

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	Петър Иванов	pepincho95@gmail.com	face-to-face
2	Георги Киряков	george.kiryakov@gmail.com	face-to-face

Project name	NightLux
--------------	----------

1. Short project description (Business needs and system features)
<p>NightLux е приложение, което ще помага на хората да намират обекти и услуги, работещи в късните часове на денонощието. Ще има обекти като аптеки, магазинчета, супермаркети, заведения за хранене и услуги (например ключар).</p> <p>Приложението ще бъде разработено като SPA, използвайки React.js за front-end и Node.js + Express.js за back-end.</p> <p>Основните роли ще са:</p> <ul style="list-style-type: none"> - Анонимен потребител, който може да търси обекти и услуги, налични в приложението - Модератор - може да добавя / променя / изтрива обекти в приложението - Администратор - е модератор, който може да изтрива и добавя модератори

2. Main Use Cases / Scenarios		
Use case name	Brief Descriptions	Actors Involved
2.1. Търсене на обекти	Потребителят може да търси обекти и услуги спрямо категория и локация.	<i>Всички потребители</i>
2.2. Добавяне на обект / услуга	Потребителят създава в базата данни обект / услуга като предостави категория, име, описание и локация за обекта / услугата.	<i>Модератор , Администратор</i>
2.3. Заявка за добавяне / промяна на обект / услуга	Потребителят може да направи заявка за добавяне / промяна на обект / услуга в приложението.	<i>Всички потребители</i>
2.4. Добавяне / изтриване на модератор	Администраторът добавя или изтрива даден модератор, т.е регистрира даден потребител с права на модератор.	<i>Администратор</i>
2.5 Оценяване на обект	Потребителя оценва обект / услуга по скалата от 1 до 5.	<i>Всички потребители</i>
2.6 Коментари		

3. Main Views (SPA Frontend)		
View name	Brief Descriptions	URI
3.1. Home	Главната страница на приложението, където ще има някои от обектите и услугите, категории и поле за търсене.	/
3.2. List Objects	Тук ще бъдат "листнати" всички обекти от дадена категория.	/tags/:id

3.2. Object Data	Страница, която предоставя различна информация на даден обект (страница на обекта)	<i>/services/:id</i>
3.3. New Request	Тук потребителите ще имат възможност да попълват форма и изпращат заявка за добавяне/променяне на нов обект или услуга	<i>/services/new</i>
3.4. Admin panel	Страница достъпна само за администраторите и модераторите. Ще могат да се виждат текущите заявки.	<i>/admin/requests</i>
3.5. Objects	Страница достъпна само за администраторите и модераторите. ще бъдат листнати всички обекти в системата	<i>/admin/objects</i>
3.5 Edit Object Data	Страница достъпна само за администраторите и модераторите. Ще се вижда информацията за конкретен обект и ще може да бъде променяна и форматирана.	<i>/admin/objects/:objectName</i>
3.6 Request review	Страница достъпна само за администраторите и модераторите. Тук ще може да се прегледа дадена заявка и съответно да се потвърди или отхвърли.	<i>/admin/requests/:requestId</i>
3.7 Users	Страница достъпна само за администраторите. Ще бъдат листнати всички регистрирани потребители, т.е администратори и модератори.	<i>/admin/users</i>
3.8 User Data	Страница достъпна само за администраторите. Тук ще се представя информацията за даден потребител от системата.	<i>/admin/users/:userId</i>
3.9 New moderator	Страница достъпна само за администраторите. Тук те ще могат да регистрират нов модератор.	<i>/admin/users/new</i>

4. API Resources (Node.js Backend)		
View name	Brief Descriptions	URI
4.1. Query Services	services(query: String, geoQuery: String, offset: Int, limit: Int): [Service]	
4.2. Service	service(id: ID!) createService(input: ServiceInput): Service deleteService(id: ID!): Service	
4.2. User	user(id: ID!): User	
4.3. Users	users()	
4.4. Current User	me(): User	