

Contents

| | |
|------------------------------------|----------|
| Functional requirements: | 1 |
| User roles: | 1 |
| User management: | 2 |
| Rooms: | 2 |
| Bookings: | 3 |
| App: | 4 |
| Nice to have features: | 4 |
| Domain model: | 4 |
| Database model: | 4 |
| REST: | 4 |

Kravspecs:

Room Booking System, is a system designed to make it easy for students to find the correct class room, and to make it easy for teachers to book rooms.

Functional requirements:

User roles:

System admin:

- Can create admin users
- Requires login.

Admin:

- Can do everything a teacher can do.
- Can create new users.
 - assign either admin or teacher role to user.
- Can update users.
- Can delete users.
 - Can't delete system admin.
 - Can't delete the current user.
- Can disable users.
 - Can disable the system admin user.
 - Can't disable the current user.
 - This only disables the users login, and de authorize them.
- Can re enable users.
 - Can re enable the system admin user.
- Can create rooms.
- Can update rooms.
- Can delete rooms.
- Can create room groups.
- Can update room groups.
- Can delete room groups.

- Can update and delete all bookings.
- Requires login.

Teacher:

- Can do everything a student can do.
- Can update own user profile.
 - Change username.
 - Change password.
- Can create bookings.
- Can update bookings.
 - Only bookings that they have created.
- Can delete bookings.
 - Only bookings that they have created.
- Requires login.

Student (Guest):

- Can view bookings
- Doesn't need login.
- Can request room booking.

User management:

- Profile view:
 - Option for changing username.
 - Option for changing password.
- User management view:
 - Only visible to admins.
 - Option to create new users.
 - * users can have either admin, or teacher role.
 - Option to update user roles.
 - * Can't update current user role.
 - * Can't update system admin user.
 - Option to delete users.
 - * Can't delete current user.
 - * Can't delete system admin user.
 - Option to disable users.
 - * Can disable the system admin user.
 - * Can't disable the current user.
 - * This only disables the users login, and de authorize them.
 - Option to re enable users.
 - * Can re enable the system admin user.

Rooms:

- Room management view:
 - Contains a list of all rooms.

- Should have options for admin users to:
 - * Create rooms, must include the following:
 - Name.
 - Room id. This is used on the bookings.
 - Description.
 - Size.
 - * Update rooms.
 - * Delete rooms.
- Group management view:
 - Groups are only to sort rooms.
 - Contains a list of all room groups.
 - Should have options for admin users to:
 - * Create groups.
 - * Update groups.
 - * Delete groups.
 - * Add rooms to a group.

Bookings:

- Templates for default page design.
- Day view:
 - Shows all rooms and their bookings.
- Week view:
 - Shows all bookings in the current week for a specific room.
- Custom time duration view:
 - Shows all bookings in the time duration for a single room.
- Bookings should have custom booking durations.
 - The min. duration is 5 minutes.
 - The max. duration is to the end of the current day.
- Create booking view:
 - A booking must include:
 - * A title.
 - * Who created it.
 - * Which room it is for.
 - * start time.
 - A booking can include:
 - * recurring interval.
 - * A description.
 - * end time, if it's not provided then it goes to the end of the current day.
- Details view:
 - When a user clicks on a booking, it opens the details view.
 - Contains all info that have been entered during creation.
 - If the teacher who created it, or an admin opens the details view, then there should be an edit icon to allow updating the booking.

App:

- Desktop app using Electronjs.
 - Requires authenticable user.
 - Designed for teachers to make it easier to manage bookings.

Nice to have features:

- Websocket communication, so that when a change is made, then all devices are notified about the change and can refresh

Domain model:

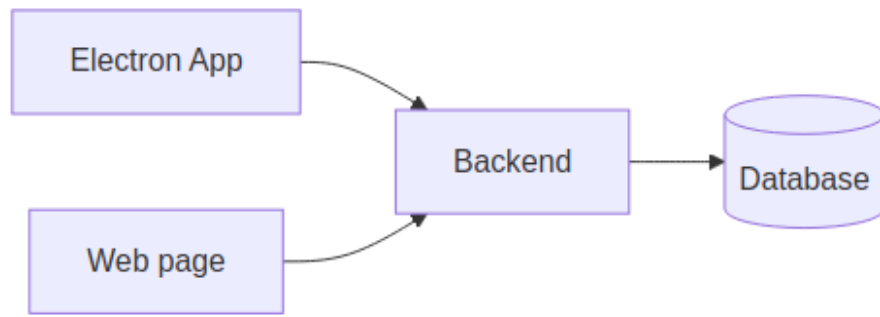


Figure 1: diagram

Database model:

REST:

Tids system.

mobil side. desktop side.

når man trykker på en booking, skal man kunne se mere info omkring booking.
når man trykker på et lokale så viser den kun bookings for det lokale. (når man åbner detaljeret info for en booking, så vil den underviser som har oprettet booking, samt admins kunne se et pencil ikon, som man kan trykke for at redigere booking.)

Alle brugere:

- liste over ledige lokaler
- liste over alle lokaler
- grupperinger af lokaler
 - Det skal være muligt at søge/filterer baseret på grupperinger.

(Underviser bruger: - kan booke lokaler - kan redigerer egne bookings - kan slette egne bookings)

(Admin bruger: - Det skal være muligt at oprette grupperinger af lokaler. - Kan redigerer alle bookings (CRUD) - Kan oprette lokaler - Kan slette lokaler - Kan tilføje nye bruger - Kan slette brugerer.)

(bruger grupper: - elever (Kræver ikke login) - underviser (Kræver login) - admin (kræver login))

(Backend: - auth using jwt - api for crud operations: - user - rooms - room-groups - bookings - database

Frontend: - User management - profile - admin

- Room management
 - CRUD rooms
 - CRUD groups)

(- Opret booking side.)

(- Default side - Day page with all rooms + bookings - Week page for single room + bookings - Custom time duration page for single room + bookings.)

(Andet: - Electron desktop/mobile app - Websocket communication, so that when a change is made, then all devices are notified about the change and can refresh)