

## INFORMATIKOS FAKULTETAS

### **T120B165 Saityno taikomųjų programų projektavimas** **Projekto „KanbanDo“ ataskaita**

#### **Studentas:**

Domas Baltrušaitis IFF-0/7

#### **Dėstytojai:**

prof. BLAŽAUSKAS Tomas  
doc. prakt. Mažutienė Rasa

Kaunas 2023

## Turinys

<b>1. Sprendžiamo uždavinio aprašymas .....</b>	<b>3</b>
<b>2. Sistemos architektūra .....</b>	<b>5</b>
<b>3. API specifikacija.....</b>	<b>6</b>
<b>4. Naudotojo sąsaja.....</b>	<b>15</b>
<b>4.1 Home puslapis .....</b>	<b>15</b>
<b>4.2 Autentifikacijos puslapiai .....</b>	<b>17</b>
<b>4.3 Pagrindinis puslapis .....</b>	<b>18</b>
<b>5. Išvados.....</b>	<b>20</b>

# **1. Sprendžiamo uždavinio aprašymas**

## **Sistemos paskirtis**

Projekto tikslas – sukurti Kanban stiliaus lentą skirtą darbų organizavimui, paskirstymui bei planavimui.

## **Veikimo principas**

Kuriamą aplikaciją sudaro dvi dalys:

- „Web“ aplikacija, kuria naudosis atitinkamai Kanban lentai priskirti vartotojai bei administratoriai.
- Aplikacijos programavimo sąsaja, atsakinga už komunikavimą su duomenų baze ir duomenų kaitą aplikacijoje.

Svečias (neregistruotas vartotojas) pirmiausiai turės susikurti naują paskyrą. Taip jis įgaus vartotojo rolę ir galės kurti savo asmenines Kanban lentas. Sukūręs savo lentą, vartotojas tampa tos lentos administratoriumi ir įgauna privilegijas lentą modifikuoti (pvz. Trinti, pervadinti). Taip pat vartotojas gali būti pakviestas prisijungti prie kito asmens sukurtos lentos.

## **Nuoroda į projekto GitHub:**

<https://github.com/rms-ctrl/KTU-KanbanDoProject>

## **Funkciniai reikalavimai**

Neregistruotas sistemos naudotojas galės:

- Peržiūrėti aplikacijos reprezentacinį puslapį.
- Prisijungti/Prisiregistruoti prie internetinės aplikacijos.

Registruotas sistemos naudotojas galės:

- Atsijungti nuo internetinės aplikacijos.
- Sukurti naują Kanban lentą.
- Sukurti užduotis.
- Mėtyti užduotis tarp atitinkamų paruoštumo lygio stulpelių (In progress, Done..)
- Priskirti asmenį užduočiai.
- Peržiūrėti kitų žmonių sukurtas užduotis.
- Matyti užduotim atlikti skirtą laiką.

Administratorius:

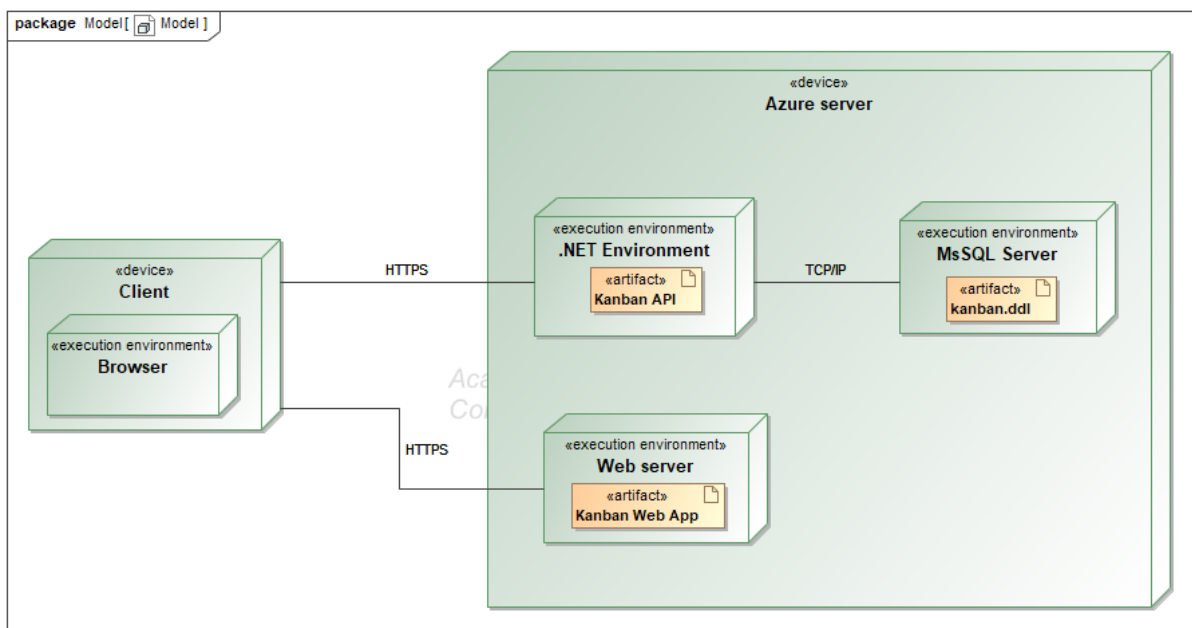
- Keisti lentos pavadinimą.
- Keisti stulpelių pavadinimus.
- Ištrinti lentą.
- Nustatyti užduočiai atlikti skirtą laiką.

## 2. Sistemos architektūra

Sistemos sudedamosios dalys:

- Kliento pusė – naudojant React.js;
- Serverio pusė – naudojant .NET 7.0. Duomenų bazė – Azure SQL Database (MsSQL).
- Autentifikacija/autorizacija – ASP.NET Core Identity sukonfigūruotas naudoti JWT žetonus.

Pav. 1 pavaizduota sistemos diegimo diagrama. Sistemai patalpinti yra naudojamas Azure serveris. Sistema duomenis iš duomenų bazės ims per Kanban API. Pats API vykdo duomenų mainus su duomenų baze per ORM sąsają (Entity Framework Core). Klientui sistema bus pasiekiamą per HTTPS protokolą su ReactJS karkaso pagalba.



*Pav. 1 Kanban projekto diegimo diagrama*

### 3. API specifikacija

Sąsajos funkcijos bus aprašomos lentelėse. Lentelėse bus galima pamatyti API metodo pavadinimą, paskirtį, kelią (route), atsakymo struktūrą, atsakymo kodą bei galimus klaidų kodus.

API metodai:

- **Views**
  - **Get()**
  - **GetMany()**
  - **Create()**
  - **Update()**
  - **Remove()**
- **Columns**
  - **Get()**
  - **GetMany()**
  - **Create()**
  - **Update()**
  - **Remove()**
- **TaskItems**
  - **Get()**
  - **GetMany()**
  - **Create()**
  - **Update()**
  - **Remove()**
- **Auth**
  - **Register()**
  - **Login()**
  - **Refresh()**
  - **Logout()**

## Views modelio API metodai

<b>API metodas</b>	Get() <b>GET</b>
<b>Paskirtis</b>	Gauti vieną „Vaizdą“
<b>Kelias (Route)</b>	api/views/{viewId}
<b>Atsakymo struktūra</b>	<pre>{   "id": 2,   "name": "Pirma lenta",   "description": "Cia mano asmenine pirmoji lenta" }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	404 - jei vaizdo Id negrąžina jokios reikšmės.

<b>API metodas</b>	GetMany() <b>GET</b>
<b>Paskirtis</b>	Gauti visus „Vaizdus“
<b>Kelias (Route)</b>	api/views
<b>Atsakymo struktūra</b>	<pre>[   {     "id": 2,     "name": "Pirma lenta",     "description": "Cia mano asmenine pirmoji lenta"   },   {     "id": 4,     "name": "Trečioji lenta.",     "description": "Tai yra mano trečioji lenta."   } ]</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	-

<b>API metodas</b>	Create() <b>POST</b>
<b>Paskirtis</b>	Sukurti naują „Vaizdą“
<b>Kelias (Route)</b>	api/views
<b>Atsakymo struktūra</b>	<pre>{   "id": 4,   "name": "Trečioji lenta.",   "description": "Tai yra mano trečioji lenta." }</pre>
<b>Atsakymo kodas</b>	201 Created
<b>Galimi klaidų kodai</b>	-

<b>API metodas</b>	Update() <b>PUT</b>
<b>Paskirtis</b>	Atnaujinti „Vaizdą“
<b>Kelias (Route)</b>	api/views/{viewId}
<b>Atsakymo struktūra</b>	<pre>{   "id": 4,   "name": "Trečioji lenta.",   "description": "Tai yra mano trečioji lenta. UPDATED" }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	404 – jei pagal Id nerandamas vaizdas. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.

<b>API metodas</b>	Remove() <b>DELETE</b>
<b>Paskirtis</b>	Ištrinti „Vaizdą“
<b>Kelias (Route)</b>	api/views/{viewId}
<b>Atsakymo struktūra</b>	
<b>Atsakymo kodas</b>	204 No Content
<b>Galimi klaidų kodai</b>	404 - jei vaizdo Id negrąžina jokios reikšmės. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.



## Columns modelio API metodai

<b>API metodas</b>	Get() <b>GET</b>
<b>Paskirtis</b>	Gauti vieną stulpelį
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}
<b>Atsakymo struktūra</b>	<pre>{   "id": 2,   "name": "To do" }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	404 - jei stulpelio arba vaizdo Id negrąžina jokios reikšmės.

<b>API metodas</b>	GetMany() <b>GET</b>
<b>Paskirtis</b>	Gauti visus stulpelius.
<b>Kelias (Route)</b>	api/views/{viewId}/columns
<b>Atsakymo struktūra</b>	<pre>[   {     "id": 2,     "name": "To do"   },   {     "id": 3,     "name": "To do"   },   {     "id": 5,     "name": "Done."   } ]</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	-

<b>API metodas</b>	Create() <b>POST</b>
<b>Paskirtis</b>	Sukurti naują stulpelį.
<b>Kelias (Route)</b>	api/views/{viewId}/columns
<b>Atsakymo struktūra</b>	<pre>{   "id": 5,   "name": "Done." }</pre>
<b>Atsakymo kodas</b>	201 Created
<b>Galimi klaidų kodai</b>	404 – jei pagal Id nerandamas stulpelis. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.

<b>API metodas</b>	Update() <b>PUT</b>
<b>Paskirtis</b>	Atnaujinti stulpelį.
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}
<b>Atsakymo struktūra</b>	<pre>{   "id": 3,   "name": "Done. UPDATED" }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	404 – jei pagal Id nerandamas stulpelis. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.

<b>API metodas</b>	Remove() <b>DELETE</b>
<b>Paskirtis</b>	Ištrinti stulpelį.
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}
<b>Atsakymo struktūra</b>	
<b>Atsakymo kodas</b>	204 No Content
<b>Galimi klaidų kodai</b>	404 - jei stulpelio Id negrąžina jokios reikšmės. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.

## TaskItems modelio API metodai

<b>API metodas</b>	Get() <b>GET</b>
<b>Paskirtis</b>	Gauti vieną užduotį.
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}/tasks/{taskId}
<b>Atsakymo struktūra</b>	<pre>{   "id": 6,   "name": "Task 1",   "description": "Need to do the first task." }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	404 - jei stulpelio arba užduoties Id negrąžina jokios reikšmės.

<b>API metodas</b>	GetMany() <b>GET</b>
<b>Paskirtis</b>	Gauti visas užduotis.
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}/tasks
<b>Atsakymo struktūra</b>	<pre>[   {     "id": 6,     "name": "Task 1",     "description": "Need to do the first task."   },   {     "id": 7,     "name": "Task 2",     "description": "Need to do the second task."   } ]</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	-

<b>API metodas</b>	<b>Create() POST</b>
<b>Paskirtis</b>	Sukurti naują užduotį.
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}/tasks
<b>Atsakymo struktūra</b>	<pre>{   "id": 7,   "name": "Task 2",   "description": "Need to do the second task." }</pre>
<b>Atsakymo kodas</b>	201 Created
<b>Galimi klaidų kodai</b>	404 – jei pagal Id nerandamas stulpelis. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.

<b>API metodas</b>	<b>Update() PUT</b>
<b>Paskirtis</b>	Atnaujinti užduotį.
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}/tasks/{taskId}
<b>Atsakymo struktūra</b>	<pre>{   "id": 7,   "name": "Task 2",   "description": "Need to do the second task. UPDATED" }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	404 – jei pagal Id nerandama užduotis. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.

<b>API metodas</b>	<b>Remove() DELETE</b>
<b>Paskirtis</b>	Ištrinti užduotį.
<b>Kelias (Route)</b>	api/views/{viewId}/columns/{columnId}/tasks/{taskId}
<b>Atsakymo struktūra</b>	
<b>Atsakymo kodas</b>	204 No Content
<b>Galimi klaidų kodai</b>	404 - jei užduoties Id negrąžina jokios reikšmės. 403 – trūksta teisių duomenims prieiti. 401 – neleidžiama prieiti prie duomenų.

## Auth API metodai

<b>API metodas</b>	Register() POST
<b>Paskirtis</b>	Prisiregistruoti
<b>Kelias (Route)</b>	api/register
<b>Atsakymo struktūra</b>	<pre>{   "id": "a6fbfe6c-4532-4a15-98b0-2529288585c2",   "userName": "domassss5",   "email": "domasaasddsa@gmail.com" }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	400 – Kai toks vartotojas egzistuoja arba nepavyksta sukurti vartotojo.

<b>API metodas</b>	Login() POST
<b>Paskirtis</b>	Prisijungti
<b>Kelias (Route)</b>	api/login
<b>Atsakymo struktūra</b>	<pre>{   "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJodHRwOi8vc2NoZW1hcy54bWxzb2FwLm9yZy93cy8yMDA1LzA1L2lkZW50aXR5L2NsYWltcy9uYW1lIjoiazG9tYXNzc3M1IiwianRpIjoia0GM2MwVkdMDAtZDJkYi00YjliLWI3ZDktNjQwMjc3YzY3NTJmIiwic3ViIjoiaYTZmYmZlNmMtNDUzMi00YTE1LTk4YjAtMjUyOTI4ODU4NWMyIiwiaHR0cDovL3NjaGVtYXMuYm91cm9zb2Z0LmNvbS93cy8yMDA4LzA2L2lkZW50aXR5L2NsYWltcy9yb2x1IjoiaS2FuYmFuVXNlciIsImV4cCI6MTcwMDY4NjA0NCwiaXNzIjoiaRG9tYXMiLCJhdWQiOiJ1UenVzdGVkQ2xpZW50In0.S4_zBDpuPpF10bPRlQKGlZjEmSKa0lT9yCdHIEQiD88",   "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJqdGkiOiJlMjJiODljNi01NDM2LTQ5ZDQtOGQzZS1lNmFjYTk2MmQxNjQiLCJzdWIiOiJhNmZiZmU2Yy00NTMyLTRhMTU0T0thMC0yNTI5Mjg4NTg1YzIiLCJleHAiOiJlMjJiODljNi01NDQsImV4cCI6MTcwMDY4NjA0NCwiaXNzIjoiaVhJ1c3RlZENsaWVudCJ9.XPwPnp47miJ9cN5dbskwpsjIGrK_DlLXuKak7QubdI" }</pre>
<b>Atsakymo kodas</b>	200 OK
<b>Galimi klaidų kodai</b>	400 – Kai slapavardis arba slaptažodis užimti.



## 4. Naudotojo sąsaja

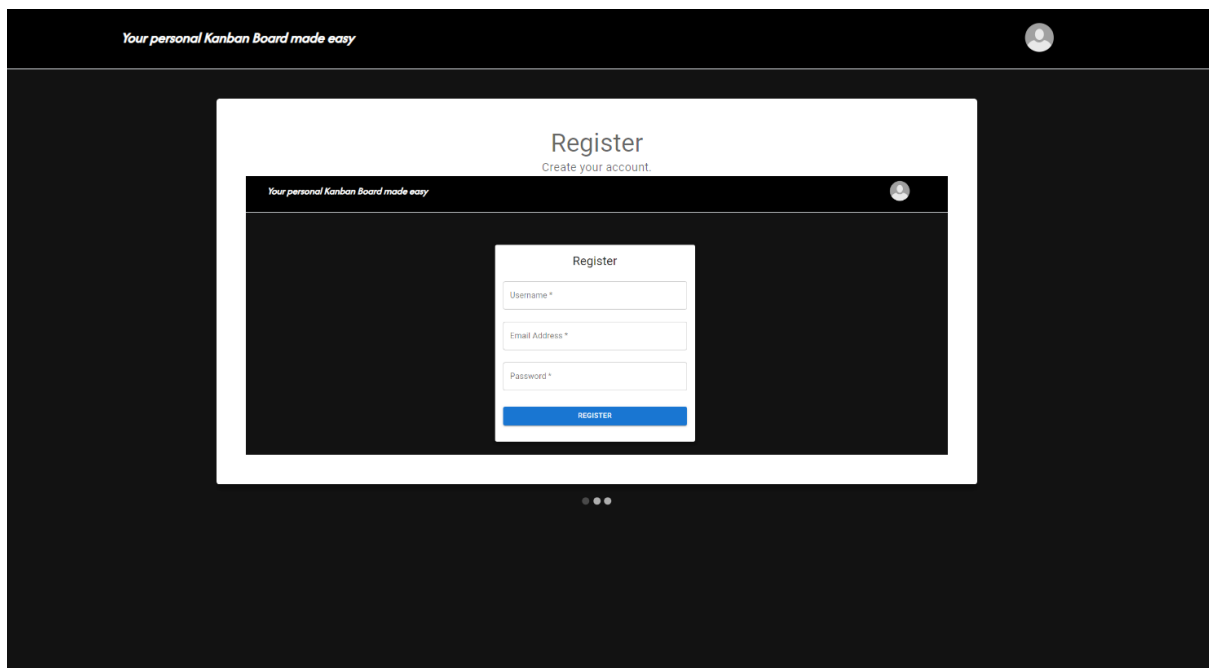
Naudotojo sąsaja yra sudaryta iš šių pagrindinių komponentų:

- Home puslapis.
- MainPage puslapis.
- Login puslapis.
- Register puslapis.

Toliau bus pateiktos ir aptariamos šių komponentų vizualizacijos.

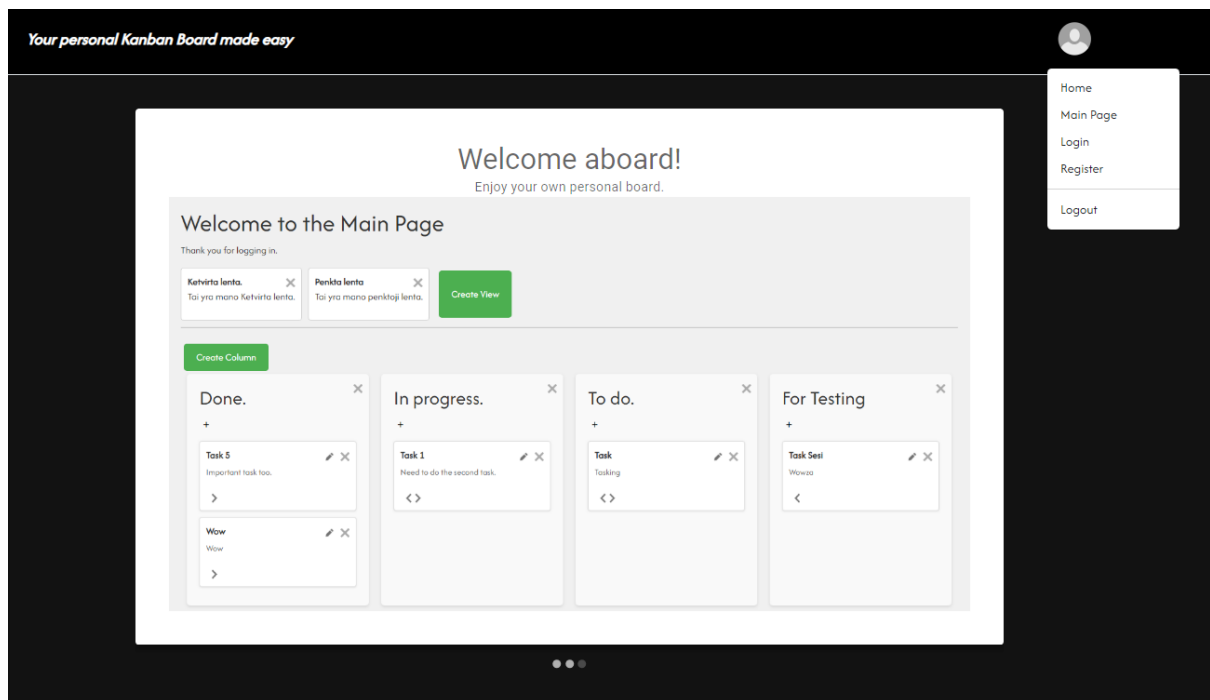
### 4.1 Home puslapis

Šį langą vartotojas mato atvykęs į puslapį. Pradiniame puslapyje matome skaidres, kurios vartotojui parodo registracijos ir prisijungimo procesą.



*Pav. 2 Home puslapis*

Vartotojas gali paspausti ant ikonėlės ir išsiskleidžiančiame meniu pasirinkti norimą veiksmą.

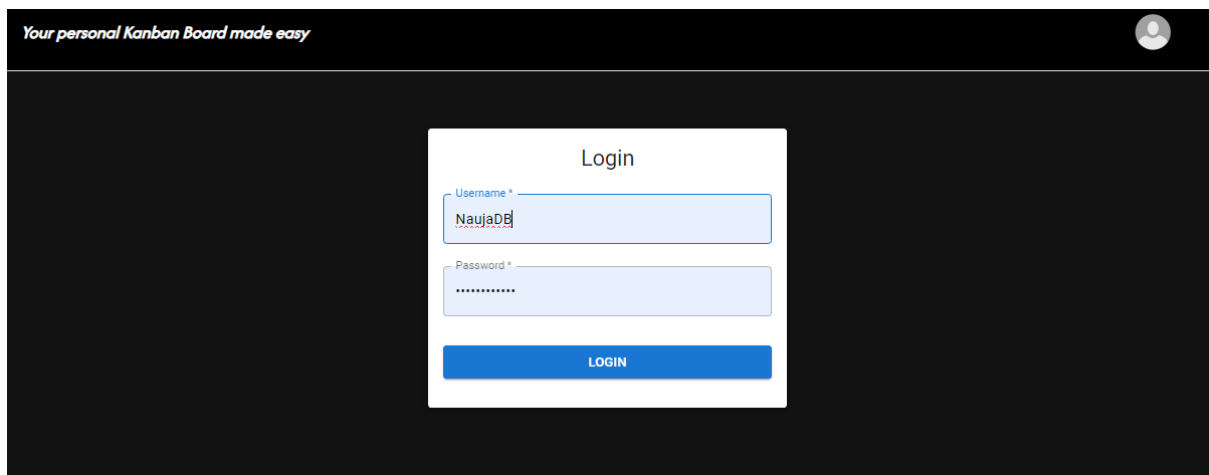


***Pav. 3 Išsiskleidžiantis meniu***



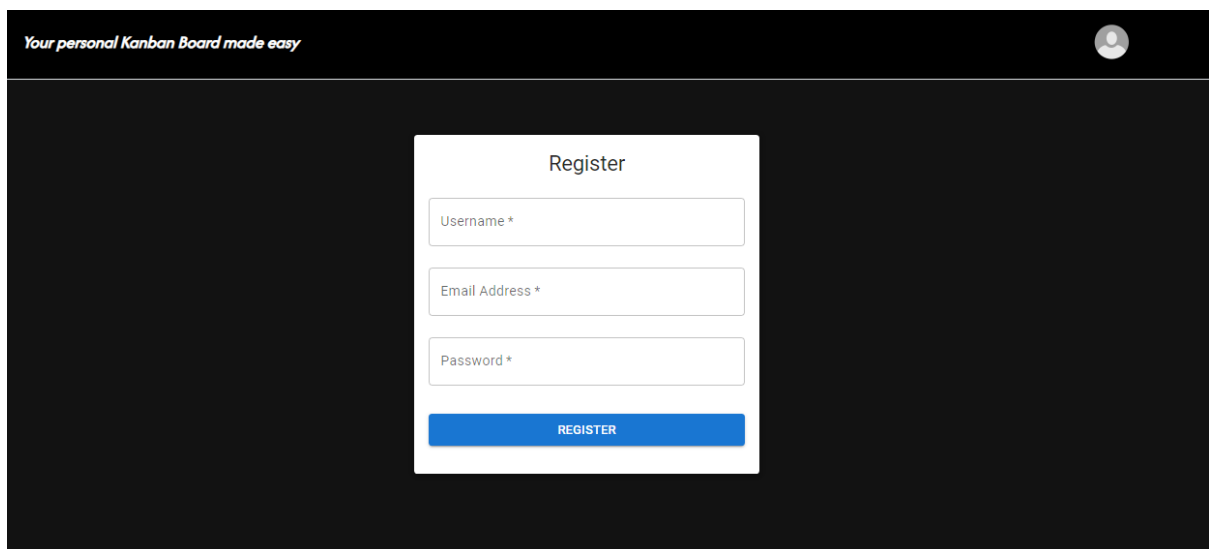
## 4.2 Autentifikacijos puslapiai

Turime du autentifikacijos puslapius – prisijungimo ir registracijos. Taip pat vartotojas turi galimybę ir atsijungti. Vartotojui prisijungus, jam yra išduodamas JWT žetonas ir naudojamas autorizavimui pagrindiniame puslapyje. Paspaudus atsijungimo mygtuką žetonas yra panaikinamas iš lokalsios saugyklos.



The screenshot shows a web application interface with a dark background. At the top left, the text "Your personal Kanban Board made easy" is displayed. At the top right, there is a user profile icon. In the center, there is a white "Login" form. The form contains two input fields: "Username \*" with the text "NaujaDB" entered, and "Password \*" with masked characters. Below the fields is a blue "LOGIN" button.

*Pav. 4 Prisijungimo puslapis*

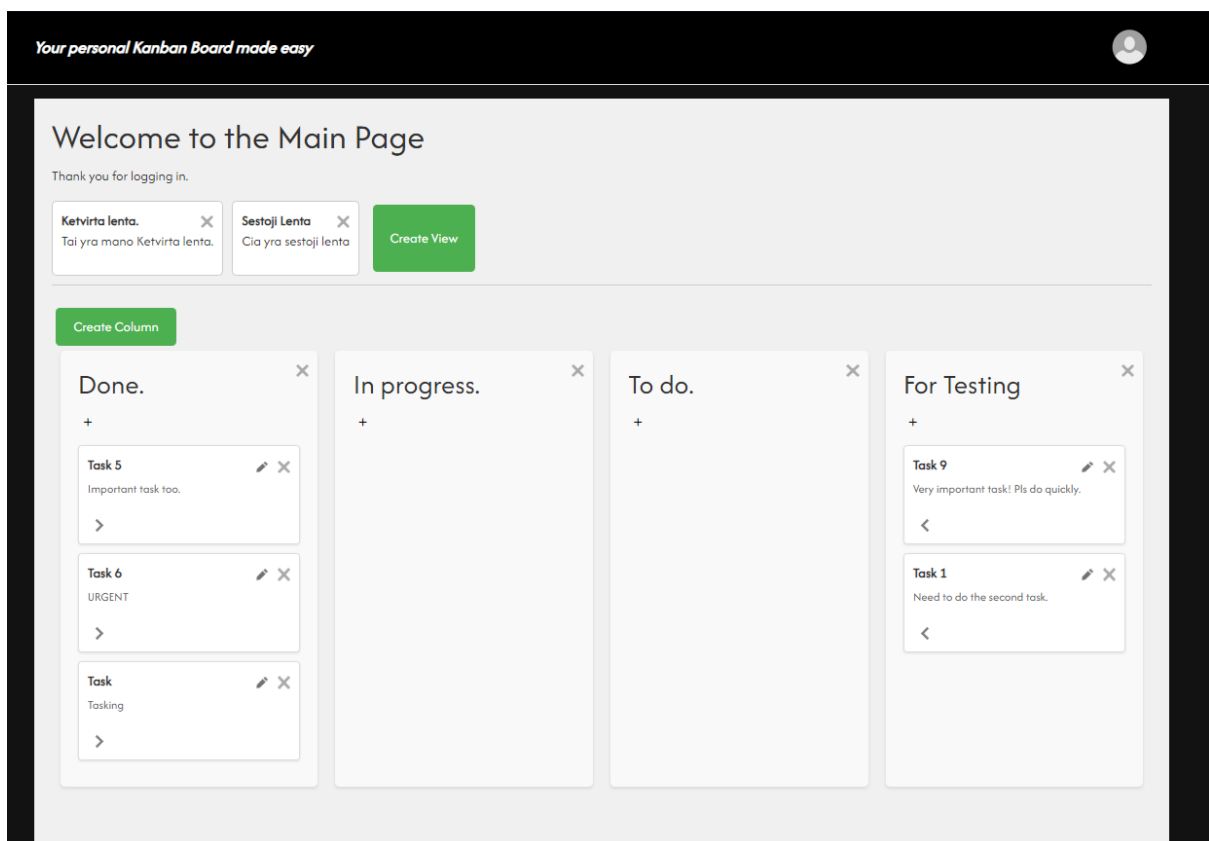


The screenshot shows a web application interface with a dark background. At the top left, the text "Your personal Kanban Board made easy" is displayed. At the top right, there is a user profile icon. In the center, there is a white "Register" form. The form contains three input fields: "Username \*", "Email Address \*", and "Password \*". Below the fields is a blue "REGISTER" button.

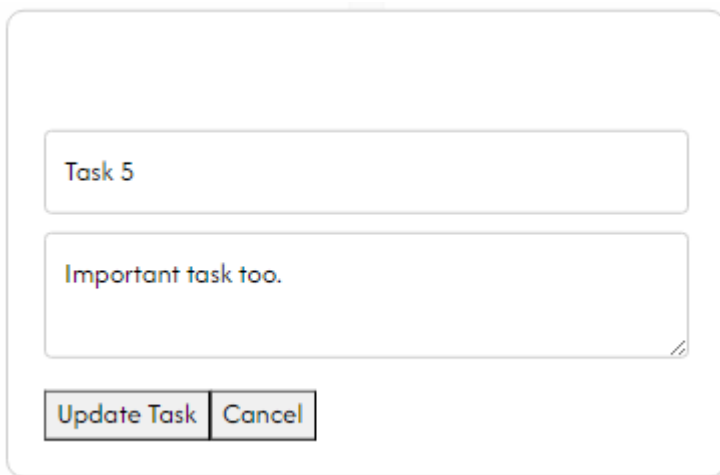
*Pav. 5 Registracijos puslapis*

## 4.3 Pagrindinis puslapis

Vartotojui prisijungus ir atvykus į šį puslapį yra paaimamos visos šiam vartotojui priklausančios lentelės su atitinkamais stulpeliais ir užduotimis. Vartotojas gali naikinti bei kurti lentas, naikinti ir kurti stulpelius bei galiausiai – naikinti, kurti, redaguoti užduotis. Užduotis galima judinti tarp stulpelių naudojant ant užduočių esančias rodykles. Taip pat prie šio puslapio prisijungus vartotojui su administratoriaus teisėmis, būtų rodomos visų vartotojų sukurtos lentos ir suteikiama galimybė jas redaguoti.



*Pav. 6 Pagrindinis puslapis*



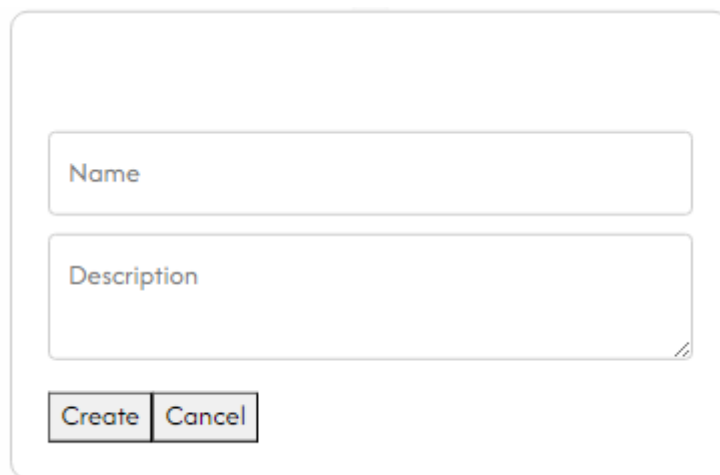
A modal dialog box with a light gray border and rounded corners. It contains two text input fields. The first field has the text "Task 5". The second field has the text "Important task too." and a small pencil icon in the bottom right corner. At the bottom, there are two buttons: "Update Task" and "Cancel".

Task 5

Important task too.

Update Task Cancel

*Pav. 7 Užduočių redagavimo modalinis langas.*



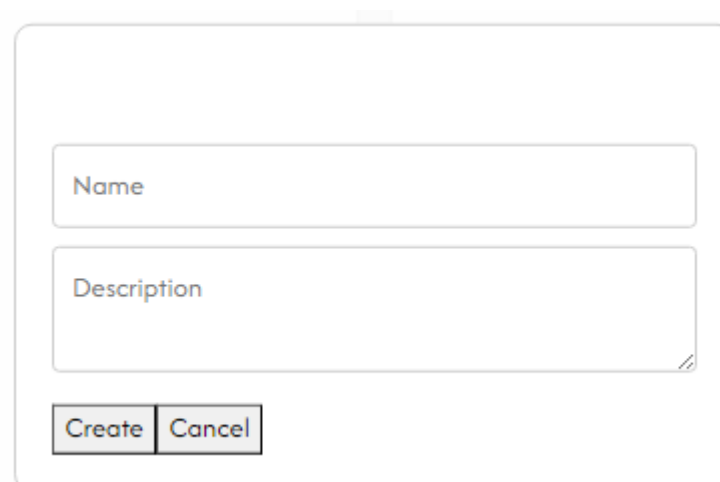
A modal dialog box with a light gray border and rounded corners. It contains two text input fields. The first field has the text "Name". The second field has the text "Description" and a small pencil icon in the bottom right corner. At the bottom, there are two buttons: "Create" and "Cancel".

Name

Description

Create Cancel

*Pav. 8 Stulpelių kūrimo modalinis langas.*



A modal dialog box with a light gray border and rounded corners. It contains two text input fields. The first field has the text "Name". The second field has the text "Description" and a small pencil icon in the bottom right corner. At the bottom, there are two buttons: "Create" and "Cancel".

Name

Description

Create Cancel

*Pav. 9 Lentos kūrimo modalinis langas.*

## 5. Išvados

Šį projektą atlikti buvo vertinga patirtis. Pagal tikslus, suprantamus pavyzdžius buvo įdomu atlikti projekto logiką, sukurti endpoint'us, turėti švarią struktūrą. Taip pat pirmą kartą naudojausi React vartotojo sąsajos įgyvendinimui ir tai turėjo savitų iššūkių. Tačiau pastebėjau, jog derėtų tiksliau susiplanuoti kurie endpoint'ai iš tikrųjų bus naudojami realioje web aplikacijoje. Galutiniame variante gavosi taip, jog ne visi endpoint'ai buvo panaudoti. Taip pat buvo įdomu pirmą kartą panaudoti JWT žetonus autentifikacijai ir autorizacijai įgyvendinti. Esu įpratęs, jog .NET Identity automatiškai sugeneruoja vartotojų roles ir reikalingas klases autentifikacijai, todėl didelę dalį autentifikacijos ir autorizacijos apsibrėžti pačiam kode buvo vertinga patirtis.