



**Kauno technologijos universitetas**

Informatikos fakultetas

**T120B165 Saityno taikomųjų programų  
projektavimas**  
**Projekto „Freddit“ ataskaita**

---

Studentas / Studentė

**Šarūnas Šiaudvytis IFF-2/4**

---

Dėstytojas / Dėstytoja  
**Tomas Blažauskas**

**Kaunas, 2025**

## Contents

1	Sprendžiamo uždavinio aprašymas .....	3
1.1	Sistemos paskirtis .....	3
1.2	Funkciniai reikalavimai .....	3
2	Sistemos architektūra .....	4

# **1 SPRENDŽIAMO UŽDAVINIO APRAŠYMAS**

## **1.1 Sistemos paskirtis**

Projekto tikslas – sukurti vietą kuris palengvintų bendravimą tarp naudotojų, kurie nori organizuoti diskusijas, dalintis žiniomis, patirtimi ar informaciją.

Veikimo principas – pačią kuriamą platformą sudaro dvi dalys: internetinė aplikacija ir aplikacijų programavimo sąsaja (angl. trump. API). Internetinę aplikaciją naudos svečiai, registruoti nariai ir administratoriai.

Nariai, norėdami naudotis forumo platformą, turės prisiregistruoti prie internetinės aplikacijos, galės kurti sub-forumus, kuriuose galės kurti diskusijas, skelbimus ir juose komentuoti ir dalyvauti, sub-forumų kūrėjai galės juos moderuoti, suteikti teises kitiems registruotiems nariams ir atitinkamai uždraust. Administratoriai galėtų valdyti sub-forumus, jei neatitinka aplikacijos taisyklių jas pašalinti ar redaguoti.

## **1.2 Funkciniai reikalavimai**

Neregistruotas sistemos narys galės:

1. Peržiūrėti platformos reprezentacinį puslapį, sub-forumus, diskusijas, komentarus.
2. Prisijungti/užsiregistruoti prie internetinės aplikacijos.

Registruotas sistemos narys galės:

1. Atsijungti nuo internetinės aplikacijos;
2. Prisijungti prie platformos;
3. Kurti turinį:
  - 3.1. Sukurti sub-forumą;
  - 3.2. Sukurti naują diskusiją/įrašą;
  - 3.3. Sukurti komentarą diskusijoje/įraše;
4. Redaguoti ir ištrinti savo sukurtus diskusijas/įrašus/komentarus.
5. Peržiūrėti kitų sukurtais sub-forumus, diskusijas/įrašus, komentarus.
6. Suteikti reitingą diskusijai ir komentarams („Patinka“, „Nepatinka“);
7. Pranešti apie taisykles laužantį turinį.
8. Siųsti žinutes kitiems internetinės aplikacijos nariams.

Administratorius galės:

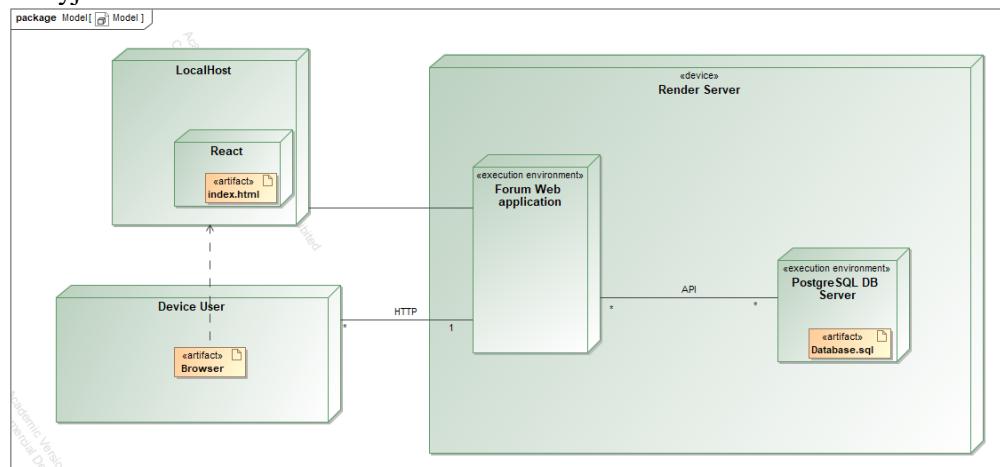
1. Valdyti sub-forumus, diskusijas/įrašus, komentarus.
2. Šalinti ar sustabdyti narius.
3. Valdyti naudotojų teises.

## 2 SISTEMOS ARCHITEKTŪRA

Sistemos sudedamosios dalys

- Kliento pusė (Front-End) – React.js
- Serverio pusė (Back-End) – Node.js
- Duomenų bazė – PostgreSQL

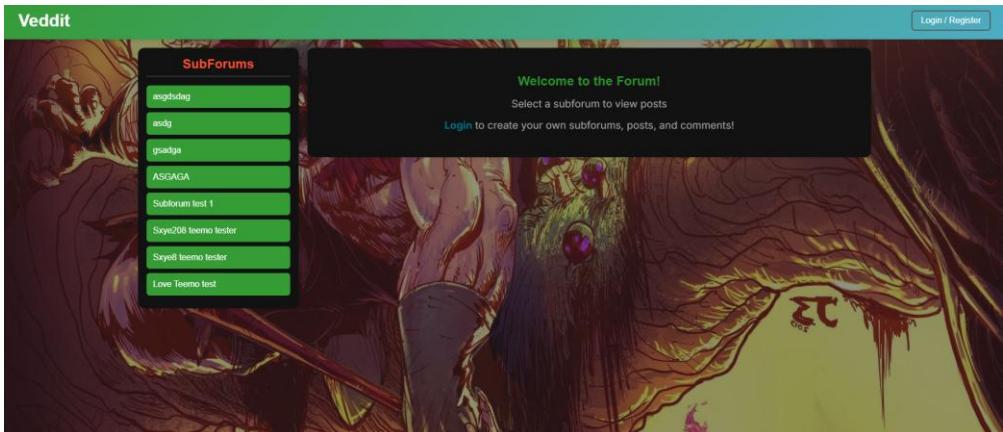
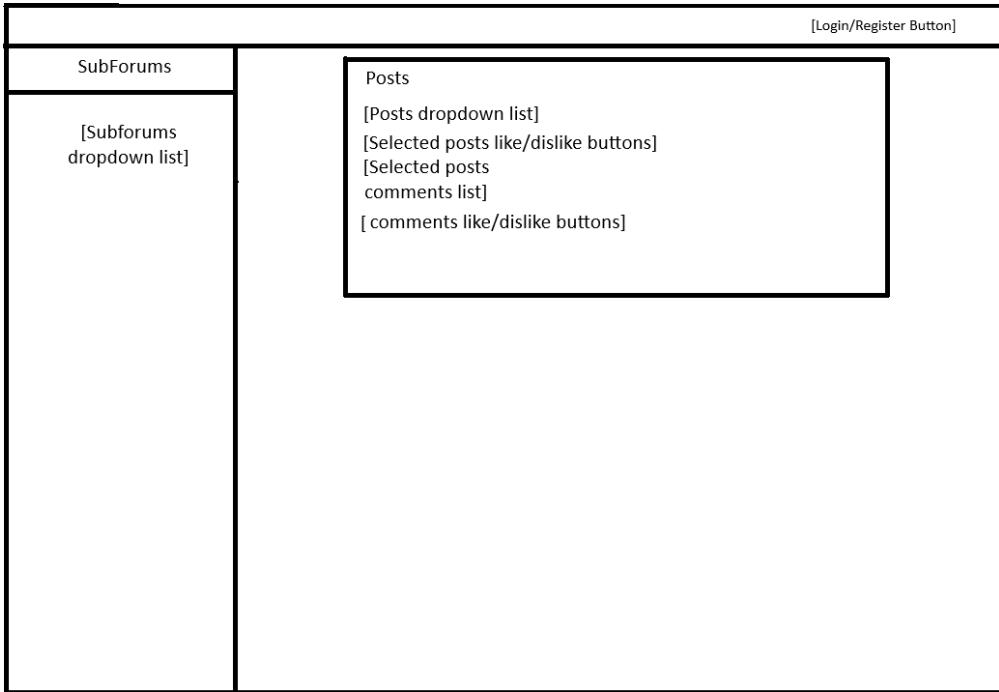
Pav. 1 Atvaizduota kuriamos sistemos diagimo diagrama. Sistemos veikimas paremtas „React“ aplikacija yra talpinama „Vercel“ serveryje ir pasiekiami naudotojui per naršykę naudojant HTTP protokolą. Forumo aplikacija vykdoma „Azure“ serveryje. Ji veikia kartu su duomenų bazę, kuri yra palaikoma PostgreSQL DB serveryje.



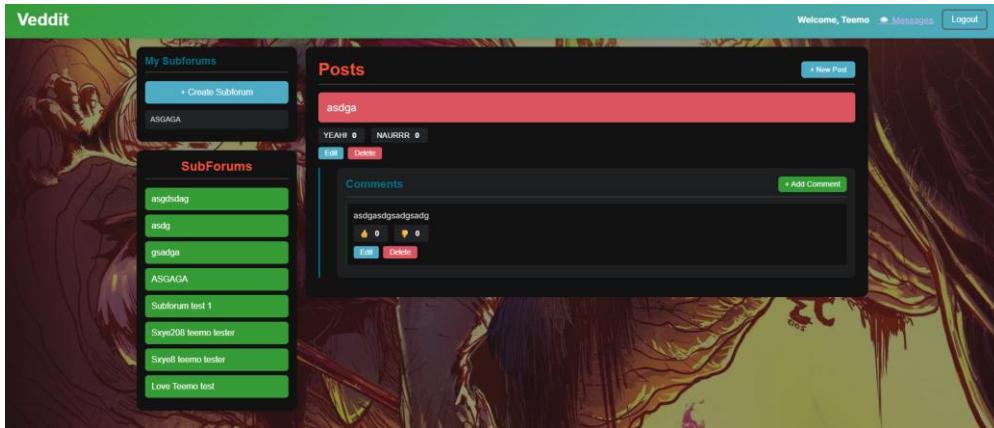
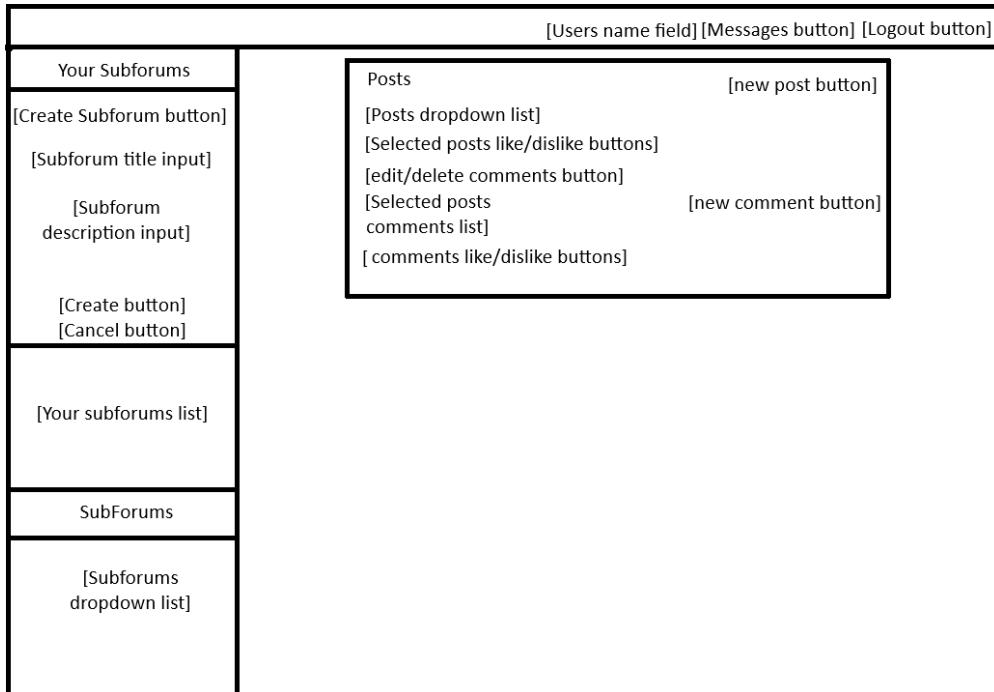
pav. 1 Sistemos diegimo diagrama

### 3 NAUDOTOJO SĄSAJOS PROJEKTAS

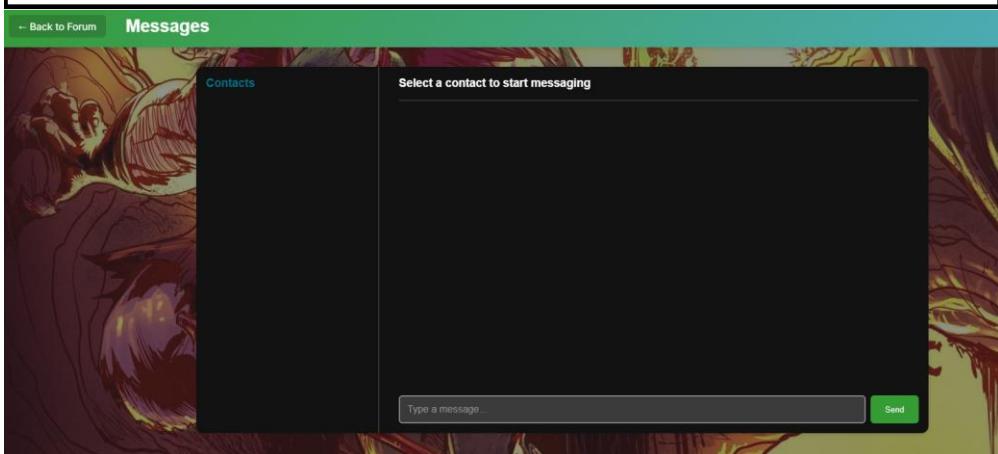
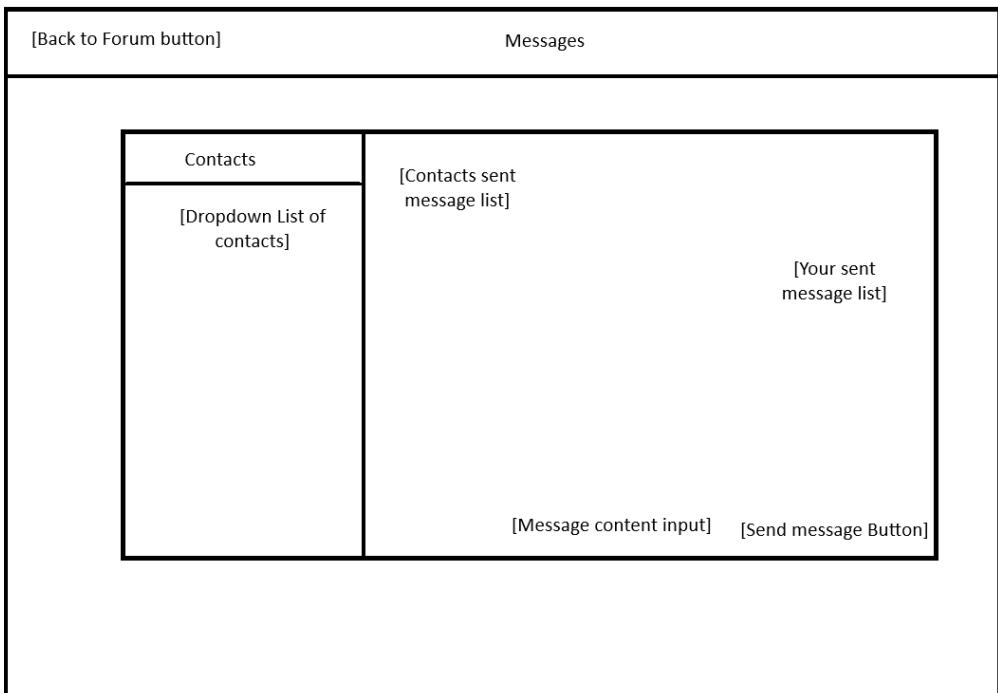
#### 3.1 Guests main page



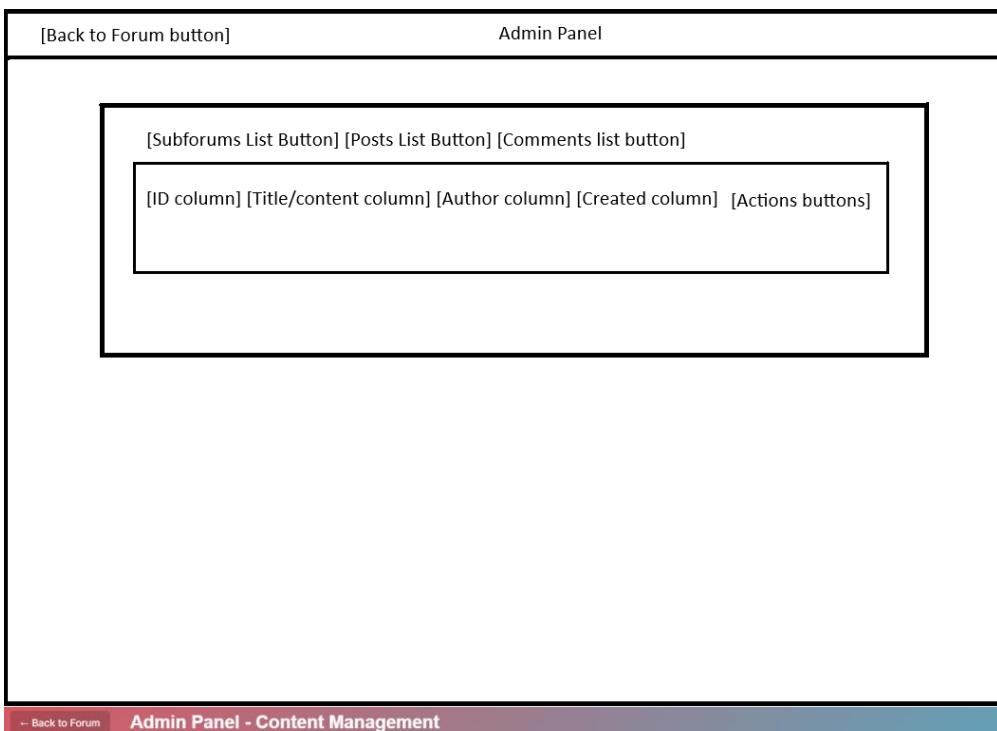
### 3.2 Logged-in users main page



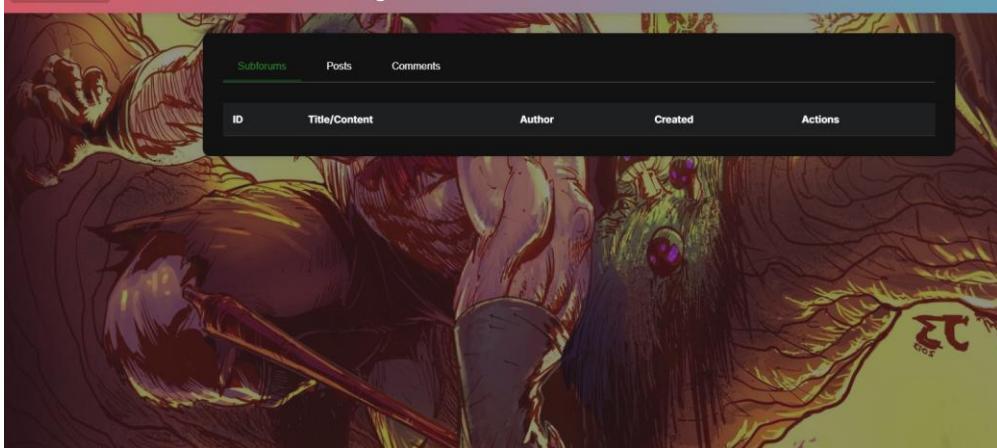
### 3.3 Users messages page



### 3.4 Admin panel page

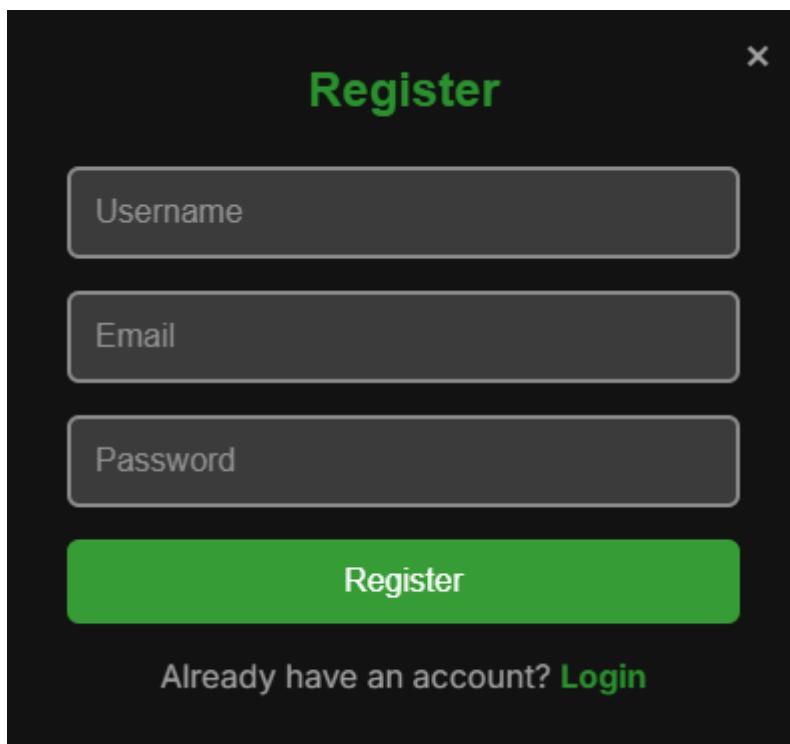
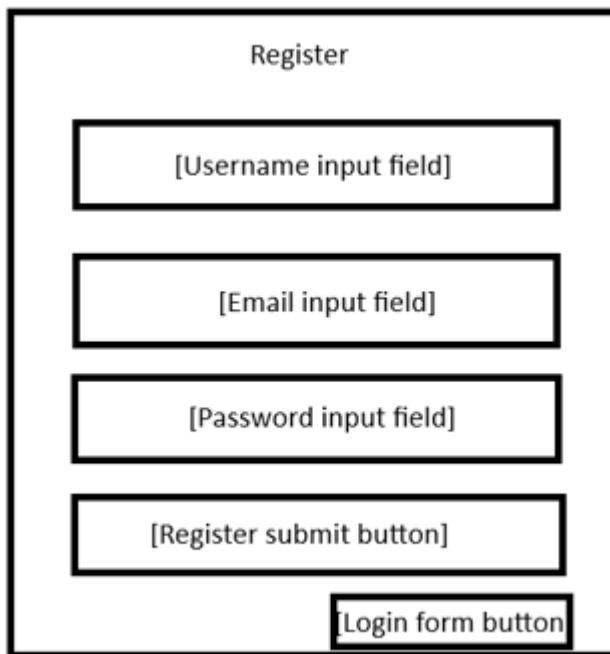


— Back to Forum Admin Panel - Content Management



The screenshot shows a dark-themed Admin Panel interface titled "Admin Panel - Content Management". At the top, there are three navigation links: "Subforums", "Posts", and "Comments". Below the navigation is a table header with columns labeled "ID", "Title/Content", "Author", "Created", and "Actions". The background of the entire interface is a vibrant, colorful illustration of a character's face, possibly from a game or comic book, with purple and yellow hues. A small "TC" logo is visible in the bottom right corner of the background image.

### 3.5 Register page



### 3.6 Login page

Login

[Email input field]

[Password Input field]

[Login submit button]

[Register button]

×

**Login**

Email

Password

**Login**

Don't have an account? **Register**

## 4 API-SPECIFIKACIJA

### 4.1 api/auth/login [POST]

Response codes:
res.status(400) : {"message":"User with this email is not registered"} res.status(500) : „Server Error“ res.status(200) : „OK“
Successful Request:
{ "email": "test1@gmail.com", "password": "test1" }
Response:
{"accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoxfSwiaWF0IjoxNzY1OTA1NTI2LCJleHAiOjE3NjU5MDY0MjZ9.M0lZNFR6jWGRT0vc4iVNdl-awI9t6z_T3xmfjaKQ7c", "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoxfSwiaWF0IjoxNzY1OTA1NTI2LCJleHAiOjE3NjY1MTAzMjZ9.IHFf0ktLnGu-LtzhZ8vsE3Y5mg5MU_Yn98M1B6fe0I8", "expiresIn": "15m", "user": {"id": 1, "username": "test1", "email": "test1@gmail.com"}}
Bad Request:
{ "email": "test1@gmail.com", "password": "test1čę" }
Response(400):
{ "message": "Password is not correct" }

## 4.2 api/auth/register [POST]

Response codes:	
res.status(400) : {"message":"Enter all of the neccessary credentials "}	
res.status(500) : {"Server Error"}	
res.status(200) : {"OK"}	
Successful Request:	
{ "username": "test67", "email": "test67@gmail.com", "password": "test67" }	
Response:	
{ "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjo4fSwiaWF0IjoxNzY1OTA1OTMwLCJleHAiOjE3NjU5MDY4MzB9.xItkaxU-6eSXIMzLrTrMtQ5yqyvaEDUt1twOOHzqstM", "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjo4fSwiaWF0IjoxNzY1OTA1OTMwLCJleHAiOjE3NjY1MTA3MzB9._QK5S0tevNpXYDEZc-9TY2NSUvUm_FWNNzRHpFXLABE", "expiresIn": "15m", "user": {"id": 8, "username": "test67", "email": "test67@gmail.com"} }	
Bad Request:	
{ "email": "test67@gmail.com", "password": "test67" }	
Response(400):	
{ "message": "Enter all of the neccessary credentials" }	

### 4.3 api/subforums/{**{subforumId}**} [GET]

Response codes:
res.status(404) : {"Me see no subforum by that id"}
res.status(500) : „Server Error“
res.status(200) : „OK“
Successful Request:
{ <b>{subforumId = 6}</b> }
Response:
{id: 6, title: 'TeemoLove', description: 'place to love teemo', userid: 2, createdat: '2025-10-12T07:49:10.612Z'}
Bad Request:
{ <b>{subforumId = -1}</b> }
Response(404):
{ message: 'Me see no subforum by that id' }

#### 4.4 api/subforums/create [POST]

Response codes:
res.status(400) : {"message": „Where mah title & description at?!?!”}
res.status(409): {„message“: „This subforum title has already been taken“}
res.status(500) : {„Server Error“}
res.status(201) : {„crated la subforum“}
Successful Request:
{ "title": "Love Teemo", "description": "place to love teemo" }
Response:
{"id":11,"title":"Love Teemo","description":"place to love teemo", "userid":1,"createdat":"2025-12-16T17:33:30.074Z", "message":"created la subforum"}
Bad Request:
{ "title": "Love Teemo", "description": "place to love teemo" }
Response(400):
{ "message": "This subforum title has already been taken" }

#### 4.5 api/subforums/{{subforumid}}/posts/{{postid}} [GET]

Response codes:
res.status(404) : {"No post by that ID"}
res.status(500) : {"Server Error"}
res.status(200) : {"OK"}
Successful Request:
<code>{{PostId = 19}}</code>
Response:
content: "Thank you everyone for gathering for todays Teemo apretiation post." createdat: "2025-11-12T11:43:28.557Z" id: 19 subforumid: 33 title: "Teemo is love, Teemo is life" updatedat: "2025-11-12T11:43:28.557Z" userid: 5
Bad Request:
<code>{{PostId = -1}}</code>
Response(404):
{ message: "No post by that ID" }

#### 4.6 /api/subforums/{subforumid}/posts/create/ [POST]

Response codes:
res.status(400) : {"message": „Title or Content missing“} res.status(404): {„message“: „Subforum by this ID doesnt exist“} res.status(500) : {„Server Error“} res.status(201) : {„created la subforum“}
Successful Request:
{ {subforumid = 6} { "title": "Teemo is love, Teemo is life", "content": "Thank you everyone for gathering for todays Teemo apretiation post." } }
Response:
{ "0": { "id": 13, "title": "Teemo is love, Teemo is life", "content": "Thank you everyone for gathering for todays Teemo apretiation post.", "userid": 1, "subforumid": 6, "createdat": "2025-12-16T17:35:58.234Z", "updatedat": "2025-12-16T17:35:58.234Z" }, "message": "post has been created" }
Bad Request(404):
{ {subforumid = -1} { "title": "Teemo is love, Teemo is life", "content": "Thank you everyone for gathering for todays Teemo apretiation post." }
Response:
{ "message": "Subforum by this ID doesnt exist" }

**4.7 /api/subforums/{{subforumid}}/{{postid}}/comments/{{commentid}}**  
**[GET]**

Response codes:
res.status(404) : {"No post by that ID"}
res.status(500) : {"Server Error"}
res.status(200) : {"OK"}
Successful Request:
<code>{{commentId = 20}}</code>
Response:
content: "I think teemo is greater than yesterday with each passing day" createdat: "2025-11-16T09:26:04.850Z" id: 20 postid: 22 updatedat: "2025-11-16T09:26:57.407Z" userid: 6
Bad Request:
<code>{{PostId = -1}}</code>
Response(404):
{ message: "comment not found by such id" }

#### 4.8 /api/subforums/{{subforumid}}/{{postid}}/comments/create [POST]

Response codes:
res.status(400) : {"message": „Content is required“}
res.status(404): {„message“: „Subforum or Post ID doesn't exist“}
res.status(500) : {„Server Error“}
res.status(201) : {„Comment has been created“}
Successful Request:
{ {postId = 5} } { "content": "Teemo truly is awesome, i agree" }
Response:
{"0": {"id":10, "content": "Teemo truly is awesome, i agree", "userid":1, "postid":5, "createdat": "2025-12-16T17:56:13.786Z", "updatedat": "2025-12- 16T17:56:13.786Z"}, "message": "Comment has been created"}
Bad Request(400):
{ {postId= 5} } { }
Response:
{ "message": "Content is required" }

#### 4.9 api/messages/chatlogs/{{friendsId}} [GET]

Response codes:
res.status(404) : {"Friend by that ID not found"}
res.status(500) : „Server Error“
res.status(200) : „OK“
Successful Request:
<pre> {{friendsID = 2}}</pre>
Response:
0: {id: '1', senderid: '1', receiverid: '2', content: 'hallo frend', datesent: '2025-10-14T16:18:43.749Z'} length: 1
Bad Request:
<pre> {{friendsId = -1}}</pre>
Response(404):
{ message: "Friend by that ID not found" }

#### 4.10 api/messages/chatlogs/send/{{friendsId}} [POST]

Response codes:
res.status(404) : {"Friend by that ID not found"}
res.status(500) : {"Server Error"}
res.status(200) : {"OK"}
Successful Request:
<pre> {{friendsID = 2}} content: "Hello, frendo"</pre>
Response:
<pre>0 : {id: '7', senderid: '1', receiverid: '2', content: 'Hello, frendo', datesent: '2025-12-17T08:05:53.418Z'} length: 1</pre>
Bad Request:
<pre> {{friendsId = -1}}</pre>
Response(404):
<pre>{ message: "Friend by that ID not found" }</pre>

## **5 IŠVADA**

„Veddit“ yra vieta kur žmonės gali susijungti ir kurti grupes, dalintis idėjomis ir mintimis, kuriant subforumus norint sutikti panašius žmones, kuriant postus ir komentarus su kuriais gali save išsireikšti, siušti žinutes ir bendrauti su panašiais interesais dalijančiais žmonėmis.