Schedulink - Document

Monday, 11 August 2025 8:36 PM

Schedulink
Smart Appointment Scheduler for Modern Businesses

Schedulink - Case Study – System Description

This system is an online appointment booking platform. It helps people and businesses organize and manage appointments easily, all in one place.

What it does:

- Admin created user accounts.
- Allows users to book appointments for services or meetings.
- Businesses or service providers can set up available time slots.
- Users can see available slots and book them instantly.
- Keeps track of all appointments, users, and available times.

Why it exists:

- To save time and avoid confusion with phone calls, emails, or paper schedules.
- Makes it easy for both customers and businesses to manage bookings.
- Reduces double-booking and missed appointments.

How it works:

- There's a website (the frontend) where users interact—register, log in, and book appointments.
- Behind the scenes, a server (the backend) handles all the data, like who booked what and when.
- The system stores all information securely in a database.
- Everything is designed to be easy to use, fast, and reliable, whether you're a customer or a business owner.

Key benefits:

- Convenience: Users can book appointments anytime, from anywhere, without needing to call or visit in person.
- Time-saving: Automates scheduling, reducing back-and-forth communication and manual tracking.
- Fewer mistakes: Prevents double-booking and missed appointments by keeping everything organized in one place.

keeping everything organized in one place.
- Better organization: Both users and businesses can easily view, manage, and update appointments.
- Professional image: Businesses appear more modern and efficient by offering online booking.
- Scalability: The system can handle many users and appointments as the business grows.
- Security: User and appointment data are stored safely and securely.
- Improved customer experience: Makes booking fast and easy, leading to happier customers.

In short, it's a digital tool to make booking and managing appointments simple and stress-free for everyone.

Schedulink - Functional Requirements:

1. User Management

Requirement:

Admins can create, edit, and deactivate user accounts. Users can log in and manage their profiles.

Example:

Admin John uses the admin panel to add a new user, Sarah, with her email address. Sarah receives her credentials, logs in, and updates her contact number.

2. Time Slot Management

Requirement:

Businesses or service providers can set up, modify, and delete available time slots for appointments. Slots can be recurring or one-time.

Example:

Dental Clinic "BrightSmiles" sets up appointment slots for Tuesday:

- 09:00, 09:30, 10:00, 10:30, ... until 17:00
  If the 10:00 AM slot is booked, it is no longer available for other users.

3. Appointment Booking

3. Appointment Booking

Requirement:

Users can view available time slots and book appointments. The system prevents double-booking of slots and provides instant confirmation.

Example:

Sarah wants an appointment at BrightSmiles on Tuesday at 11:30 AM. She logs in, sees the available times, and books 11:30 AM. The system immediately confirms her booking and marks the slot as taken.

4. Appointment Management

Requirement:

Users can view, reschedule, or cancel their appointments. Businesses can view and manage all bookings.

Example:

Sarah realizes she cannot make her 11:30 AM appointment, so she logs in and reschedules to 13:00 PM. The system updates her appointment and frees up the original slot.

5. Notifications

Requirement:

Users receive notifications for booking confirmations, reminders, cancellations, and updates. Businesses get notified about new bookings and cancellations.

Example:

Sarah books an appointment and receives a confirmation email. She gets an SMS reminder one hour before her appointment. The clinic receives an email when Sarah reschedules.

6. Admin Dashboard

Requirement:

Admins can manage all users, businesses, time slots, and appointments, but do not

use 2FA for login.

Example:

Admin John logs into the dashboard, views statistics on appointments, adds a new user, and deactivates an inactive account.

7. Business Dashboard

Requirement:

Businesses can view and manage their available slots and bookings.

Example:

The owner of BrightSmiles views all upcoming appointments, adjusts slot availability for next week, and prints a daily schedule.

8. Security & Data Protection

Requirement:

All user and appointment data is stored securely. Only authenticated users can access their own information.

Example:

Sarah’s personal information and appointment history are securely stored in the database. She can only see her own bookings.

9. Scalability

Requirement:

The system can support multiple businesses, thousands of users, and appointments without performance issues.

Example:

During holiday periods, BrightSmiles receives hundreds of bookings in a day, but the platform remains fast and reliable.

10. Audit & Logging

Requirement:

All important actions (bookings, changes, logins) are logged for auditing and troubleshooting.

Example:

If there's a dispute about a missed appointment, the admin checks the logs to see when the appointment was booked or changed.

Summary Table

Requirement	User Example	Business/Admin Example
User Management	Log in, update profile	Create/deactivate user accounts
Time Slot Management	See available slots	Set up/manage available slots
Appointment Booking	Book/cancel/reschedule appointments	View/manage all bookings
Notifications	Receive reminders, confirmations	Get alerts for new/canceled bookings
Dashboards	View own appointments	View stats, adjust availability
Security	Data privacy, secure access	Secure data storage
Scalability	No slowdowns with many users	High-volume booking support
Audit & Logging	All actions tracked	Investigate issues

Schedulink – Non-Functional Requirements

1. Performance

- The system should handle at least 1,000 concurrent users without noticeable delay.
- Page load times should not exceed 2 seconds under normal load. **Example:** During peak hours, users can view and book slots with quick responses.

2. Scalability

- The platform must support easy addition of new businesses and users without major redesign.
- The backend and database should be able to scale horizontally (e.g.,

distributed servers, sharding). **Example:** If 10 new clinics join, the platform accommodates their slots and appointments seamlessly.

3. Availability & Reliability

- The system should have 99.9% uptime, except for scheduled maintenance.
- Automatic failover and backup in case of hardware or software failure. **Example:** If a server goes down, users still access the platform via a backup server.

4. Security

- All user data and appointment information must be encrypted at rest and in transit (TLS/SSL).
- Only authenticated users can access personal/account information.
- Admin/business roles must have access controls to prevent unauthorized changes. **Example:** Sarah's appointment details are protected from unauthorized access and interception.

5. Data Integrity

- Changes to appointment data should be atomic and consistent (no double-booking, no partial updates).
- The system must prevent and detect conflicting bookings. **Example:** If two users try to book the same slot simultaneously, only one succeeds; the other gets an error.

6. Maintainability

- The codebase should be modular, documented, and follow best practices to ease bug fixes and feature additions.
- The system should support automated testing (unit, integration). **Example:** Developers can quickly fix a bug or add a new notification type with minimal risk of breaking existing features.

7. Usability

- The user interface should be intuitive, responsive, and accessible (WCAG 2.1 compliance).
- Minimal clicks required to book, view, or manage appointments. **Example:** Sarah can book, reschedule, or cancel with just a few clicks, even on mobile.

8. Portability

- The web application should work on major browsers (Chrome, Firefox, Safari, Edge).
- The system should be deployable on different cloud platforms (AWS, Azure,

GCP). **Example:** Users on both Windows and Mac devices have the same experience.

9. Backup & Disaster Recovery

- Daily backups of all appointment and user data.
- Recovery procedures must ensure data can be restored within 2 hours. **Example:** If data is lost due to a system crash, it can be restored from the latest backup.

10. Logging & Monitoring

- The system must log errors, warnings, and important actions (e.g., bookings, cancellations).
- Real-time monitoring for performance, security breaches, and system health. **Example:** Admins are alerted if there's a spike in booking failures.

11. Legal & Compliance

- The system must comply with applicable data protection laws (e.g., GDPR).
- Users must be informed of data use and privacy policies. **Example:** Sarah sees a privacy notice and consents before creating her account.

Summary Table

Non-Functional Requirement	Description & Example
Performance	Fast response times, quick bookings
Scalability	Supports growth, more users/businesses easily
Availability	High uptime, failover in place
Security	Data encrypted, access control
Data Integrity	No double-booking, atomic updates
Maintainability	Easy to fix, add features, well documented
Usability	Simple UI, accessible, responsive
Portability	Works on all browsers, deployable anywhere
Backup & Disaster Recovery	Daily backups, quick restore
Logging & Monitoring	Logs actions, monitors health, alerts admins
Legal & Compliance	GDPR compliant, privacy policies shown

Actors in Schedulink

Primary Actors

These are the main users who interact directly with the system to achieve their goals.

- **User (Customer)**
    - Books, reschedules, cancels, and views appointments.
    - Receives notifications and manages their profile.
- **Business / Service Provider**
    - Sets available time slots.
    - Views and manages appointments.
    - Interacts with their dashboard for daily operations.
- **Admin**
    - Creates, edits, and deactivates user accounts.
    - Manages businesses, appointments, and slots.
    - Accesses system-wide analytics and controls.

Secondary Actors

These actors support the system or interact indirectly, often triggered by the actions of primary actors.

- **Notification Service**
    - Sends emails/SMS for confirmations, reminders, and updates.
- **Database System**
    - Stores and retrieves appointments, users, slots, and logs.
- **Authentication Service**
    - Validates user credentials during login.

Offstage Actors

These are external entities or systems that influence or are influenced by the system but do not interact regularly.

- **System Administrator (IT/Support)**
    - Maintains the platform, monitors logs, handles backups and disaster recovery.
    - Performs maintenance and troubleshooting.
- **Regulatory Authorities**
    - Ensure system compliance with data protection laws (e.g., GDPR).
- **Third-Party Integrations (Optional)**
    - External calendar services (Google Calendar, Outlook) if/when integrated.
    - May sync appointments for users/businesses.

Schedulink - Casual Problem Statement – Use Cases

Use Case 1: Book an Appointment

Actor: User (Sarah)

Description: Sarah wants to book a dental checkup at BrightSmiles clinic.

Steps:

1. Sarah logs into Schedulink.
2. She browses available slots for BrightSmiles.
3. She picks Tuesday at 11:30 AM and books it.
4. She gets a confirmation notification.

Use Case 2: Reschedule Appointment

Actor: User (Sarah)

Description: Sarah needs to reschedule her dental appointment because a conflict came up.

Steps:

1. Sarah logs in and goes to her appointments.
2. She chooses her dental booking and clicks "Reschedule."
3. She picks a new time for Wednesday at 10:00 AM.
4. The system updates her appointment and sends her a notification.

4. The system updates her appointment and sends her a notification.

Use Case 3: Set Available Appointment Slots

Actor: Business (BrightSmiles Clinic Owner)

Description: The clinic wants to set up appointment slots for next week.

Steps:

1. The clinic owner logs into the business dashboard.
2. They select days and set available time ranges.
3. Schedulink generates appointment slots for those days.
4. Slots show up for users to book.

Use Case 4: Create New User Account

Actor: Admin (John)

Description: A new employee joins BrightSmiles and needs a Schedulink account.

Steps:

1. John logs into the admin dashboard.
2. He clicks “Create User” and fills in the new employee’s details.
3. Schedulink sends login credentials to the new user.

Use Case 5: Cancel Appointment

Actor: User (Sarah)

Description: Sarah decides to cancel her dental appointment.

Steps:

1. Sarah logs in and goes to her appointments.
2. She selects the booking and clicks "Cancel."
3. Schedulink updates the status and frees up the slot.
4. Both Sarah and the clinic get cancellation notifications.

Use Case 6: View Daily Schedule

Actor: Business (BrightSmiles Clinic Owner)

Description: The clinic wants to view today's schedule.

Steps:

1. The owner logs into the dashboard.
2. They select "Today's Schedule."
3. Schedulink shows all booked slots and client details for that day.

Use Case 7: Deactivate User Account

Actor: Admin (John)

Description: An employee leaves BrightSmiles and their account needs to be deactivated.

Steps:

1. John logs in and opens the user list.
2. He finds the employee and clicks "Deactivate."
3. Schedulink disables the account, blocking access.

Schedulink – Fully Dressed Use Cases

Use Case Name	Book an Appointment
Scenario Name	Sarah books a dental checkup at BrightSmiles
Level	User Goal
Triggering Event	User selects "Book Appointment" after logging in
Brief Description	A user browses available slots and books an appointment at a clinic.
Actors	User (Sarah), Business (BrightSmiles), System
Use Cases	Appointment Booking
Stakeholders	User: Wants easy booking and confirmation. Business: Wants efficient booking and no double-booking. Admin: Wants accurate system operation.
Pre-Condition	User is logged in; Business has available slots.
Post-Condition	Appointment is booked; Notifications sent to user and business.

<table>
<tr><td>Flow of Events</td><td>
<table>
<tr><th>Actor</th><th>Action</th><th>System</th><th>Response</th></tr>
<tr><td>Sarah</td><td>Logs in</td><td></td><td></td></tr>
<tr><td>Sarah</td><td>Opens booking page</td><td></td><td></td></tr>
<tr><td>Sarah</td><td>Selects clinic and slot</td><td></td><td></td></tr>
<tr><td>Sarah</td><td>Confirms booking</td><td>Receives request</td><td>Checks slot availability, books slot, sends confirmation</td></tr>
</table>
</td></tr>
<tr><td>Extensions</td><td>Slot becomes unavailable before confirmation: System prompts to select another slot.</td></tr>
<tr><td>Exceptional Scenarios</td><td>No slots available: System displays "No slots available."
Network failure: Booking fails, user prompted to retry.</td></tr>
<tr><td>Technology/Data Variations</td><td>Web/Mobile interface; Email/SMS notifications; Slot/user data in relational DB.</td></tr>
</table>

Use Case Name	Reschedule Appointment
Scenario Name	Sarah changes her dental appointment time
Level	User Goal
Triggering Event	User selects "Reschedule" on an existing appointment
Brief Description	A user changes the time of an existing appointment.
Actors	User (Sarah), Business, System
Use Cases	Appointment Management
Stakeholders	User: Wants to change timing easily. Business: Needs updated schedule. Admin: Wants accurate records.
Pre-Condition	Appointment exists; User is logged in.
Post-Condition	Appointment updated; Notifications sent.
Flow of Events	<table><tr><th>Actor</th><th>Action</th><th>System</th><th>Response</th></tr><tr><td>Sarah</td><td>Logs in</td><td></td><td></td></tr><tr><td>Sarah</td><td>Goes to appointments</td><td></td><td></td></tr><tr><td>Sarah</td><td>Selects appointment</td><td></td><td></td></tr><tr><td>Sarah</td><td>Picks new slot</td><td>Receives request</td><td>Checks slot availability, updates appointment, sends notification</td></tr></table>
Extensions	New slot unavailable: System prompts to choose another slot.

Extensions	New slot unavailable: System prompts to choose another slot.
Exceptional Scenarios	No alternative slots: System displays "No slots available." Appointment already past: Cannot be rescheduled.
Technology/Data Variations	Web/Mobile interface; Email/SMS notifications; Appointment data in relational DB.

Use Case Name	Set Available Appointment Slots
Scenario Name	BrightSmiles owner sets up slots for next week
Level	Business Process
Triggering Event	Business logs in and navigates to slot management
Brief Description	Business defines days and times for appointments.
Actors	Business Owner, System
Use Cases	Slot Management
Stakeholders	Business: Wants to manage availability. Users: Need up-to-date options. Admin: Wants real-time changes.
Pre-Condition	Business account exists; Owner is logged in.
Post-Condition	Slots are available for users to book.
Flow of Events	<table><tr><th>Actor</th><th>Action</th><th>System</th><th>Response</th></tr><tr><td>Owner</td><td>Logs in</td><td></td><td></td></tr><tr><td>Owner</td><td>Navigates to slot mgmt.</td><td></td><td></td></tr><tr><td>Owner</td><td>Sets days/times</td><td>Receives request</td><td>Generates slots, updates data, slots available to users</td></tr></table>
Extensions	Overlapping slots: System prompts for correction.
Exceptional Scenarios	Invalid time range: System displays error.
Technology/Data Variations	Web/Mobile interface; Slot data in relational DB.

Use Case Name	Create New User Account
Scenario Name	Admin adds a new employee to BrightSmiles
Level	Admin Process
Triggering Event	Admin selects "Create User"
Brief Description	Admin creates an account for a new staff member.
Actors	Admin (John), System

Use Cases	User Management
Stakeholders	Admin: Wants to add users. Business: Wants staff access. New Employee: Needs credentials.
Pre-Condition	Admin has dashboard access.
Post-Condition	User account created; Credentials sent.
Flow of Events	**Actor** / **Action** / **System** / **Response** John / Logs in / / John / Selects "Create User" / / John / Enters user details / Receives info / Creates account, sends credentials via email
Extensions	Duplicate email: System requests different email.
Exceptional Scenarios	Email delivery fails: Credentials not received.
Technology/Data Variations	Admin dashboard; Email system for credentials; User data in relational DB.

Use Case Name	**Cancel Appointment**
Scenario Name	Sarah cancels her scheduled dental appointment
Level	User Goal
Triggering Event	User selects "Cancel" on an appointment
Brief Description	Cancels an existing booking and frees up the slot.
Actors	User (Sarah), Business, System
Use Cases	Appointment Management
Stakeholders	User: Needs to cancel easily. Business: Wants notification. Admin: Wants accurate records.
Pre-Condition	Appointment exists; User is logged in.
Post-Condition	Appointment canceled; Notifications sent; Slot freed.
Flow of Events	**Actor** / **Action** / **System** / **Response** Sarah / Logs in / / Sarah / Goes to appointments / / Sarah / Selects appointment to cancel / Receives request / Cancels appointment, frees slot, sends notification

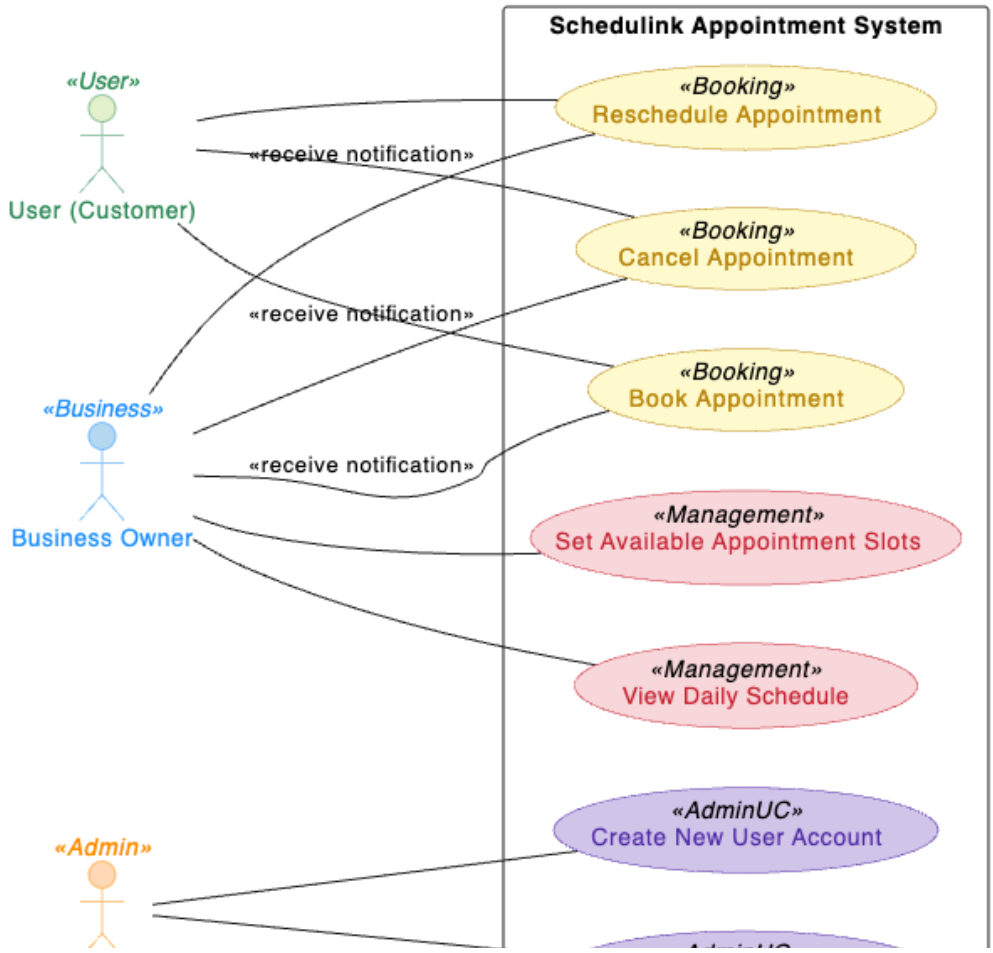
Extensions	Appointment past: Cannot be canceled.
Exceptional Scenarios	Network failure: Cancellation fails, user prompted to retry.
Technology/Data Variations	Web/Mobile interface; Email/SMS notifications; Appointment data in relational DB.

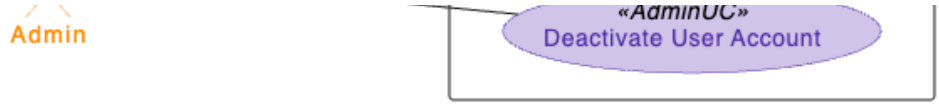
Use Case Name	**View Daily Schedule**
Scenario Name	BrightSmiles owner views today’s appointments
Level	Business Process
Triggering Event	Owner selects "Today’s Schedule" from dashboard
Brief Description	Shows all appointments for the current day.
Actors	Business Owner, System
Use Cases	Appointment Management
Stakeholders	Business: Needs daily view. Users: Expect appointments honored. Admin: Wants accurate reporting.
Pre-Condition	Owner is logged in.
Post-Condition	Daily schedule displayed.
Flow of Events	**Actor** / **Action** / **System** / **Response** Owner / Logs in / / Owner / Selects "Today’s Schedule" / Receives request / Displays list of appointments and client details
Extensions	No appointments: System displays "No appointments scheduled."
Exceptional Scenarios	Data sync error: Incomplete information shown.
Technology/Data Variations	Web dashboard; Appointment data in relational DB.

Use Case Name	**Deactivate User Account**
Scenario Name	Admin disables an employee’s access after they leave
Level	Admin Process
Triggering Event	Admin selects "Deactivate" on user list
Brief Description	Disables user’s access to the system.

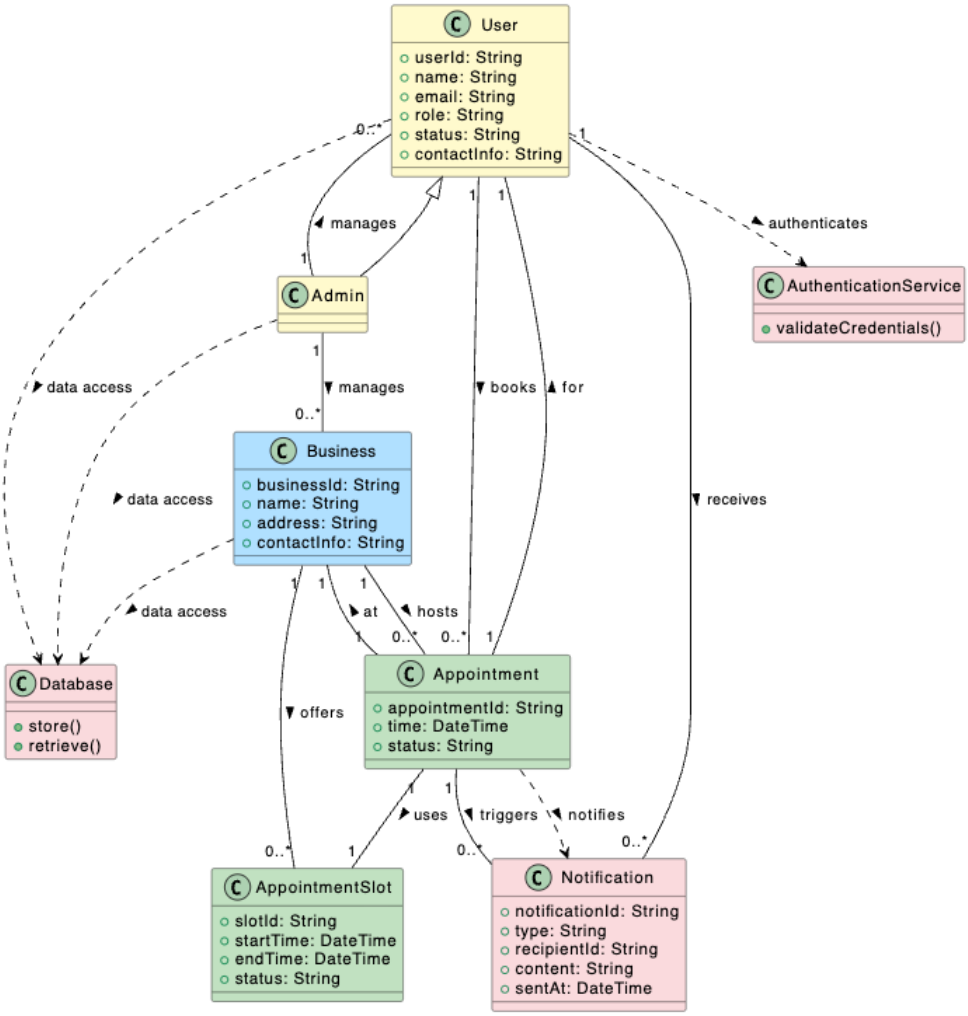
<table>
<tr><td>Actors</td><td>Admin (John), System</td></tr>
<tr><td>Use Cases</td><td>User Management</td></tr>
<tr><td>Stakeholders</td><td>Admin: Needs to remove access.
Business: Wants staff removed.
User: Account disabled.</td></tr>
<tr><td>Pre-Condition</td><td>User account exists; Admin is logged in.</td></tr>
<tr><td>Post-Condition</td><td>Account deactivated; access blocked.</td></tr>
<tr><td>Flow of Events</td><td>
<table>
<tr><th>Actor</th><th>Action</th><th>System</th><th>Response</th></tr>
<tr><td>John</td><td>Logs in</td><td></td><td></td></tr>
<tr><td>John</td><td>Opens user list</td><td></td><td></td></tr>
<tr><td>John</td><td>Selects employee to deactivate</td><td>Receives request</td><td>Deactivates account, blocks access, shows success message</td></tr>
</table>
</td></tr>
<tr><td>Extensions</td><td>Account already inactive: System displays "Already deactivated."</td></tr>
<tr><td>Exceptional Scenarios</td><td>Error updating database: Deactivation fails, admin prompted to retry.</td></tr>
<tr><td>Technology/Data Variations</td><td>Admin dashboard; User data in relational DB.</td></tr>
</table>

Schedulink - Use Case Diagram:

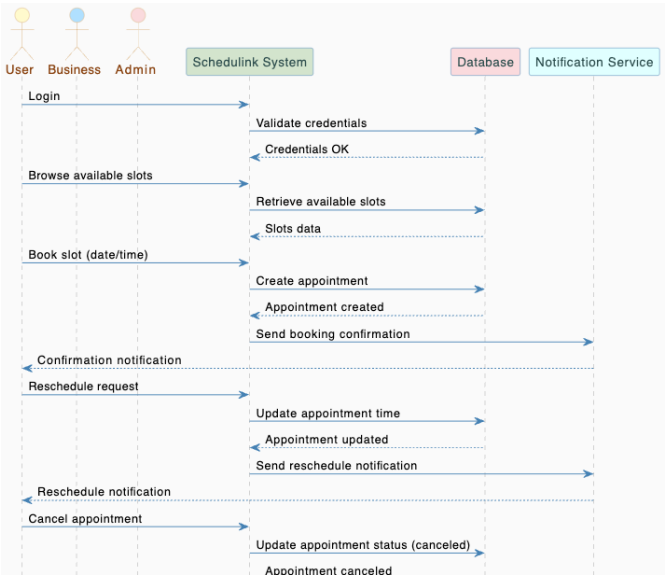


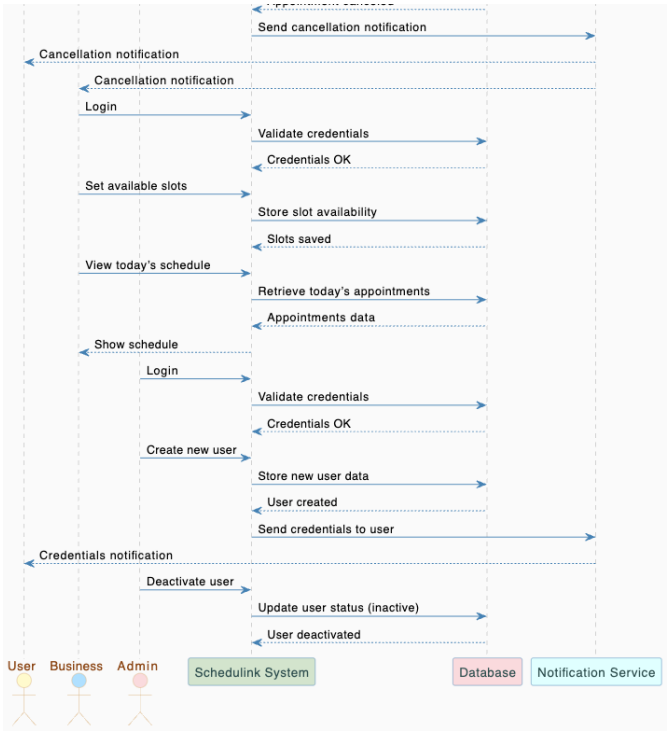


Schedulink - Domain Diagram

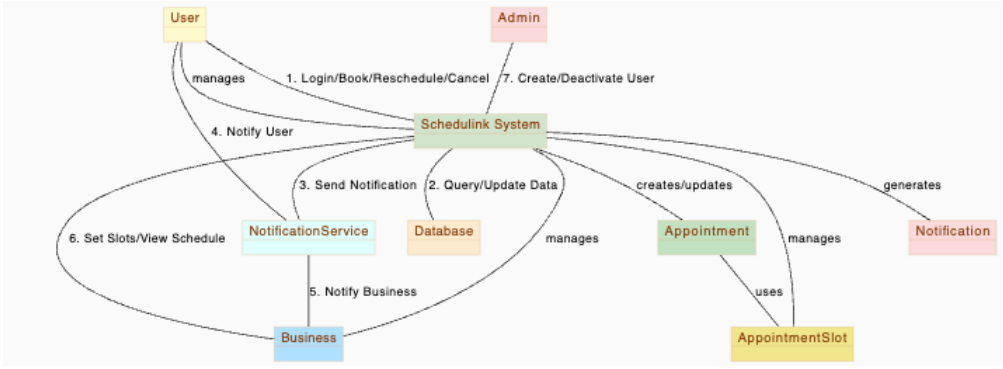


Schedulink - Sequence Diagram





Schedulink - Collaboration Diagram:



Schedulink Operation Contracts

UC1: Book Appointment

Field	Description
Operation	bookAppointment()
Cross References	Use Case: Book Appointment Sequence Diagram: Book Appointment
Preconditions	User is authenticated Selected slot is available
Postconditions	- Appointment instance a is created (instance creation) - a is associated with the user and slot (association formed) - Slot status is set to 'booked' (attribute modification) - User and business are notified (notification sent)

UC2: Reschedule Appointment

Field	Description
Operation	rescheduleAppointment()
Cross References	Use Case: Reschedule Appointment Sequence Diagram: Reschedule Appointment
Preconditions	User is authenticated User has an existing appointment New slot is available
Postconditions	- Appointment instance a is updated with new slot/time (attribute modification) - Slot associations are updated (association update) - Notifications are sent to affected parties (notification sent)

UC3: Cancel Appointment

Field	Description
Operation	cancelAppointment()
Cross References	Use Case: Cancel Appointment Sequence Diagram: Cancel Appointment
Preconditions	User is authenticated User has an existing appointment
Postconditions	- Appointment instance a is marked as cancelled (attribute modification) - Associated slot is released (attribute modification) - Notifications sent to user and business (notification sent)

UC4: Set Available Appointment Slots

Field	Description
Operation	setAvailableSlots()
Cross References	Use Case: Set Available Appointment Slots Sequence Diagram: Set Slots
Preconditions	Business user is authenticated
Postconditions	- New slot instances are created (instance creation) - Slots are associated with business (association formed) - Slots become available for booking (attribute modification)

UC5: View Daily Schedule

Field	Description
Operation	viewDailySchedule()
Cross References	Use Case: View Daily Schedule Sequence Diagram: View Schedule
Preconditions	Business user is authenticated
Postconditions	- Schedule data is retrieved (data retrieval) - Schedule is displayed to the business user (output shown)

UC6: Create New User Account

Field	Description
Operation	createUserAccount()
Cross References	Use Case: Create New User Account Sequence Diagram: Create User
Preconditions	Admin is authenticated User/email does not already exist
Postconditions	- User account instance u is created (instance creation) - Credentials are sent to the user (notification sent)

UC7: Deactivate User Account

Field	Description
Operation	deactivateUserAccount()
Cross References	Use Case: Deactivate User Account Sequence Diagram: Deactivate User
Preconditions	Admin is authenticated User account exists and is active
Postconditions	- User account is marked as inactive (attribute modification) - User access is revoked (effect enforcement)

Schedulink - Class Identification – Using Noun Phrase Analysis

Noun Phrase	Candidate Class	Domain/Core?
User	User	Yes
Business User	Business (User)	Yes
Admin	Admin (User)	Yes
Appointment	Appointment	Yes
Appointment Slot	AppointmentSlot	Yes
Schedule	Schedule	Yes
Account	Account	Yes
Notification	Notification	Yes
Notification Service	NotificationService	Supporting
Authentication Service	AuthenticationService	Supporting

Schedulink Class-Responsibility-Collaborator (CRC) Cards

User

Responsibilities	Collaborators
Register and manage personal info	Account, Appointment
Book, reschedule, and cancel appointments	Appointment, AppointmentSlot
View appointments and notifications	Notification

Business

Responsibilities	Collaborators
Manage business profile and availability	AppointmentSlot, Schedule
Set/update available slots	AppointmentSlot
View and manage appointments	Appointment, Schedule

Admin

Responsibilities	Collaborators
Create/deactivate user & business accounts	User, Account

Oversee system operations	User, Business

Appointment

Responsibilities	Collaborators
Store appointment details and status	User, Business, AppointmentSlot
Link user, slot, and business	AppointmentSlot
Notify parties of changes	Notification

AppointmentSlot

Responsibilities	Collaborators
Represent available or booked time slots	Business, Appointment
Update slot status	Appointment

Schedule

Responsibilities	Collaborators
Aggregate appointments for a period	Appointment, Business
Provide schedule view/reporting	Business

Account

Responsibilities	Collaborators
Store authentication and profile info	User, Admin, AuthenticationService
Manage account status	User, Admin

Notification

Responsibilities	Collaborators
Store notification data	User, Appointment
Deliver notifications to users	NotificationService

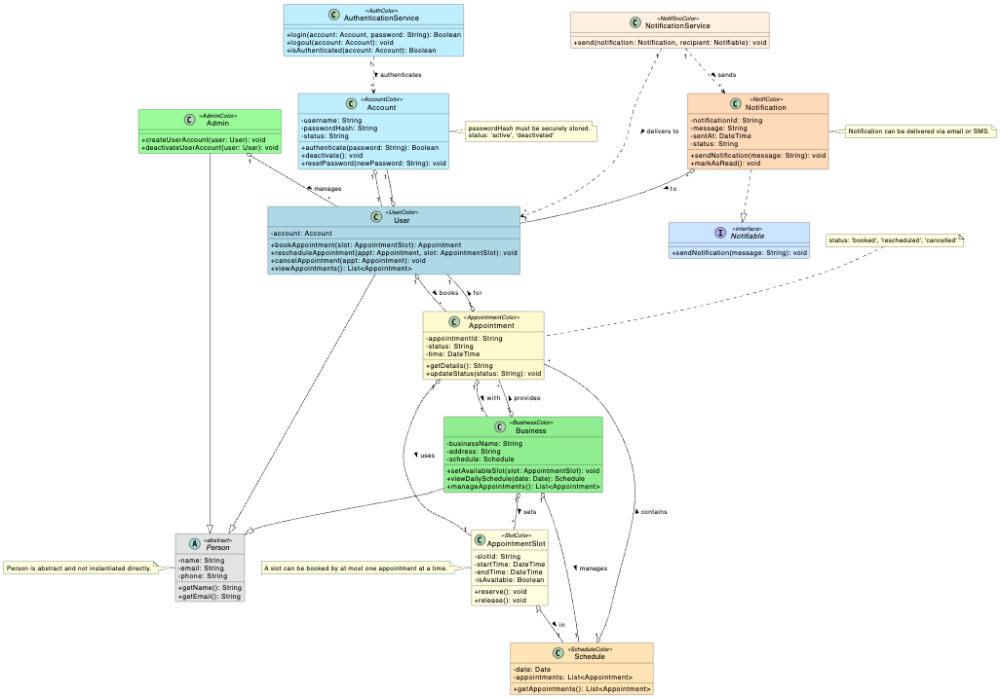
NotificationService

Responsibilities	Collaborators
Send notifications (email, SMS, etc.)	Notification, User, Appointment

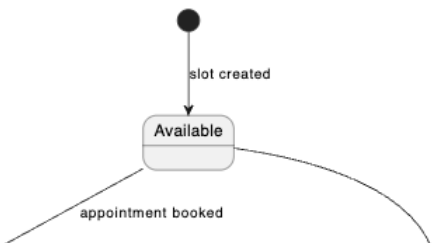
AuthenticationService

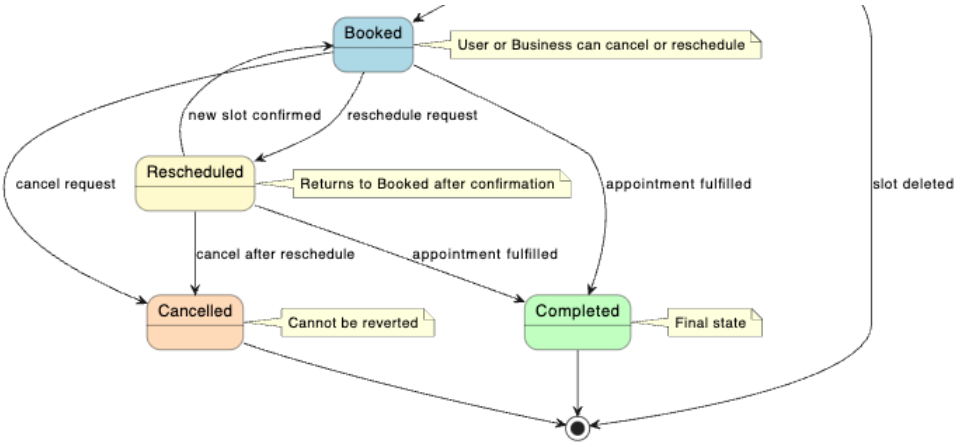
Responsibilities	Collaborators
Authenticate users, manage sessions	Account, User

Schedulink - Class Diagram



Schedulink - State Transition Diagram





Schedulink - Activity Diagram