

Assignment 1	Project Summary
Course	Fullstack Application Development with Node.js + Express.js + React.js - 2019

Project author		
No	Pseudonym	Face-to-face/ online
1	krislander	face-to-face

Project name	Vacation spot finder
--------------	----------------------

### 1. Short project description (Business needs and system features)

If summer is coming soon and you are still wondering where to take your summer break the Vacation spot finder is the application for you. By filling up a short survey and adjusting a few filters the app will suggest a series of vacation spots based on the activities, budget and other features you want to include. The app will include a login system, which allows every user to have a profile with set requirements which will determine the perfect destinations for you and your friends. There will be a contact us page, where users can ask questions. 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.

- *Regular User* (extends *Registered User*) – can modify his settings and filters in the profile view and search for destinations that may be suitable for him. Then he can click on the shown results and find more about the hotels, location, local landmarks and average budget needed
- *Administrator* (extends *Registered User*) – can add/modify destinations and filters for better accuracy of the recommended places to visit

### 1. Main Use Cases / Scenarios

Use case name	Brief Descriptions	Actors Involved
1. Browse information and try sample tests	The <i>User</i> can browse the implemented views (Home, Top Destinations, Profile,...), and can choose to change search settings and inspect the recommended destinations.	All users
2. Register	<i>Regular User</i> can register in the system by providing a valid e-mail address, first and last	<i>RegisterUser</i> , <i>Administrator</i>

	name, current address, and choose a password.  <i>Administrator can register other users too.</i>	
<b>3. Change User Data</b>	<i>Registered User</i> can view and edit her personal <i>User Data</i>  <i>Administrators can edit user</i>	<i>Registered User, Administrator</i>
<b>4. Manage Users</b>	<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>
<b>5. Add/Edit Destination</b>	<i>Administrator</i> can choose to edit every aspect of a destination, as well as add or delete one	<i>Administrator</i>
<b>6. Add/Edit Filters</b>	<i>Administrator</i> can choose to edit every aspect of the search filters such as budget, type of vacation, required features, as well as add or delete filters	<i>Administrator</i>
<b>7. Send Request</b>	<i>In the Profile page there will be another Contact us form where users can submit questions and suggestions directly to the administrator of the page</i>	<i>Registered User</i>

1. Main Views (SPA Frontend)		
View name	Brief Descriptions	URI
<b>1. Search Destinations</b>	Presents the main view as well as detailed instructions how to start using it. Prominently offers ability to register.	/vacation-spot-finder
<b>2. Register</b>	Presents a view allowing the <i>Anonymous Users</i> to register.	/register
<b>3. Login</b>	Presents a view allowing the users to login.	/login
<b>4. Profile</b>	Presents ability to view and edit personal <i>data and set filters and requirements for the search engine of the app.</i>	/profile{id}
<b>5. Top Destinations</b>	Presents a view with the top destinations liked and rated by other users	/top-destinations
<b>6. Users</b>	Presents ability to manage (CRUD) Users and their User Data (available for <i>Administrators</i> only, as described in).	/users

<b>7. Add Destination</b>	Presents a functionality to manage (CRUD) add/delete/edit a destination to the(Administrator only)	<i>/add-destination</i>
<b>8. My Travel List</b>	Shows a page of all the selected destinations by the user	<i>/travel-list{id}</i>
<b>9. Contact</b>	A form in the profile page to send information to the administrator by normal users	<i>/contactus</i>
<b>10. Destination View</b>	Shows a page which contains the information about the vacation spot the user has currently selected. It has a destination Id	<i>/destination/{id}</i>

1. API Resources (Node.js Backend)		
View name	Brief Descriptions	URI
<b>1. Users</b>	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>
<b>1. User</b>	GET, PUT, DELETE <i>User Data</i> for <i>User</i> with specified <i>Id</i> .	<i>/api/users/{Id}</i>
<b>1. Login</b>	POST <i>User Credentials</i> (e-mail address and password)	<i>/api/login</i>
<b>1. Logout</b>	POST a logout request for ending the active session	<i>/api/logout</i>
<b>1. Destinations</b>	GET get a list of all destinations	<i>/api/destinations</i>
<b>1. Destination</b>	GET, PUT, DELETE <i>destination with specified Id</i>	<i>/api/destinations/{id}</i>
<b>1. User information</b>	GET, PUT, DELETE user information with specified id	<i>/api/user/{id}</i>
<b>1. Travel list</b>	GET,PUT, POST, DELETE from user's travel list with a specified id	<i>/api/travellist/{id}</i>
<b>1. Request to Admin</b>	POST message by user for admin	<i>/api/contactadmin</i>