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

Project author		
№	Pseudonym	Face-to-face/ online
1	Dimitar Ganichev	face-to-face
2	Hristo Raydovski	face-to-face
3	Mario Petrov	face-to-face

Project name	Project management tool

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

Project management tool allows users to manage the projects managing its tasks, resources and budgets. The system will be developed as a *Single Page Application (SPA)* using **React.js** as front-end, and **Node.js + express** as backend technologies.

The main user roles (actors in UML) are:

• *Anonymous User* – Can see information about the system and register in order to use it

• *Registered user* - Can create own budgets and modify budgets he is invited to.

2. Main Use Cases / Scenarios

Use case name	Brief Descriptions	Actors Involved
2.1. Login	As an anonymous user I want to log into the system so that I can use it.	*Anonymous User*
2.2. Register	As an anonymous user I want to register in the system so that I can use it.	*Anonymous User*
2.3. Manage projects	*As a registered user I want to manage the projects I created or I was invited to.*	*Registered User*
2.4. Manage project's users	*As a registered user I want to be able to manage the users who are invited to a project I created or I was invited to.*	*Registered User*
2.5. Manage projects's budgets	*As a registered user I want to be able to manage the budgets which are part of a project I created or I was invited to.*	*Registered User*
2.6. Manage projects's resources	*As a registered user I want to be able to manage the resources which are part of a project I created or I was invited to.*	*Registered User*
2.7. Manage projects's tasks	*As a registered user I want to be able to manage the tasks which are part of a project I created or I was invited to.*	*Registered User*

3. Main Views (SPA Frontend)

View name	Brief Descriptions	URI
3.1. Home	Presents basic information about the system. Presents ability to register and log into the system.	/
3.2. Projects	Presents ability to view and edit information about the projects the user owns or is invited to.	*/projects*

3.3. **Resources**	Presents ability to manage (CRUD) Resources of a project.	*/projects/{projectId}/resources*
3.4. **Users**	Presents ability to manage (CRUD) Users invited to a project.	*/projects/{projectId}/users*
3.5. **Tasks**	Presents ability to manage (CRUD) Tasks of a project	*/projects/{projectId}/tasks*
3.6 **Budgets**	Presents ability to manage (CRUD) Budgets of a project	*/projects/{projectId}/budgets*

4. API Resources (Node.js Backend)		
Resource name	**Brief Descriptions**	**URI**
4.1. **Login**	POST User Data(email and password) and returns JWT token	*/api/authenticate*
4.2. **Register**	POST User Data(email, password and confirmed password) and returns JWT token	*/api/users*
4.3. **Projects**	GET user's projects POST project data and creates new project	*/api/projects*
4.4. **Project**	GET given project and returns project data. PUT project data and modifies project.	*/api/projects/{projectId}*
4.5. **Project users**	GET given project's users.	*/api/projects/{projectId}/users*
4.6. **Project user**	DELETE given user from a given project. POST given user to a given project.	*/api/projects/{projectId}/users/{userId}*
4.7. **Project budgets**	GET given project's budgets.	*/api/projects/{projectId}/budgets*

	POST given budget to a given project.	
4.8. Project budget	PUT budget data and мodifies the budget. DELETE given budget from a given project	*/api/projects/{projectId}/budgets/{budgetId}*
4.9 Linked Project Resources to Project Budget	GET resources linked to given budget	*/api/projects/{projectId}/budgets/{budgetId}/resources/*
4.10 Link Project resource to Project budget	POST given resource to given budget DELETE given resource from a given budget	*/api/projects/{projectId}/budgets/{budgetId}/resources/{resourceId}*
4.11 Linked Project Resources to Project Task	GET resources linked to given task	*/api/projects/{projectId}/budgets/{budgetId}/tasks/*
4.12. Link Project resource to Project Task	POST given resource to given task DELETE given resource from a given task	*/api/projects/{projectId}/budgets/{budgetId}/tasks/{taskId}*
4.13. Project resources	GET given project's resources. POST given resource to a given project.	*/api/projects/{projectId}/resources/*
4.14. Project resource	DELETE given resource from a given project PUT resource data and modifies the resource	*/api/projects/{projectId}/resources/{resourceId}*
4.15. Project tasks	GET given project's tasks.	*/api/projects/{projectId}/tasks*

	POST given budget to a given task.	
4.16. Project task	DELETE given task from a given project PUT task data and modifies the task	*/api/projects/{projectId}/tasks/{taskId}*
4.17. Linked Project Budgets to Project Resource	GET budges linked to given resource	*/api/projects/{projectId}/resources/{resourceId}/budgets/*
4.18 Link Project Budget to Project Resource	POST given budget to given resource DELETE given budget from a given resource	*/api/projects/{projectId}/resources/{resourceId}/budgets/{budgetId}*
4.19. Linked Project Budgets to Project Task	GET budges linked to given task	*/api/projects/{projectId}/tasks/{taskId}/budgets/*
4.20. Link Project Budget to Project Task	POST given budget to given task DELETE given budget from a given task	*/api/projects/{projectId}/tasks/{taskId}/budgets/{budgetId}*
4.21. Linked Project Tasks to Project Resource	GET tasks linked to given resource	*/api/projects/{projectId}/resources/{resourceId}/tasks/*
4.22 Link Project Task to Project Resource	POST given task to given resource DELETE given task from a given resource	*/api/projects/{projectId}/resources/{resourceId}/tasks/{taskId}*
4.23. Linked Project Tasks to Project	GET tasks linked to given budget	*/api/projects/{projectId}/budges/{budgetId}/tasks/*

Budget		
4.24. Link Project Task to Project Budget	POST given task to given budget DELETE given task from a given budget	*/api/projects/{projectId}/resources/{resourceId}/budges/{budgetId}*