

Assignment 1	Project Summary
Course	Fullstack Application Development with Node.js + Express.js + React.js - 2017

Project author			
№	First name, last name	E-mail	Face-to-face/ online
1	Ivan Cvetanov	mladenovv.ivan@gmail.com	face-to-face
2	Branimir Kirilov	kirilov.branimir@gmail.com	

Project name	Sportsemblr.2
--------------	---------------

1. Short project description (Business needs and system features)
<p>Sportsemblr.2 helps people who enjoy sports to find teammates and opponents to play with/against. The number one cause of people not exercising enough is the lack of community around them. With Sportsemblr.2 everyone who wishes to play sports just has to add a sport event with the time and location of the gathering for other members to see and join. Of course after a game, the participants can give a review of each other for a better gathering next time. The system will be developed as a <i>Single Page Application (SPA)</i> using React.js as front-end, and Node.js + express as backend technologies.</p> <p>The main user roles are:</p> <ul style="list-style-type: none"> • <i>Anonymous User</i> – can only view the information about sport events • Registered user - can send requests to join events, create his own events and review other people and places. They can also send messages in the event chat • <i>Event organiser</i> – user that creates and manages events • Participant – user, participating in given event

- Place owner – *can add and manage his places, and also create and manage events on these places*
- Administrator (extends User) – *can manage (create, edit user data and delete) all Registered Users, as well as Events and Reviews.*

2. Main Use Cases / Scenarios		
Use case name	Brief Descriptions	Actors Involved
2.1. Browse sport events	The <i>User</i> can browse the all the events which are divided in different categories.	All users
2.2. Register	<i>Anonymous User</i> can register in the system by providing a valid e-mail address, first and last name, and choosing password.	<i>Anonymous User, Administrator</i>
2.3. Change User Data	<i>User</i> can view and edit her personal <i>User Data</i> . <i>Administrator</i> can view and edit <i>User Data</i> of all <i>Users</i> .	<i>Registered User, Administrator</i>
2.4. Manage Users	<i>Administrator</i> can browse and filter users based on different criteria: first and last name, email, Role. <i>Administrator</i> can choose a <i>User</i> to manage, and can manage the chosen <i>User</i> - edit (using Change User Data UC) or delete.	<i>Administrator</i>
2.5. Manage Events	<i>User</i> can manage his own events as well as view people that have joined the event and approve requests <i>Administrator</i> can browse and manage all events	Registered Users (owners of event), Administrator
2.6. Add Event	<i>Owner</i> or <i>Administrator</i> can create events.	<i>Registered User, Administrator</i>

2.7. Request to join	User sends a request to join an event	Registered User
2.8. Approve/Deny request	Event owner approves or denies requests	Registered Users (owners of event)
2.9. Leave review	Event participants leave review after event is finished	Registered User (that took part in the event)
2.10. Edit Event	The organizer of the event can edit its contents	Registered Users (owners of event), Administrators
2.11 Room Messages	Every room has direct messages before the event starts and it is persistent several minutes after the events end.	Registered Users (which joined an event)
2.12 Manage sports	Administrator can CRUD different sports	Administrator
2.13 Manage terrain (sport place)	Every owner can edit the places he posted	Registered Users (owners of event)

3. Main Views (SPA Frontend)		
View name	Brief Descriptions	URI
3.1. 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.	/
3.2. Events	Presents all the events available and the functionality to filter / order them by several parameters	/events
3.3. User Data	Presents ability to view and edit personal <i>User Data</i> , and also view all events that are hosted by the user.	/profile

3.4. Users	Presents ability to manage (CRUD) Users, their User Data and all the events (available for Administrators only, as described in UCs).	<i>/users</i>
3.5. About	Presents information about the system.	<i>/about</i>
3.6. Event	Presents the information for a single event. Offers the functionality to join it or go to the other user profile.	<i>/event?id={id}</i>
3.7. Places	Shows all the places that exist with a picture and a short brief description	<i>/places</i>
3.8. Place	Presents the information for a certain place, including rating of this place and reviews	<i>/place?id={id}</i>
3.9 Place reviews	List of all reviews for given place.	<i>/place/:id/reviews</i>

4. API Resources (Node.js Backend)		
View name	Brief Descriptions	URI
4.1. Users	GET <i>User Data</i> for all users, and POST new <i>User Data</i> . Available only for <i>Administrators</i> .	<i>/api/users</i>
4.2. User	GET, PUT, DELETE <i>User Data</i> for <i>User</i> with specified <i>userId</i> .	<i>/api/users/{userId}</i>
4.3. Events	GET users (according to <i>User's Role</i> and identity) and POST new <i>Events</i>	<i>/api/events</i>
4.4. Event	GET, PUT, DELETE <i>Event Data</i> for a specific event	<i>/api/events/{id}</i>

4.5. Event join / leave request	POST for <i>Event</i> with specified <i>eventId</i> that sends a notification to the host of the event.	<i>/api/events/{testId}/join</i>
4.6. User review	GET, POST, DELETE User Review	<i>/api/users/{userId}/reviews</i>
4.7. Place	GET, POST, DELETE Place	<i>/api/places/{placeId}</i>
4.8. Places	GET	<i>/api/places</i>
4.9 Event Chat Room	POST	<i>/api/event/{id}/talk</i>
4.10 Place reviews	GET	<i>/api/places/{placeId}/reviews</i>