Prerequisite:

1.install nodejs in your pc

Link : https://nodejs.org/en/download/

2. Then install latest npm in your pc

Command: `npm install -g npm`

3. Then install nodemon

Command : npm i -g nodemon

Now go to the phase_3 directory.open the directory in vs code

Backend

In this, you can see two folders server and client

Open the vs code terminal. Then command in terminal cd server

then, expand the server folder, and in the server directory, you can see the package.json file. In this file, all the dependencies are added. Now command "npm install", That will do the package installation. After successfully the installation type another command "npx nodemon server" that will start the server. If successfully run, you should see the "successfully connected to DB".

Frontend

While running the server, split another terminal in vs code then command cd client. Now, open the client folder, and in the client directory, you can see the package.json file. In this file, all the dependencies are added. Now command "npm install", That will do the package installation. After successfully the installation type another command "npm start" that will start the client.

Demo Data

First login with admin account then create other user

Credentials:

nslID : 1001,

password: Admin12

Methods And Response

1. Sign in

Authenticate the user with the system and obtain the auth_token

Request

Method	URL
POST	/userSignin

Type	Params	Values
BODY BODY	`nslId` `password`	`string` `string`

`nslID, password` must be in the request body.

Status	Response
| 200 | ```
{
    "data": {
        "name": "saimom",
        "nslId": "1002",
        "role": "teamlead"
    }
}
``` |
| 400 | `{"error":"Incorrect username or password."}` |

2. Create User

Create new user
Only Amin or Team Leader can create user

Request

Method	URL

POST	/userSignup

Type	Params	Values
BODY	name nslId password cpassword role	string string string string string

Response

Status	Response

200	```
	{
	    "data": {
	        "name": "sakir",
	        "nslId": "1003",
	        "role": "user"
	    }
	}
	``
	Response message : User created Successfully

400	{"message": "Fill All the fields"} {" message"}: " User Id is already exist" {" message"}: "Password doesn't match"

3. update password

Update individual user password

Request

Method	URL
PUT	`/updatePassword/:id`

Type	Params	Values
HEADERS	Authoriazation(token)	string
BODY	User info(password)	string
Parameter	User id	string

Response

Status	Response

200	{
	    "_id": "62888c5e25476aefb4c2b4da",
	    "name": "saimom",
	    "nslId": "1002",
	    "password": "$2b$10$oZRcY/l3ApWdolZsamarS.krWHwYtiPK6b21UhsZ/vJnHlnwbVae6",
	    "cpassword": "$2b$10$oZRcY/l3ApWdolZsamarS.krWHwYtiPK6b21UhsZ/vJnHlnwbVae6",
	    "role": "teamlead",
	    "tokens": [
	        {
	            "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg4OG M1ZTI1NDc2YWVmYjRjMmI0ZGEiLCJuYW1lIjoic2FpbW9tIiwiaWF0IjoxNjUzMTE2MzUzfQ.oqJB tLWDs8ZZzWC4a9ddzmo1oeXatAC-W88R5MWzDbw",
	            "_id": "62888dc1f27d968b902eaed5"
	        },
	        {
	            "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg4OG M1ZTI1NDc2YWVmYjRjMmI0ZGEiLCJuYW1lIjoic2FpbW9tIiwiaWF0IjoxNjUzMTE2MzcxfQ.8sVJ F6w4IQ6dvth1VTRwUeLY7QYFCZJPfsf8pWVtWds",
	            "_id": "62888dd33d2750fef3eb28ad"
	        }
	    ],
	    "createdAt": "2022-05-21T06:53:18.437Z",
	    "updatedAt": "2022-05-21T08:05:18.074Z",
	    "__v": 2
	}

	Message:"Password updated successfully"
400	{"message": "password does not match"}

4. get all or filtered user

Get more information on a particular recipe

Request

Method	URL
GET	/getUser?status=(all or teamlead or user)

Type	Params	Values
HEADERS query	Authoriazation(token) Status(User type status)	String String

Here Only role admin get the all user info.

Response

Status	Response

200	[
	{
	    "_id": "6288a3cc308bfe6c2f4154b6",
	    "name": "sakir",
	    "nslId": "1003",
	    "password": "$2b$10$xUeh0RlEpjLSVkUTfRBU.ux7yoOXqSIpUm3YHeU2WUq5qdGmoDCIe",
	
	    "cpassword": "$2b$10$xUeh0RlEpjLSVkUTfRBU.ux7yoOXqSIpUm3YHeU2WUq5qdGmoDCIe",
	
	    "role": "user",
	    "tokens": [
	        {

                "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg4Y
TNjYzMwOGJmZTZjMmY0MTU0YjYiLCJuYW1lIjoic2FraXIiLCJpYXQiOjE2NTMxNTY2ODV9.QdwJLKFx
tc4CQx-yjI-qXqEKaIsZEaocLSguyZqBVzI",
                "_id": "62892b4d6662a4f53c96bbfb"
            },
            {
                "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg4Y
TNjYzMwOGJmZTZjMmY0MTU0YjYiLCJuYW1lIjoic2FraXIiLCJpYXQiOjE2NTMxNjkzNTl9.x7Qy-
BEyWpiQNNAX2OeHH6x8WD0m96Z0Qhl0bx5IA-U",
                "_id": "62895ccfe19381c4fe848f96"
            }
        ],
        "createdAt": "2022-05-21T08:33:16.757Z",
        "updatedAt": "2022-05-21T21:42:39.124Z",
        "__v": 2
    },
    {
        "_id": "6289613de19381c4fe849015",
        "name": "Abir",
        "nslId": "1004",
        "password": "$2b$10$PRsVANNIDp2MWTAMGaxttOaDKk6lzHlaQA.gsyoTItigCBgAlWns
m",
        "cpassword": "$2b$10$PRsVANNIDp2MWTAMGaxttOaDKk6lzHlaQA.gsyoTItigCBgAlWn
sm",
        "role": "user",
        "tokens": [
            {
                "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg5N
jEzZGUxOTM4MWM0ZmU4NDkwMTUiLCJuYW1lIjoiQWJpciIsImlhdCI6MTY1MzE3MjM2NH0.5tGEGuVcF
Gu1mo1Ds7wsve-7ucEKaM3ITkYWm8YGhPo",
                "_id": "6289688c0170d6fcd516ebd4"
            },
            {
                "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg5N
jEzZGUxOTM4MWM0ZmU4NDkwMTUiLCJuYW1lIjoiQWJpciIsImlhdCI6MTY1MzE5MzIyMn0.P-
_3QO8kI4jI_2pXHmhigdRr1FH7Uf2FYo7251DjfbE",
                "_id": "6289ba06449bad38f05a4a59"
            },
            {
                "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg5N
jEzZGUxOTM4MWM0ZmU4NDkwMTUiLCJuYW1lIjoiQWJpciIsImlhdCI6MTY1MzE5NDA1Mn0.6304Lx2Zb
DXcjcHs1j3-GXEhCKvAQmO8dmaCYyvsZcA",
                "_id": "6289bd44449bad38f05a4c98"
            },
            {

```
                    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg5N
jEzZGUxOTM4MWM0ZmU4NDkwMTUiLCJuYW1lIjoiQWJpciIsImlhdCI6MTY1MzE5NDQzM30.Mh8XlGfpe
FCbxuc1xQ8z0v_B5GjIGVjPMoZONnEk6mw",
                    "_id": "6289bec1449bad38f05a4da5"
                },
                {
                    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2Mjg5N
jEzZGUxOTM4MWM0ZmU4NDkwMTUiLCJuYW1lIjoiQWJpciIsInJvbGUiOiJ1c2VyIiwiaWF0IjoxNjUzM
jI2MTI1fQ.o2HqcQx3qNZeyAvOKlazKcw-IHEJ6n5GpYUkgn9sRTM",
                    "_id": "628a3a8d0b373fb4727bbd9a"
                }
            ],
            "createdAt": "2022-05-21T22:01:33.663Z",
            "updatedAt": "2022-05-22T13:28:45.890Z",
            "__v": 19
        },

    ]
```

| 500 | {"message": "Something went wrong"} |

5. get Auth data

Getting the authenticate user info

Request

Method	URL
GET	/protecteddata

Type	Params	Values
HEADERS	Authoriazation(token)	string

6. Project create

Create user by admin and team leader

Request

Method	URL
POST	/createproject

Type	Params	Values
HEADERS BODY	Authoriazation(token) Title startDate endDate description teamlead members	string string string string string string string

Response

Status	Response
200	```{

    "message": "Project created Successfully"

}``` |
| 400 | { "message": "Required fields must be filled" } |
| | |

7. get Project

Getting all the Project to show

Request

Method	URL
GET	/getproject

Type	Params	Values
HEADERS	Authoriazation(token)	string

Response

Status	Response
200	```[
{
"_id": "6288a4fe2eb71960b35e04bf",
"title": "test",
"startDate": "2022-01-04T18:00:00.000Z",
"endDate": "2022-02-04T18:00:00.000Z",
"description": "sdasdas",
"teamLead": {
"name": "saimom",
"nslId": "1002",
"role": "teamlead"
},
"members": [
{
"name": "sakir",
"nslId": "1003",
"role": "user"
}
],```

```
        "research": [],

        "createdAt": "2022-05-21T08:38:22.945Z",

        "updatedAt": "2022-05-21T08:38:22.945Z",

        "__v": 0

    },

    {

        "members": [],

        "_id": "6288aceb8342beaaabd4c4ba",

        "title": "test",

        "startDate": "2022-01-04T18:00:00.000Z",

        "endDate": "2022-02-04T18:00:00.000Z",

        "description": "sdasdas",

        "teamLead": {

            "name": "saimom",

            "nslId": "1002",

            "role": "teamlead"

        },

        "research": [],

        "createdAt": "2022-05-21T09:12:11.995Z",

        "updatedAt": "2022-05-21T09:12:11.995Z",

        "__v": 0

    },

    {

        "_id": "6288c1e657c4a6bc97ee8b32",

        "title": "test",

        "startDate": "2022-01-04T18:00:00.000Z",

        "endDate": "2022-02-04T18:00:00.000Z",

        "description": "deepICR",

        "teamLead": {
```

```json
                "name": "saimom",
                "nslId": "1002",
                "role": "teamlead"
            },
            "members": [],
            "research": [],
            "createdAt": "2022-05-21T10:41:42.183Z",
            "updatedAt": "2022-05-21T10:41:42.183Z",
            "__v": 0
        },
        {
            "_id": "6288c21c9025034ac255c403",
            "title": "test",
            "startDate": "2022-01-04T18:00:00.000Z",
            "endDate": "2022-02-04T18:00:00.000Z",
            "description": "deepICR",
            "teamLead": {
                "name": "saimom",
                "nslId": "1002",
                "role": "teamlead"
            },
            "members": [
                {
                    "name": "sakir",
                    "nslId": "1003",
                    "role": "user"
                },
                {
                    "name": "sakir",
```

```
                    "nslId": "1003",

                    "role": "user"

               }

          ],

          "research": [],

          "createdAt": "2022-05-21T10:42:36.463Z",

          "updatedAt": "2022-05-21T10:42:36.463Z",

          "__v": 0

     }

]
```

| 400 | { "message": "There is no project" } |

8. update project

update individual project by admin and teamleader

Request

Method	URL
PUT	/updateProject/:id

Type	Params	Values
HEADERS BODY	Authoriazation(token) Title Members Start date End date Teamlead	string string [string] String String String(user_id)

Response

Status	Response

| 200 | {

    "_id": "6288c21c9025034ac255c403",

    "title": "model train",

    "startDate": "2022-01-04T18:00:00.000Z",

    "endDate": "2022-02-04T18:00:00.000Z",

    "description": "deepICR",

    "teamLead": {

        "name": "saimom",

        "nslId": "1002",

        "role": "teamlead"

    },

    "members":

        {

            "name": "sakir",

            "nslId": "1003",

            "role": "user"

        },

    "research": [],

    "createdAt": "2022-05-21T10:42:36.463Z",

    "updatedAt": "2022-05-21T10:57:48.556Z",

    "__v": 0

} |
| 400 | { "message": "All fields must be field up" } |

8. delete single project

Delete only by admin and teamleader

Request

Method	URL
DELETE	/deleteproject/:projectid

Type	Params	Values
HEADERS	Authorization(token)	string

Response

Status	Response

200	{
	    "message": "deleted successfully"
	}

400	{ "message": "Project not found" }

9. Search R&D Project

Request

Method	URL
POST	/searchproject

Type	Params	Values
HEADERS BODY	Authoriazation(token) searchText	string string string

Response

Status	Response
| 200 | ```json
{
        "_id": "6288c1e657c4a6bc97ee8b32",
        "title": "test",
        "startDate": "2022-01-04T18:00:00.000Z",
        "endDate": "2022-02-04T18:00:00.000Z",
        "description": "deepICR",
        "teamLead": {
            "name": "saimom",
            "nslId": "1002",
            "role": "teamlead"
        },
        "members": [],
        "research": [],
        "createdAt": "2022-05-21T10:41:42.183Z",
        "updatedAt": "2022-05-21T10:41:42.183Z",
        "__v": 0
    }
``` |
| 400 | {"message": "There is no data to show"} |

10. upload paper

Request

Method	URL
POST	/uploadpaper

Type	Params	Values
HEADERS BODY	Authoriazation(token) title paperLink contributerName	string string string string

Response

Status	Response

200	```
	{
	    "data": {
	        "title": "exercise",
	        "link": "https://res.cloudinary.com/dm9wer83l/image/upload/v1653143100
	/w6egehgxryizpzpmzz7u.pdf",
	        "_id": "6288f63dd802c3cbf78b854a",
	        "__v": 0
	    }
	}
	{"message": "Paper Posted Successfully"}
	``

400	{"message": "Fill all the fields"}

11.Get All Paper

Getting All Papers to show

Request

Method	URL
GET	/getpaper

Type	Params	Values
HEADERS	Authoriazation(token)	string

Response

Status	Response

200	`[`
	`{`
	    `"data": {`
	       `"title": "exercise",`
	       `"link": "https://res.cloudinary.com/dm9wer83l/image/upload/v1653143100`
	`/w6egehgxryizpzpmzz7u.pdf",`
	       `"_id": "6288f63dd802c3cbf78b854a",`
	       `"__v": 0`
	    `}`
	`},`
	`{`
	    `"data": {`
	       `"title": "exercise",`
	       `"link": "https://res.cloudinary.com/dm9wer83l/image/upload/v1653143100`
	`/w6egehgxryizpzpmzz7u.pdf",`
	       `"_id": "6288f63dd802c3cbf78b854a",`
	       `"__v": 0`
	    `}`
	`}`
	
	`]`

400	`{"message": "There is no paper to show"}`

12. Delete Paper

Delete paper by admin and team leader.

Request

Method	URL
DELETE	/deletepaper/:paperid

Type	Params	Values
HEADERS POST	Authoriazation(token) User info(password, name)	string string

Response

Status	Response
200	{"message": "Deleted Successfully"}
400	{"message": "Paper not found"}

13. Get contribution paper uploading

Get the contribution result on paper uploading

Request

Method	URL
GET	/getpaperContribution

Type	Params	Values
HEADERS	Authoriazation(token)	string

Response

Status	Response
200	[{name:saimom: total:9 }, {name:arafath: total:2 }

	]