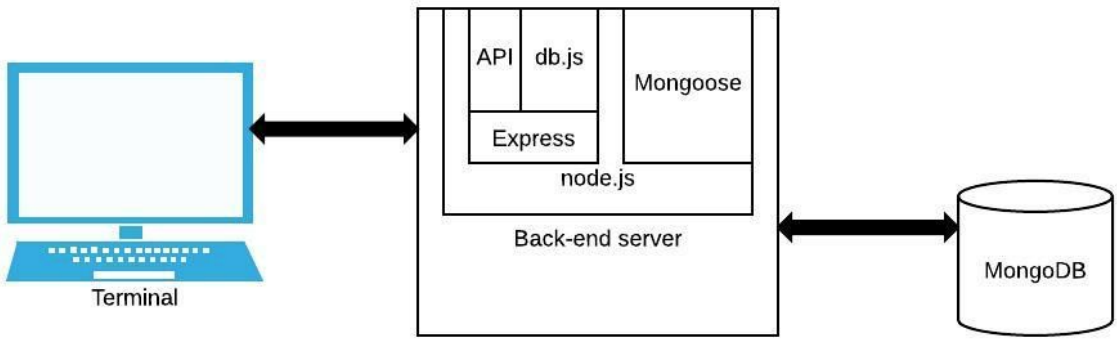
Service Layer Design

For

Project Daily Report

Student: Parisa Nikoukar

ProjectDailyReport app will be developed by Express.js framework and MongoDB as the database, the following image gives the whole idea about this application.



One of the most popular use-cases for Node.js is to write RESTful APIs using it.

RESTful API is the main focus for describing service endpoints in this application. The following HTTP verbs will be used for RESTful services for this context:

- POST: This method is used to create new reports in the app
- GET: This method is used to retrieve the list of reports based on their reporter or project name

Identify all service endpoints required for your MVP includes HTTP verbs, query parameters, path parameters, messages structures, and error codes

1.Endpoint: http://www.projectdailyreport.com/createreport

HTTP verb: POST => By using this HTTP verb a new report is added to the list of daily reports **Path parameter:** /createreport **Path query:** /createDailyReport?action = create	
Name	**Description**

body * required
 	
and
 	
Parameter content type:
	{
	reporter_id: 100
	project_id: 50,
	completedTask: “Corrosion evaluation of columns”,
	tasksInProgress: “Approach Eastern side columns
	evaluate above water portion”,
	Obstacles: “thunder storm’,
	nextTask: ‘Bridge surface structure’
	}
	application\json

Responses	**Description**
200	The request has been fulfilled and the new daily report being created.
304	The request cannot be fulfilled due to missing data

2.Endpoint: http://www.projectdailyreport.com

This is the root end point

HTTP verb: GET => By using this HTTP verb, all the reports created, list of reporters, list of projects **Path parameter:** http://www.projectdailyreport.com	
Name	**Description**

body * required	{
	reporter_id: 100
	project_id: 50,
	completedTask: “Corrosion evaluation of columns”,
	tasksInProgress: “Approach Eastern side columns and evaluate above water portion”,
	Obstacles: “thunder storm’,
	nextTask: ‘Bridge surface structure’
	} => getting daily reports
	{
	reporterName: “Sarah” => getting reporter names
	}
	{
	projectName:”Summit Bridge Maintenance”
	} => getting project names

Parameter content type:	application\json
Responses	**Description**
200	The request has been fulfilled and the new daily report being created.
304	The request cannot be fulfilled due to missing data

3. Endpoint: http://www.projectdailyreport.com/reporterresult

HTTP verb: POST => By using this HTTP verb, the daily reports are searched by their reporter name **Path parameter:** /reporterresult **Path query:** /?reporter_id = 100 => finds all the daily reports related to reporter with reporter_id of 100	
Parameters	
Name	*Description*
reporter_id*required integer($int64) Parameter content type:	the id of the reporter to return application\json
Responses	**Description**

200	Successful operation
	
	Example:
	{
	reporter_id: 100
	project_id: 50,
	completedTask: “Corrosion evaluation of columns”,
	tasksInProgress: “Approach Eastern side columns and evaluate above water portion”,
	Obstacles: “thunder storm’,
	nextTask: ‘Bridge surface structure’
	}

404	No report was found

4. Endpoint: http://www.projectdailyreport.com/projectresult

HTTP verb: POST => By using this HTTP verb, the daily reports are searched by their project name **Path parameter:** /projectresult **Path query:** /?project_id = 50 => finds all the daily reports related to reporter with project_id of 50	
Parameters	
Name	*Description*
project_id*required integer($int64) Parameter content type:	the id of the project to return application\json
Responses	**Description**

200	Successful operation
	
	Example:
	{
	reporter_id: 100
	project_id: 50,
	completedTask: “Corrosion evaluation of columns”,
	tasksInProgress: “Approach Eastern side columns and evaluate above water portion’,
	Obstacles: “thunder storm’,
	nextTask: ‘Bridge surface structure’
	}

404	No report was found

5. **Stretch feature:** Endpoint: http://www.projectdailyreport.com/newUser

HTTP verb: POST => By using this HTTP verb a new user is added to the list of users **Path parameter:** /newUser **Path query:** /newUser?action = create	
Name	**Description**

body * required
 	
Parameter content type:
	{
	reporter_id: 105
	FirstName: 'Martha'
	LastName: 'Anderson'
	Department:'Sales'
	Email:'MarthaAnderson@example.com
	}
	application\json

Responses	**Description**
201 304	The request has been fulfilled and the new user being created The request cannot be fulfilled due to missing data

6. **Stretch feature:** Endpoint: http://www.projectdailyreport.com/newProject

HTTP verb: POST => By using this HTTP verb a new project is added to the list of projects **Path parameter:** /newProject **Path query:** /newProject?action = create	
Name	**Description**

body * required	{
	project_id: 110
	ProjectName: 'Solar system analysis'
	ProjectManager: 'Gary Hunt'
	StartDate:'11/29/18'
	EndDate:'12/20/18'
	}

Parameter content type:	application\json
Responses	**Description**
201	The request has been fulfilled and the new project being created
304	The request cannot be fulfilled due to missing data

Describes the purpose of each endpoint based on the goals of a user persona as it relates to their data.

- http://www.projectdailyreport.com/

This root endpoint is considered as the root endpoint which users can check the last 10 created daily reports. Moreover, for the personas who want to search daily report by the reporter name or project name, this endpoint is going to be the start point. For instance,

a persona called George who is a project manager wants to search all the daily reports submitted by one of the project team members like John or he wants to search all the daily reports based on the project name like ‘Summit Bridge Maintenance’. GET http verb is used in this endpoint for getting all reporter names and project names in their related dropdowns and also daily reports.

- http://www.projectdailyreport.com/createreport

This endpoint is for the personas of the application who want to create new daily reports. For example, a persona like Emily use this endpoint to create her daily report and submit it which is done by POST http verb, her report will be added to the list of daily reports. If she does not enter data some of the required fields, the saving operation will be unsuccessful and there will be response of 304 that there is a missing data and creating the daily report should be repeated.

- http://www.projectdailyreport.com/reporterresult

This endpoint is used to show the daily reports filtered by the reporter. The POST method is used to send filtered data into this page. Also GET method is used to receive data the filtered daily reports by reporter name.

- http://www.projectdailyreport.com/projectresult

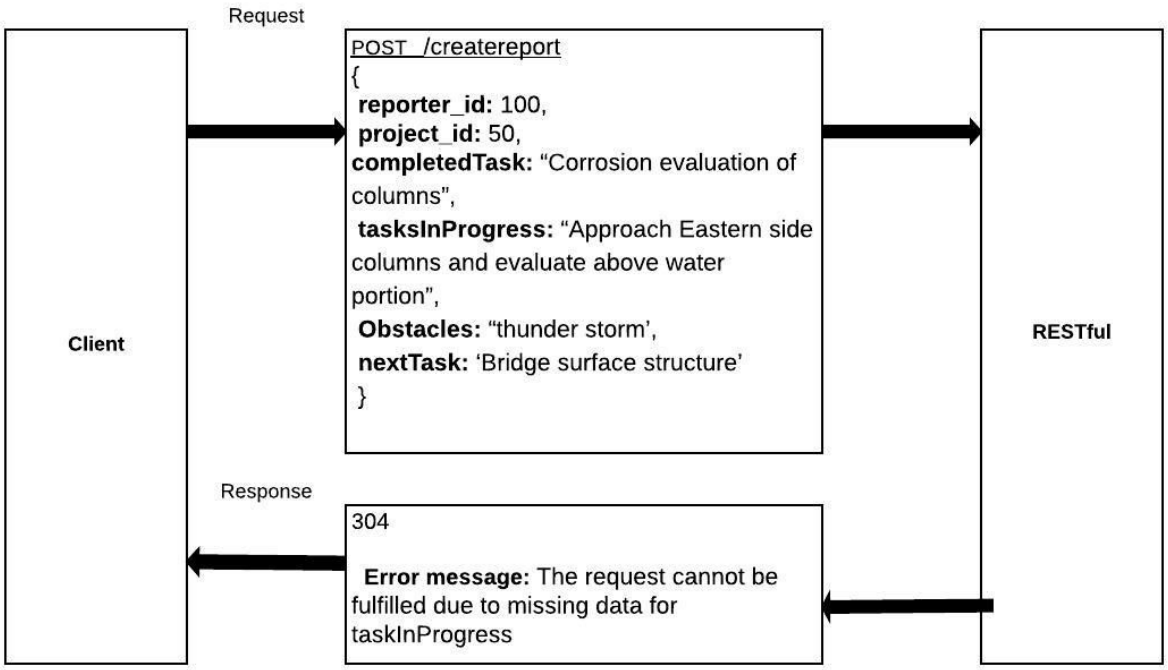
This endpoint is used to show the daily reports filtered by the project name. The POST method is used to send filtered data into this page. Also GET method is used to receive data the filtered daily reports by project name.

-Stretch feature: http://www.projectdailyreport.com/newUser
This endpoint is for the personas who are managers and would like to enter new user names in the application themselves. For instance a persona called Ben is the vice president of a solar system company and would like to enter new people who are added to their team, for doing this he goes to newUser page and fills in the form for the new user, all these operations are done by POST http verb which the a new user will be added to the list of users. If she does not enter data some of the required fields, the saving operation will be unsuccessful and there will be response of 400 that there is a missing data and creating the new user should be repeated.

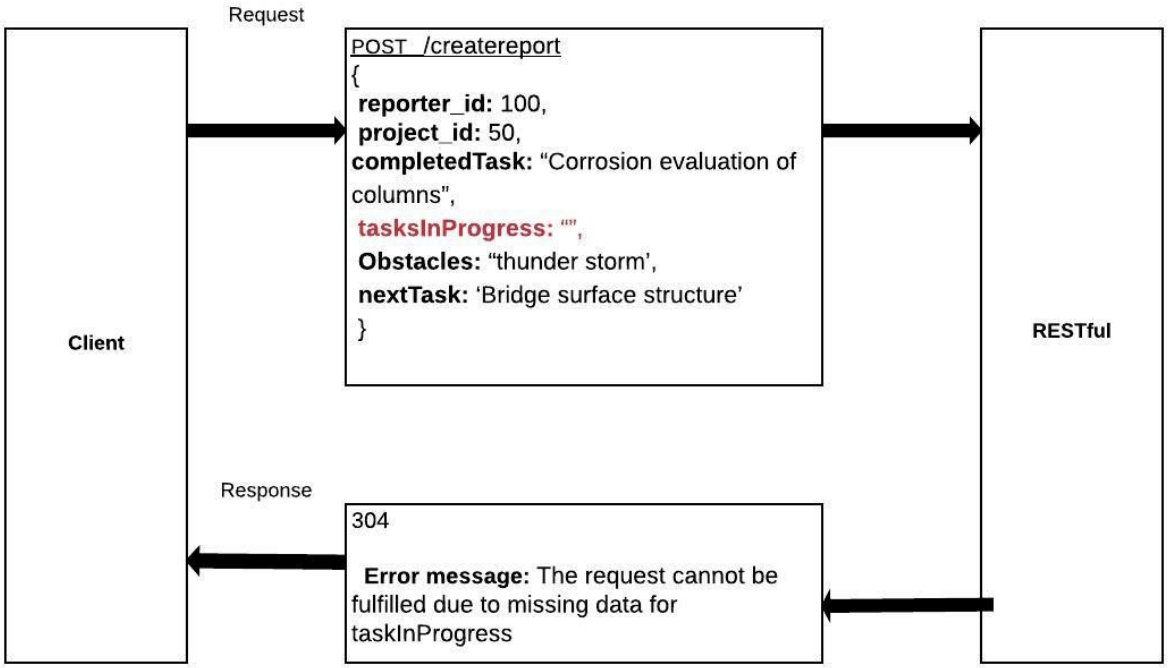
-Stretch feature: http://www.projectdailyreport.com/newProject
This endpoint is for the personas who are managers and would like to enter new projects information in the application themselves. For instance a persona called Ben is the vice president of a solar system company and would like to enter new projects in their company, for doing this he goes to newProject page and fills in the form for the new project, all these operations are done by POST http verb which the a new project will be added to the list of projects. If she does not enter data some of the required fields, the saving operation will be unsuccessful and there will be response of 400 that there is a missing data and creating the new project should be repeated.

Provide example requests and responses, including erroneous requests:

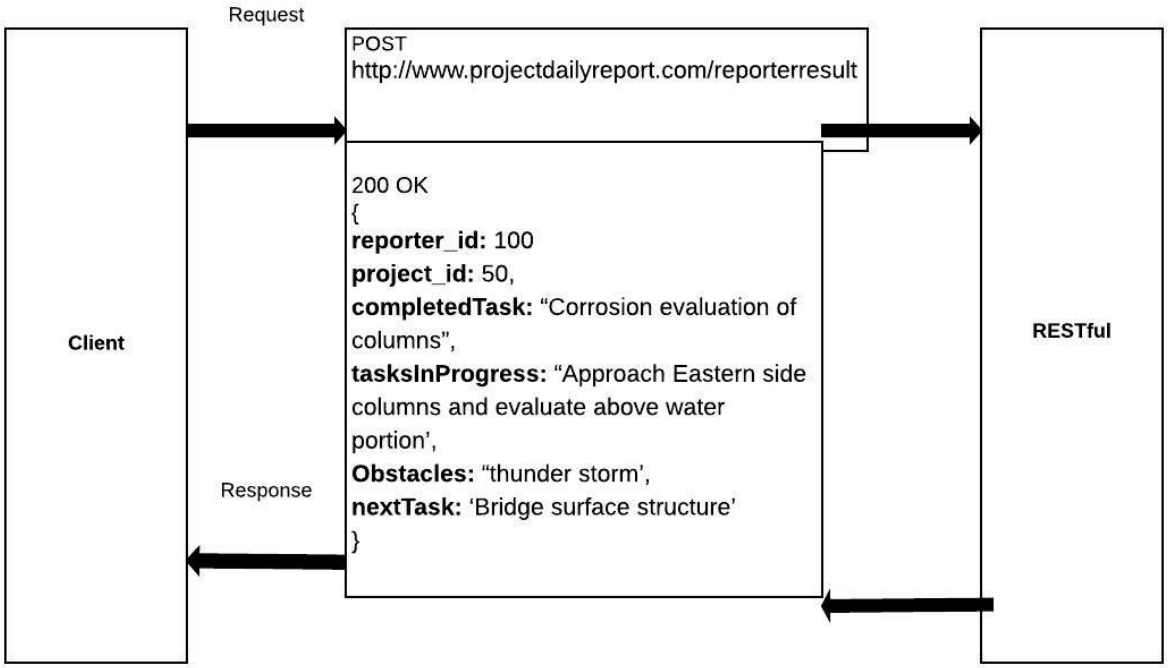
1. Request - Response related to successful creating of a daily report



2. Request-response related to unsuccessful creation of a daily report:

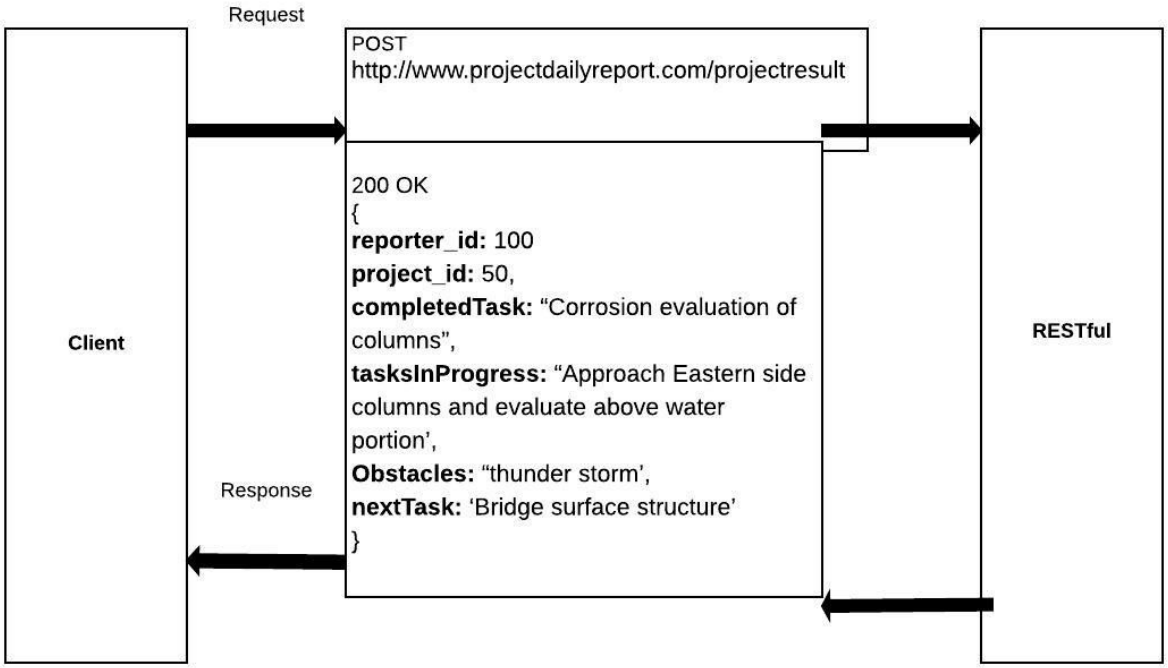


3. Request-response related to finding daily reports by reporter name:



4. Error message related to invalid reporter_id for searching daily reports:

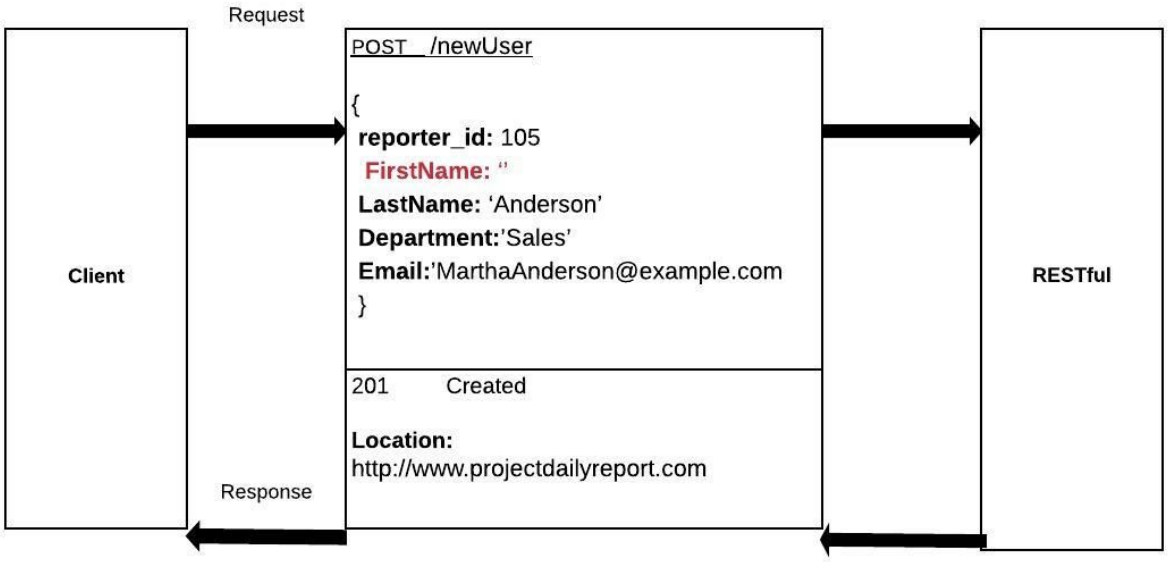
5. Request-response related to finding daily reports by project name:



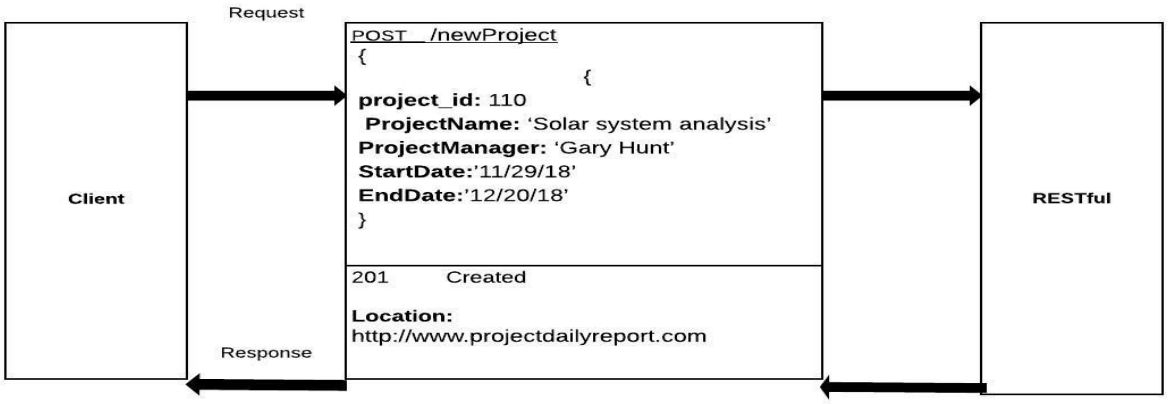
6. Error message related to invalid project_id for searching daily reports:

7. Stretch feature: Request - Response related to successful creating of a new user

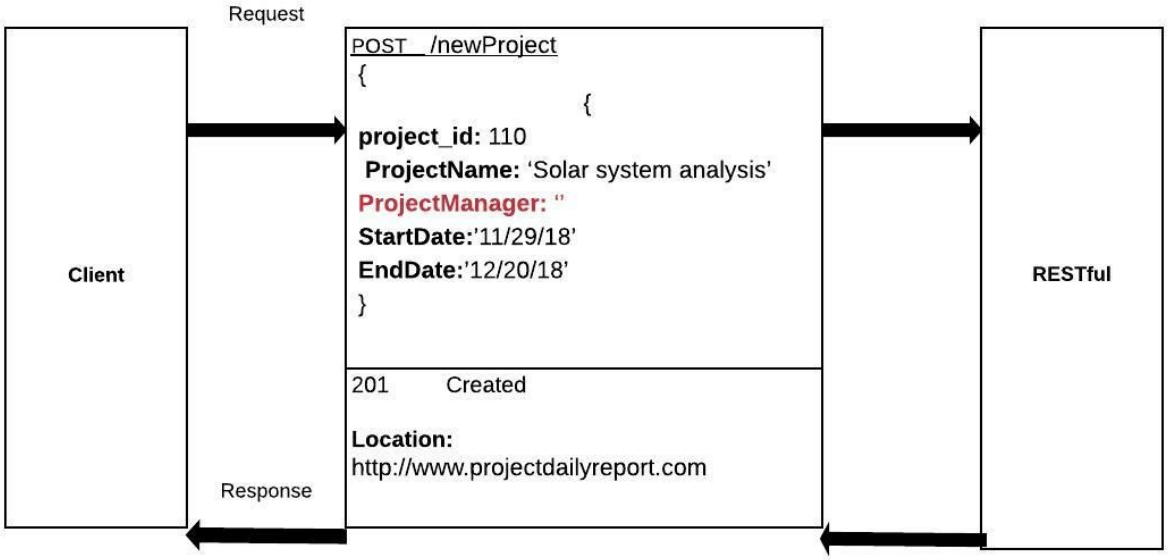
8. Stretch feature: Request-response related to unsuccessful creation of a user:



9. Stretch feature: Request - Response related to successful creating of a new project

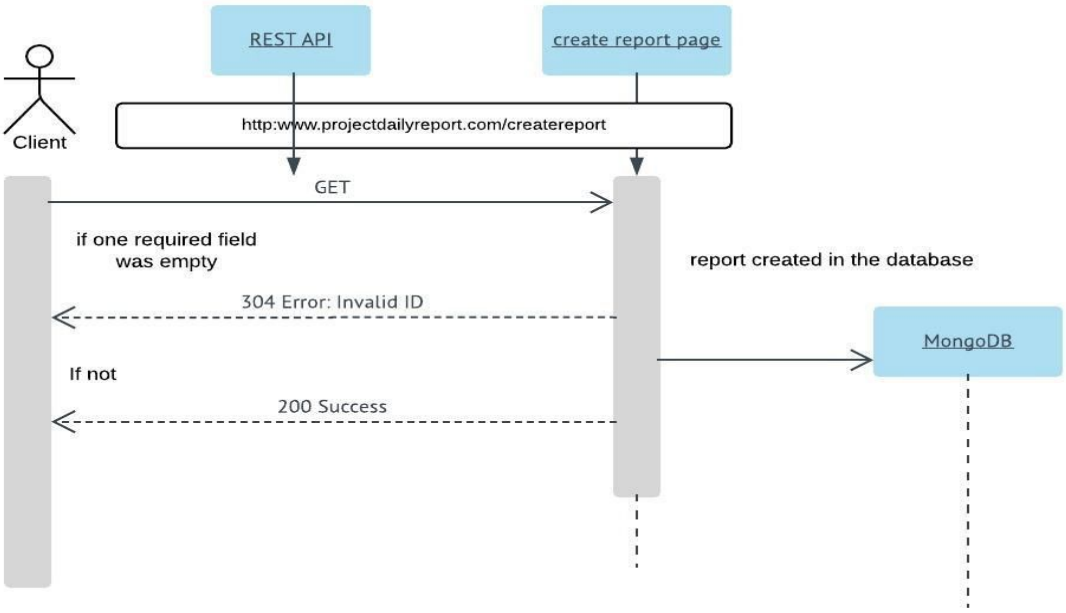


10. Stretch feature: Request - Response related to unsuccessful creating of a new project

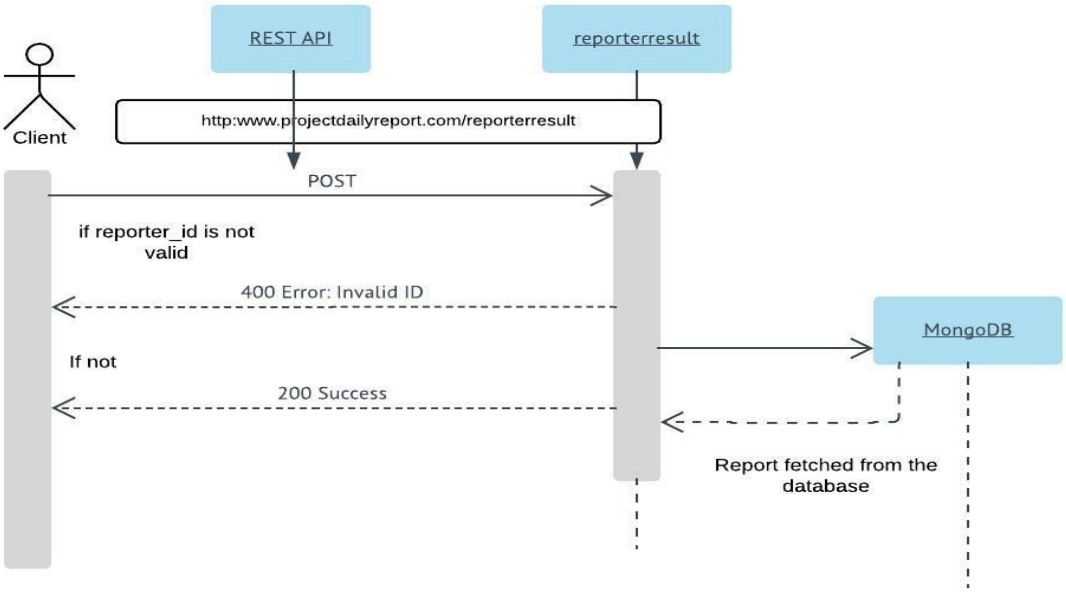


Diagrams which user interface pages will interact with which service endpoints:

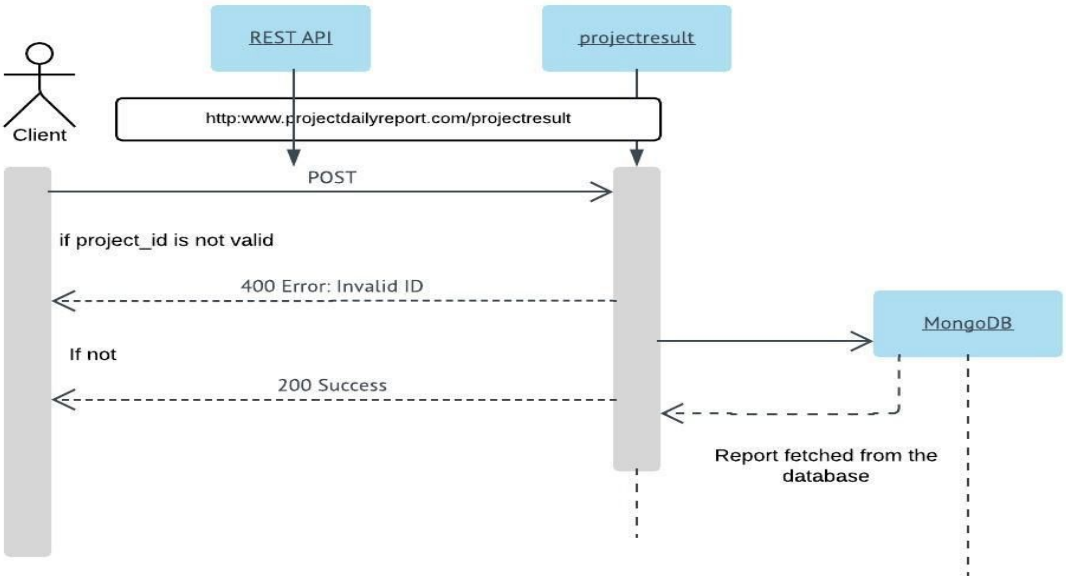
1. Interaction with the interface for creating a new daily report:



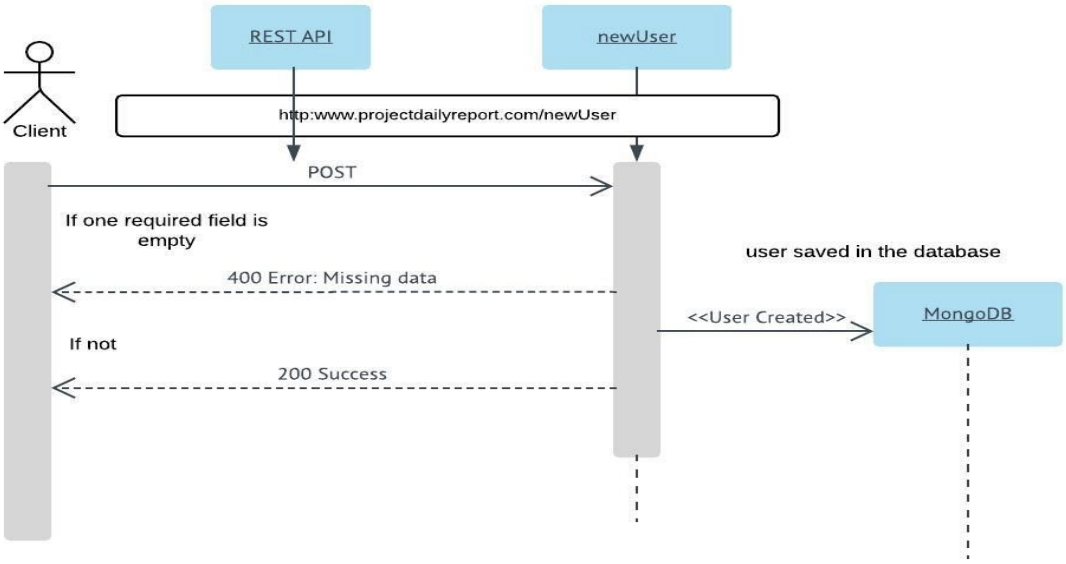
2. Interaction with the interface for searching reports by the reporter:



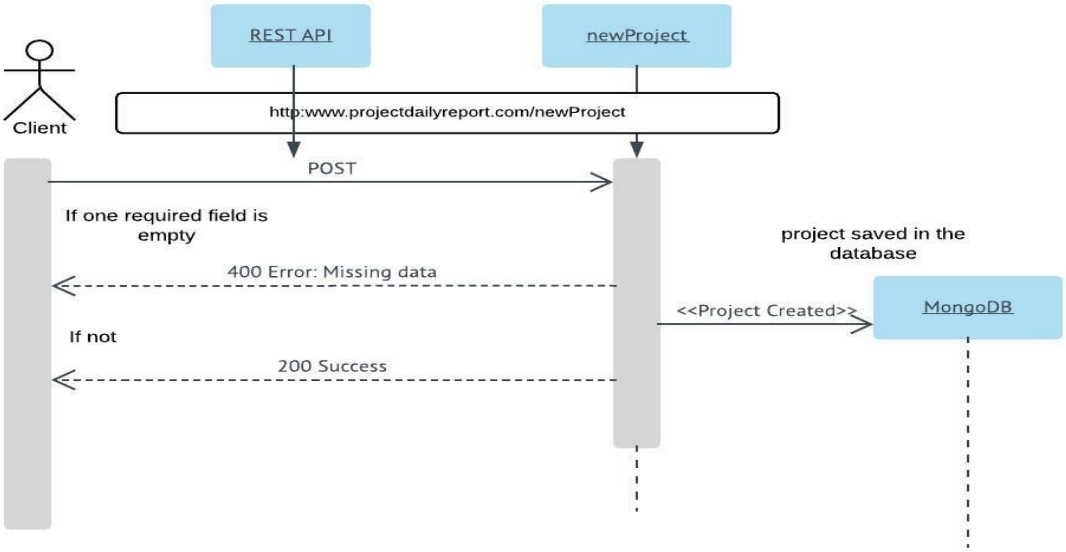
3. Interaction with the interface for searching reports by the project name:



4. Stretch feature: Interaction with the interface for creating a new user:



5. Stretch feature: Interaction with the interface for creating a new project:


REST API
newProject
Client
http:www.projectdailyreport.com/newProject
POST
If one required field is empty
400 Error: Missing data
If not
200 Success
project saved in the database
<<Project Created>>
MongoDB