The system shall prevent concurrent bookings of the same room and time slot
* **Functional Description**:
  + Implements database-level constraints for time conflicts
  + Performs real-time availability checks before confirming bookings
  + Handles race conditions with optimistic locking
  + Provides clear error messages for booking conflicts
* **Data Validation**:
  + Room ID + start time + end time must be unique combination
  + Time overlap detection using database queries
  + Concurrent request handling with transaction isolation

**3.2 Epic 2: User Account & Profile Management**

**3.2.1 Description**

Users can create accounts and manage their profiles and bookings.

**3.2.2 Functional Requirements**

**FR-2.1: User Registration**

* **Description**: The system shall allow new users to register with required information
* **Functional Description**:
  + Collects user information through registration form
  + Validates email uniqueness
  + Hashes password before storage
  + Creates user profile in database
  + Sends welcome email upon successful registration
* **Data Validation**:
  + Email: valid format, unique in system, corporate domain required
  + Password: minimum 8 characters, requires uppercase, lowercase, number
  + Full Name: 2-50 characters, alphabetic with spaces allowed
  + Department: optional, max 50 characters

**FR-2.2: User Authentication**

* **Description**: The system shall allow registered users to log in securely
* **Functional Description**:
  + Verifies user credentials against stored hashes
  + Creates secure session upon successful authentication
  + Manages session timeout (30 minutes inactivity)
  + Provides secure logout functionality
  + Tracks login attempts for security
* **Data Validation**:
  + Email: must exist in system
  + Password: verified against stored hash
  + Session tokens: cryptographically secure
  + Failed attempts: max 5 before temporary lockout

**FR-2.3: Booking Management**

* **Description**: The system shall display and manage user's bookings
* **Functional Description**:
  + Retrieves user's bookings from database
  + Separates upcoming and past bookings
  + Provides cancellation functionality with confirmation
  + Updates booking status in real-time
  + Sends cancellation notifications
* **Data Validation**:
  + User can only cancel their own bookings
  + Cancellation allowed up to 15 minutes before meeting start
  + Booking status transitions validated
  + Audit trail maintained for cancellations

**FR-2.4: Password Recovery**

* **Description**: The system shall provide secure password reset functionality
* **Functional Description**:
  + Generates secure, time-limited reset tokens
  + Sends reset links to verified email addresses
  + Validates token expiration and usage
  + Requires password confirmation on reset
  + Invalidates existing sessions after password change
* **Data Validation**:
  + Reset tokens: valid for 1-hour, single use
  + Email: must belong to registered user
  + New password: meets security requirements
  + Token usage: tracked and invalidated after use

**3.3 Epic 3: Advanced Booking Management & Conflict Prevention**

**3.3.1 Description**

Enhanced booking features and robust conflict handling.

**3.3.2 Functional Requirements**

**FR-3.1: Booking Editing**

* **Description**: The system shall allow users to modify existing bookings
* **Functional Description**:
  + Provides edit interface for booking details
  + Validates changes against availability
  + Updates all related records
  + Notifies participants of changes
  + Maintains edit history
* **Data Validation**:
  + Time changes must not create conflicts
  + Title and description length limits
  + Participant count within room capacity
  + Edit allowed up to 1 hour before meeting

**FR-3.2: Meeting Invitations**

* **Description**: The system shall allow booking organizers to invite attendees
* **Functional Description**:
  + Collects attendee email addresses
  + Sends invitation emails with meeting details
  + Tracks invitation responses
  + Updates attendee lists
  + Handles duplicate invitations
* **Data Validation**:
  + Email addresses: valid format
  + Max attendees: room capacity minus organizer
  + Duplicate emails: prevented
  + External emails: allowed with warning

**3.4 Epic 4: Room & Resource Management (Admin)**

**3.4.1 Description**

Administrative functions for managing rooms and system configuration.

**3.4.2 Functional Requirements**

**FR-4.1: Room Management**

* **Description**: The system shall allow administrators to manage room inventory
* **Functional Description**:
  + Provides room creation/editing interface
  + Validates room configurations
  + Handles room deactivation (soft delete)
  + Maintains room history
  + Prevents deletion of rooms with future bookings
* **Data Validation**:
  + Room names: unique within location
  + Capacity: reasonable limits (1-500)
  + Features: from predefined list
  + Location: valid building/floor format

**FR-4.2: Maintenance Scheduling**

* **Description**: The system shall allow administrators to schedule room maintenance
* **Functional Description**:
  + Creates maintenance time blocks
  + Prevents bookings during maintenance
  + Notifies affected users of maintenance
  + Supports recurring maintenance schedules
  + Allows maintenance cancellation
* **Data Validation**:
  + Maintenance periods: cannot overlap
  + Future bookings: notified and cancelled/rescheduled
  + Duration: reasonable limits (1 minute - 24 hours)

**3.5 Epic 5: Enhanced Discovery & Filtering**

**3.5.1 Description**

Advanced search and filtering capabilities.

**3.5.2 Functional Requirements**

**FR-5.1: Room Filtering**

* **Description**: The system shall allow filtering rooms by various criteria
* **Functional Description**:
  + Provides filter interface with multiple criteria
  + Updates results in real-time
  + Shows available count for each filter
  + Persists filter settings during session
  + Handles complex filter combinations
* **Data Validation**:
  + Capacity filters: min <= max
  + Feature selections: valid feature codes
  + Date ranges: valid, not in past
  + Time ranges: within business hours

**3.6 Epic 6: Notifications & Calendar Integration**

**3.6.1 Description**

Email notifications and external calendar integration.

**3.6.2 Functional Requirements**

**FR-6.1: Email Notifications**

* **Description**: The system shall send email notifications for booking events
* **Functional Description**:
  + Generates appropriate email templates
  + Includes all relevant booking details
  + Handles email delivery failures
  + Provides notification preferences
  + Supports HTML and plain text formats
* **Data Validation**:
  + Email addresses: valid format
  + Template variables: properly escaped
  + Attachment sizes: within limits
  + Send times: appropriate for event type

**3.7 Epic 7: Reporting & Analytics (Admin)**

**3.7.1 Description**

Usage reporting and analytics for administrators.

**3.7.2 Functional Requirements**

**FR-7.1: Usage Reporting**

* **Description**: The system shall generate room usage reports
* **Functional Description**:
  + Collects usage data from bookings
  + Calculates utilization rates
  + Identifies peak usage patterns
  + Generates visual charts and graphs
  + Supports date range selection
* **Data Validation**:
  + Date ranges: valid, ordered correctly
  + Report parameters: within reasonable limits
  + Data aggregation: accurate calculations
  + Access control: admin privileges required

**4. External Interface Requirements**

**4.1 User Interfaces**

* **Web Interface**: Responsive design supporting desktop, tablet, and mobile
* **Admin Interface**: Separate administrative panel with enhanced controls
* **Authentication Interface**: Clean, secure login and registration forms

**4.2 Hardware Interfaces**

* Standard web server hardware
* Database server
* Email server for notifications

**4.3 Software Interfaces**

* **Database**: Relational database (MySQL, PostgreSQL)
* **Web Server**: Apache/Nginx
* **Programming Language**: JavaScript (Node.js/Python/Java/etc.)
* **Email Service**: SMTP integration

**4.4 Communications Interfaces**

* HTTP/HTTPS for web access
* SMTP for email notifications
* Potential future REST API for integrations

**5. Non-Functional Requirements**

**5.1 Performance Requirements**

* **Response Time**: Page loads under 2 seconds
* **Availability**: 99.5% uptime during business hours
* **Concurrent Users**: Support for 500+ concurrent users
* **Booking Transaction**: Complete within 3 seconds

**5.2 Safety Requirements**

* No specific safety requirements (software-only system)

**5.3 Security Requirements**

* **Authentication**: Secure password storage (hashing and salting)
* **Authorization**: Role-based access control
* **Session Management**: Secure session handling with timeout
* **Data Protection**: Protection against SQL injection and XSS attacks

**5.4 Software Quality Attributes**

* **Reliability**: System should recover gracefully from failures
* **Maintainability**: Modular code structure with documentation
* **Portability**: Cross-browser compatibility
* **Scalability**: Ability to handle increasing numbers of rooms and users
* **Usability**: Intuitive interface requiring minimal training

**6. Other Requirements**

**6.1 Database Requirements**

* User data storage with proper indexing
* Booking history maintenance
* Room configuration data
* Audit trails for administrative actions

**6.2 Operational Requirements**

* Regular database backups
* System monitoring and alerting
* Log rotation and management

**6.3 Business Rules**

* Rooms can only be booked during business hours (configurable)
* Maximum booking duration: 4 hours (configurable)
* Booking lead time: Up to 90 days in advance
* Cancellation policy: Up to 15 minutes before meeting start

**6.4 Implementation Priorities**

**Phase 1: MVP (Minimum Viable Product)**

* Epic 1: Core Room Discovery & Basic Booking
* Epic 2: User Account & Profile Management

**Phase 2: Enhanced Features**

* Epic 3: Advanced Booking Management
* Epic 4: Room & Resource Management (Admin)
* Epic 5: Enhanced Discovery & Filtering

**Phase 3: Advanced Features**

* Epic 6: Notifications & Calendar Integration
* Epic 7: Reporting & Analytics (Admin)

**7. Data Validation Rules**

**7.1 User Data Validation**

| Field | Data Type | Constraints | Validation Rules |
| --- | --- | --- | --- |
| Email | String | Required, Unique | Valid email format, corporate domain required |
| Password | String | Required | Min 8 chars, uppercase, lowercase, number |
| Full Name | String | Required | 2-50 characters, alphabetic and spaces only |
| Department | String | Optional | Max 50 characters |

**7.2 Room Data Validation**

| Field | Data Type | Constraints | Validation Rules |
| --- | --- | --- | --- |
| Room Name | String | Required, Unique | 1-50 characters, alphanumeric and spaces |
| Location | String | Required | 1-100 characters |
| Capacity | Integer | Required | 1-500, positive integer |
| Features | Array | Optional | Predefined feature codes only |

**7.3 Booking Data Validation**

| Field | Data Type | Constraints | Validation Rules |
| --- | --- | --- | --- |
| Meeting Title | String | Required | 1-100 characters |
| Start Time | DateTime | Required | Future date, within business hours |
| End Time | DateTime | Required | After start time, max 4 hours duration |
| Participants | Integer | Optional | Positive integer, less than room capacity |

**7.4 Business Logic Validation**

* **Time Conflicts**: No overlapping bookings for same room
* **Capacity Limits**: Participants cannot exceed room capacity
* **Business Hours**: Bookings only within configured business hours
* **Advance Booking**: Maximum 90 days in advance
* **Cancellation Window**: Up to 15 minutes before meeting start

**7.5 Input Sanitization**

* **SQL Injection Prevention**: Parameterized queries, input escaping
* **XSS Prevention**: HTML encoding of user-generated content
* **File Upload Validation**: Type and size restrictions
* **Email Injection Prevention**: Header validation for email fields

**7.6 Cross-Field Validation**

* End time must be after start time
* Participant count must be less than or equal to room capacity
* Recurring booking patterns must maintain consistency
* Maintenance schedules cannot conflict with existing bookings

**Approval**

| Role | Name | Signature | Date |
| --- | --- | --- | --- |
| Product Owner |  |  |  |
| Project Manager |  |  |  |
| Lead Developer |  |  |  |
| QA Lead |  |  |  |

**Document Status:** Draft | In Review | Approved | Archived