|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | Fullstack Application Development with Node.js + Express.js + React.js - 2020 | |
|  | | | |
| Project author | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1 | VeeS | | face-to-face |

|  |  |
| --- | --- |
| Project name | Babysitting System (BSS) (HappySitting) |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| The Babysitting System aims to provide a platform where Parents can easily find a suitable babysitter for their children. In addition to that it allows users to register, and administrators to manage them. The system will be developed as a *Single Page Application (SPA)* using ***React.js*** as front-end, and ***Node.js + express*** as backend technologies. Each view will have a distinct URL, and the routing between pages will be done client side using ***React Router***. The backend will be implemented as a ***REST/JSON API*** using JSON data serialization. The main user roles (actors in UML) are:  • *Anonymous User* – can only view a list of current babysitting task posts and is prompted to register if he tries to open any of them.  •Parent (extends *Registered User*) – can make new babysitting task posts, review and edit all of his currently active babysitting task posts, view and accept offers from Babysitters, rate Babysitters(1-5 stars).  •Babysitter (extends *Registered User*) – can view all available babysitting task posts, make offers to Parents, view own profile – tasks completion history and rating (received from Parents).  • *Administrator* (extends *Registered User*) – can manage (create, edit user data and delete) all *Registered Users*, as well as Babysitting Task Posts. |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Browse Babysitting Task Posts** | Users with role – Babysitter and *Administrator* can browse the Babysitting Task Posts. | Babysitter,  *Administrator* |
| * 1. **Register** | *Anonymous User* can register in the system by providing at least a valid e-mail address, first and last name, and choosing password and role (either Parent or Babysitter).If the Parent role is selected, age, gender, medical conditions and allergies of the Child/Children are required. Optional data is – avatar(picture of themselves), brief information about the Babysitter(experience) or child(behavior, favorite things etc.). | *Anonymous User* |
| * 1. **Change User Data** | *Registered User* can view and edit her personal *User Data.*  *Administrator* can view and edit *User Data* of all *Users* and assign them *Roles*: Parent*, Babysitter*, or *Administrator*. | *Registered User, Administrator* |
| * 1. **Manage Users** | *Administrator* can browse and filter users based on different criteria: first and last name, email, Role.  *Administrator* can choose a *User* to manage, and can manage the chosen User - edit (using Change User Data UC) or delete.  *Administrator* can create a new user using *Register UC*. | *Administrator* |
| * 1. **Manage Babysitting Task Posts** | *Administrator* can browse Babysitting Task Posts and delete them if necessary. | *Administrator* |
| * 1. **Create Babysitting Task Post** | *Parent can create new Babysitting Task Post by selecting Child (has 1 or more children, created on registration), and specifying time slot for the Babysitting Task.*  *Administrator* can browse *Babysitting Task Posts* of all *Parents*, edit and delete them. | *Parent, Administrator* |
| * 1. **Make Offer to a Babysitting Task Post and view own offers** | *Babysitter can make Offers to Babysitting Task Posts by selecting the desired Babysitting Task Post and specify the rate(BGN) for which he/she is willing to complete the Task. Babysitter can view own unaccepted offers and accepted offers and can edit and delete unaccepted offers.*  *Administrator* can browse all Offers of all *Babysitters*, edit and delete them. | *Babysitter, Administrator* |
| * 1. **View Offers** | *Parent can view all active offers on his/her active Babysitting Task Posts and accept one offer per Task, once an offer is accepted the Babysitting Task Post can no longer be viewed by other Babysitters except the one which made the offer(can be viewed in a section – Accepted Tasks).*  *Administrator* can browse accepted offers of all *Parents*, edit and delete them. | *Parent, Administrator* |
| * 1. **Resolve Babysitting Task Post** | *Parent* or *Administrator* can resolve a Babysitting Task Post by making it status to: resolved. Parent gives rating to Babysitter after every resolved Task. | *Parent, Administrator* |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (SPA Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Home** | Presents the introductory information for the purpose of the system as well as detailed instructions how to start using it. Prominently offers ability to register. | / |
| * 1. **Babysitting Task Posts** | Presents all unaccepted Babysitting Task Posts, can view more details about them and make offers (for Babysitters and Administrators only). | */babysitting-task-posts \*\*\*(/offers)* |
| * 1. **Create Babysitting Task Post** | Create Babysitting Task Post by specifying child/children and time slot for the task (chosen from a calendar). (For Parents and Administrators only). | */create-task* |
| * 1. **View Accepted Babysitting Task Posts** | View information about the accepted task, make comments, ask questions. (for Parents, Babysitters and Administrators). | */accepted-tasks* |
| * 1. **Resolve Babysitting Task** | Mark task as resolved, rating is required (1-5 stars) and optional text. (for Parents and Administrators only). | */resolve-task* |
| * 1. **Login** | Presents a view allowing the users to login. | */login* |
| * 1. **User Data** | Presents ability to view and edit personal *User Data*, as well as deregister from the system. | */personal* |
| * 1. **Users** | Presents ability to manage (CRUD) Users and their User Data (available for *Administrators* only). | */users* |
| * 1. **About** | Presents information about theproject and his owner. | */about* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (Node.js Backend) (needs more work)\*\*\* | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Users** | GET *User Data* for all users, and POST new *User Data* (Id is auto-filled by BSSand modified entity is returned as result from POST request). Available only for *Administrators*. | */api/users* |
| * 1. **User** | GET, PUT, DELETE *User Data* for *User* with specified *userId* | */api/users/{userId}* |
| * 1. **Login** | POST *User Credentials* (e-mail address and password) | */api/login* |
| * 1. **Logout** | POST a logout request for ending the active session with BSS. | */api/logout* |
| * 1. **Babysitting Task Post** | GET, PUT, DELETE Babysitting Task Post *Data*  with specified task*Id*. | */api/tasks/{tasktId}* |