A short description of a beauty salon website. Developed as a Single Page Application (SPA), with React.js for front-end and Node.js + Hapi.js for backend technologies.

1. Main user roles

1.1 Anonymous user -> can browse information and can register

* 1. Registered user -> can browse information and can make an appointment
  2. Beautician -> can create procedures and edit those already created

1.4 Admin -> can manage all users and procedures

1. Main use cases/scenarios

2.1 All users can browse information about the procedures

2.2 Anonymous users can register and admin can register a user with all roles

2.3 Registered users can edit their info and admin can edit other users` info and assign them roles

2.4 Admin can browse users and can choose a given user by some filter and manage the user

2.5 Registered user can pick a procedure and make an appointment

2.6 Admin and beautician can add/edit/delete a procedure

1. Main views (SPA frontend)

3.1 Home -> presents information about the beauty salon and its importance for every human being. URI -> “ / “

3.2 Login or Register -> users can login with their existing account or register a new one.

URI -> “ /login/register “

3.3 Procedures -> presents all procedure available at the salon. URI -> “ /procedures “

3.4 Current procedure -> Displays info for a chosen procedure with its price, appointment options and for beautician and admin – editing and deleting the procedure.

URI -> “ /procedure/{procedureId}

3.5 User info -> presents viewing and editing personal information. URI -> “ /personal “

3.6 Users -> page for beautician and admin to manage registered users. URI -> “ /users “

3.7 About -> information about the beauty salon

1. Api resources / (Node backend)

4.1 Users -> GET all user data and POST data for a new user. Available for admin only.

URI -> “ /api/users ”

* 1. User -> GET, PUT data for user with specified id. URI -> “ /api/users/{userId} ”
  2. Procedures -> GET all procedures, POST a new procedure. URI -> “ /api/procedures “
  3. Procedure -> GET, PUT, DELETE a procedure with specified id.

URI -> “ /api/procedures/{procedureId} “