

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

T120B165 Saityno taikomųjų programų projektavimas
Projekto „PetHelp“ ataskaita

Atliko:

IFF-0/6 gr. studentas

Ostinas Mačiulis

2023 m. gruodžio 18 d.

Priėmė:

Dėst.Val. Baltulionis Simonas

Dėst.Val. Lukošius Tomas

KAUNAS 2023

TURINYS

1.	Sprendžiamo uždavinio aprašymas.....	3
1.1.	Sistemos paskirtis	3
1.2.	Funkciniai reikalavimai	3
2.	Sistemos architektūra	3
3.	Sistemos implementacija	5
3.1.	Sistemos struktūra	5
4.	Naudotojo sąsajos projektas ir realizacija.....	5
5.	API specifikacija.....	9
5.1.	OwnersController	9
5.2.	PetsController	11
5.3.	AdsController	14
5.4.	ApplicantsController	16
6.	Išvados	18

1. Sprendžiamo uždavinio aprašymas

1.1. Sistemos paskirtis

Sistema skirta suteikti galimybę gyvūnų savininkams atiduoti prižiūrėti savo augintinį, kol jie to negalės daryti.

Sistemą sudarys dvi dalys. Pirmoji dalis bus internetinė aplikacija, kuria naudosis visi sistemos naudotojai. Antroji dalis bus aplikacijų programavimo sąsaja (angl. trump. API)

Internetinėje aplikacijoje bus galima peržiūrėti visus esamus skelbimus, kuriuos įkelia prisiregistravę naudotojai. Norint įkelti skelbimą bus privaloma įkelti augintinio informaciją, nurodyti augintinio atidavimui susitikimo vietą bei laikotarpį, per kurį reikės prižiūrėti augintinį. Norint pretenduoti į skelbimą naudotojas turės pateikti savo el. paštą, aprašymą, telefono numerį ir nurodyti miestą.

1.2. Funkciniai reikalavimai

Svečias sistemoje galės:

1. Peržiūrėti galiojančius skelbimus
2. Peržiūrėti naudotojų augintinius
3. Prisijungti prie sistemos

Naudotojas galės:

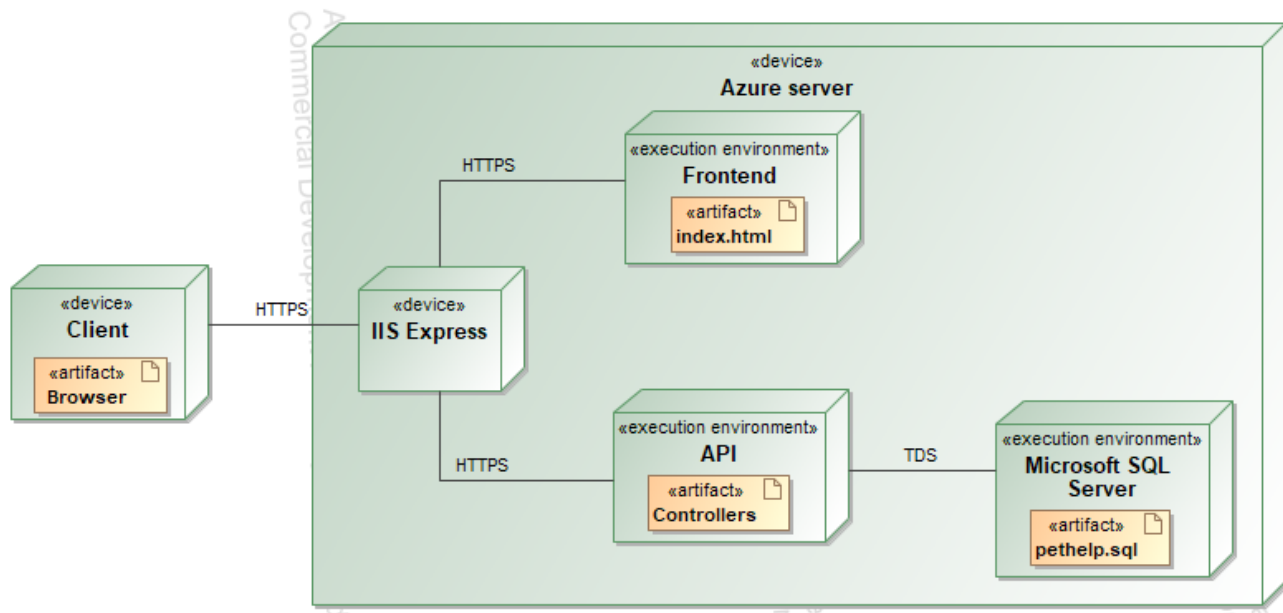
1. Atsijungti
2. Peržiūrėti bei redaguoti savo asmeninį profilį, pateikiant:
 - a. Savo aprašymą
 - b. El. paštą
 - c. Miestą, iš kurio yra
 - d. Telefono numerį
3. Pretenduoti į kitų naudotojų skelbimus
4. Pridėti savo augintinį, pateikiant:
 - a. Vardą
 - b. Amžių
 - c. Lytį
 - d. Aprašymą
5. Redaguoti augintinio informaciją
6. Pašalinti augintinį
7. Sukurti skelbimą pasirenkant:
 - a. Augintinį, kurį reikės prižiūrėti
 - b. Susitikimo vietą
 - c. Priežiūros laikotarpį
 - d. Galimą atlyginimą
8. Redaguoti arba pašalinti skelbimą
9. Peržiūrėti skelbimo aplikantus
10. Pasirinkti ir susisiekti su norimu aplikantu

2. Sistemos architektūra

Sistemą sudarys:

- Kliento pusė (angl. Frontend), sukurta naudojant ASP.NET Core BlazorWebAssembly
- Serverio pusė (angl. Backend), sukurta naudojant ASP.NET Core bei Microsoft SQL Server duomenų bazę

Žemiau pateiktame 1 pav. matome sistemos diegimo diagramą. Naudotojas per naršyklę HTTPS protokolu galės pasiekti sistemą, kuri bus talpinama Azure serveryje. Sistemos veikimui bus naudojamas API, kuris vykdys duomenų mainus su duomenų baze per Entity Framework ORM.



1 pav. Sistemos diegimo diagrama

3. Sistemos implementacija

3.1. Sistemos struktūra

Server pusė sudaryta iš AdsController, ApplicantsController, OwnersController ir PetsController, kurie atitinkamai atsakingi už augintinio priežiūros skelbimų, aplikantų į skelbimus, augintinių savininkų ir augintinių CRUD operacijas.

Autorizacijai ir autentifikacijai naudojami JWT. Prisijungiant prie sistemos serveris sugeneruoja žetoną, kuris tada yra saugomas kliento pusėje ir naudojamas atliekant API užklausas. Augintinių savininkai ir aplikantai turi savo atitinkamas roles, kurios yra saugomos žetone. Dėl JWT technologijos panaudojimo tam tikri veiksmai yra neleidžiami savininkui ar aplikantui. Pavyzdžiui, aplikantas negali sukurti skelbimo augintiniui, negali pašalinti augintinių.

4. Naudotojo sąsajos projektas ir realizacija

The diagram shows a web page layout for a login form. At the top is a navigation bar with the text 'PetHelp' and links 'Home', 'Ads', and 'Applicants'. On the right side of the navigation bar is a circular profile icon. The main content area contains a 'Login' form with three input fields: 'Email input', 'Password input', and a 'Login' button. At the bottom right of the page is a 'Footer text' label.

2 pav. Planuota prisijungimo lango struktūra

The screenshot shows the implemented login page. The navigation bar is teal with 'PetHelp' and links 'HOME', 'BROWSE ADS', and 'MY APPLICANTS'. A circular profile icon is on the right. The login form is white with a teal border. It has a 'Login' title, 'Email*' and 'Password*' labels, and corresponding input fields. A teal 'LOGIN' button is at the bottom. The footer is teal with the text '© 2023 Oskinas Mačiulis'.

3 pav. Realizuotas prisijungimo langas

PetHelp

Login

Email input

Password input

Login

Footer text

4 pav. Planuota prisijungimo esant mažai rezoliucijai

PetHelp

Login

Email*

Password*

LOGIN

© 2023 Ostinas Mačiulis

5 pav. Realizuotas prisijungimo langas esant mažai rezoliucijai

PetHelp
Home
Ads
Applicants

My pets

Teddy

Nice description

Male

10 years old

Teddy

Nice description

Male

10 years old

Other pets

Teddy

Nice description

Male

10 years old

Teddy

Nice description

Male

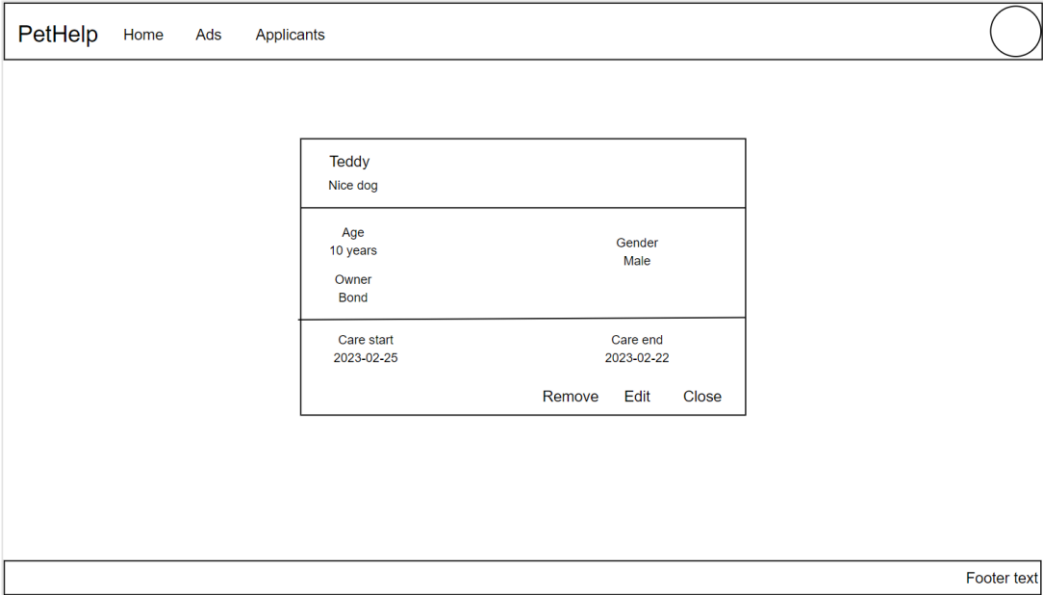
10 years old

Footer text

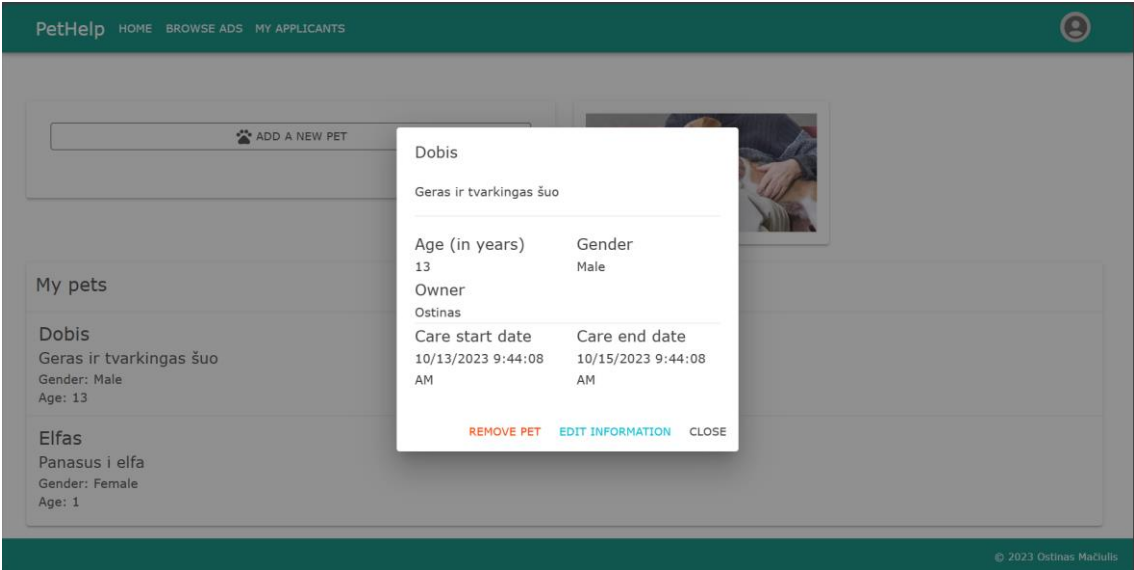
6 pav. Planuota pagrindinio lango struktūra



7 pav. Realizuota pagrindinis langas



8 pav. Planuota augintinio peržiūros lango struktūra



9 pav. Realizuotas augintinio peržiūros langas

PetHelp
Home
Ads
Applicants

Add new pet

Pet name input

Pet description input

Age input

Gender select list

Add new pet

Footer text

10 pav. Planuota augintinio pridėjimo lango struktūra

PetHelp
HOME
BROWSE ADS
MY APPLICANTS

Add a new pet

Name*
0 / 25

Description
0 / 2000

Age of the pet in years*
0

ADD NEW PET

© 2023 Ostinas MacKulls

11 pav. Realizuotas augintinio pridėjimo langas

PetHelp
Home
Ads
Applicants

Create new ad

Pet name

Pet description

Pet age

Pet gender

Pay for care input

Start date of care input

End date of care input

Create new ad

Footer text

12 pav. Planuota skelbimo sukūrimo lango struktūra

PetHelp
HOME
BROWSE ADS
MY APPLICANTS

Create new ad for Dobis

Pet description
Geras ir tvarkingas šuo

Age
13

Gender
Male

Meeting address*

Pay
€ 0.00

Start date of needed care
12/17/2023

End date of needed care
12/18/2023

CREATE

© 2023 Ostinas Mačiulis

13 pav. Realizuotas augintinio pridėjimo langas

5. API specifikacija

Visi sistemos API metodai pateikiami lentelėse, kurias sudaro metodo tipas, paskirtis, kelias iki jo, reikiama JSON formatu struktūra, kad būtų sėkmingai atlikta užklausa, ar naudotojas turi būti autentifikuotas ir turėti autorizaciją metodui atlikti, bei užklaustos atsakymo struktūra su galimais klaidų kodais.

5.1. OwnersController

1 lentelė. Owners GET specifikacija

API metodas	GET
Paskirtis	Gauti visų augintinių savininkų sąrašą.
Kelias iki metodo (angl. route)	/api/owners
Užklaustos struktūra	-
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	[{ "id": 1, "name": "string", }]

	<pre> "email": "string", "password": "string", "phoneNumber": "string", "address": "string", "city": "string" }] </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerasta savininkų duomenų bazėje. 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

2 lentelė. Owners GET specifikacija

API metodas	GET
Paskirtis	Gauti nurodytą augintinių savininką.
Kelias iki metodo (angl. route)	/api/owners/{ownerId}
Užklauso struktūra	-
Užklauso „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": 1, "name": "string", "email": "string", "password": "string", "phoneNumber": "string", "address": "string", "city": "string" } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei savininkas nerastas duomenų bazėje. 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

3 lentelė. Owners PUT specifikacija

API metodas	PUT
Paskirtis	Atnaujinti augintinių savininko informaciją.
Kelias iki metodo (angl. route)	/api/owners/{ownerId}
Užklauso struktūra	<pre> { "id": 1, "name": "string", "email": "string", "password": "string", "phoneNumber": "string", "address": "string", "city": "string" } </pre>
Užklauso „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": 1, "name": "string", "email": "string", "password": "string", "phoneNumber": "string", "address": "string", "city": "string" } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei savininkas nerastas duomenų bazėje.

	403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 400 – jei nurodomi neteisingu formatu užklausos duomenys
--	--

4 lentelė. Owners POST specifikacija

API metodas	POST
Paskirtis	Sukurti naują augintinių savininką.
Kelias iki metodo (angl. route)	/api/owners/{ownerId}
Užklausos struktūra	{ "name": "string", "email": "string", "password": "string", "phoneNumber": "string", "address": "string", "city": "string" }
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	{ "id": 1, "name": "string", "email": "string", "password": "string", "phoneNumber": "string", "address": "string", "city": "string" }
Atsakymo kodas	201 (Created)
Galimi klaidų kodai	403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 400 – jei neteisingu formatu nurodomi užklausos duomenys

5 lentelė. Owners DELETE specifikacija

API metodas	DELETE
Paskirtis	Pašalinti augintinių savininką.
Kelias iki metodo (angl. route)	/api/owners/{ownerId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	204 (No content)
Galimi klaidų kodai	500 – jei įvyko klaida šalinant savininką iš duomenų bazės 404 – jei savininkas nerastas duomenų bazėje. 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

5.2. PetsController

6 lentelė. Pets GET specifikacija

API metodas	GET
Paskirtis	Gauti augintinių sąrašą.
Kelias iki metodo (angl. route)	/api/pets
Užklausos struktūra	-
Užklausos „Header“ dalis	-
Atsakymo struktūra	[

	<pre>{ "id": 1, "name": "string", "email": "string", "password": "string", "phoneNumber": "string", "address": "string", "city": "string" }</pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerasta augintinių duomenų bazėje.

7 lentelė. Pets GET specifikacija

API metodas	GET
Paskirtis	Gauti nurodytą augintinį.
Kelias iki metodo (angl. route)	/api/pets/{petId}
Užklausos struktūra	-
Užklausos „Header“ dalis	-
Atsakymo struktūra	<pre>{ "id": 0, "name": "string", "description": "string", "sex": "string", "age": 0, "ownerName": "string", "ownerId": 0, "careStart": "2023-12-18T17:35:10.417Z", "careEnd": "2023-12-18T17:35:10.417Z" }</pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerastas augintinis duomenų bazėje.

8 lentelė. Pets PUT specifikacija

API metodas	PUT
Paskirtis	Atnaujinti nurodyto augintinio informaciją.
Kelias iki metodo (angl. route)	/api/pets{petId}
Užklausos struktūra	<pre>{ "name": "string", "description": "string", "sex": "string", "age": 0, "ownerName": "string", "ownerId": 0, "careStart": "2023-12-18T17:35:10.417Z", "careEnd": "2023-12-18T17:35:10.417Z" }</pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>{ "id": 0, "name": "string", "description": "string", "sex": "string", "age": 0, "ownerName": "string", </pre>

	<pre>"ownerId": 0, "careStart": "2023-12-18T17:35:10.417Z", "careEnd": "2023-12-18T17:35:10.417Z" }</pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerastas augintinis duomenų bazėje. 400 – jei neteisingu formatu nurodomi užklausos duomenys

9 lentelė. Pets POST specifikacija

API metodas	POST
Paskirtis	Sukurti naują augintinio įrašą duomenų bazėje.
Kelias iki metodo (angl. route)	/api/pets/
Užklausos struktūra	<pre>{ "name": "string", "description": "string", "sex": "string", "age": 0, "ownerName": "string", "ownerId": 0, "careStart": "2023-12-18T17:35:10.417Z", "careEnd": "2023-12-18T17:35:10.417Z" }</pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>{ "id": 0, "name": "string", "description": "string", "sex": "string", "age": 0, "ownerName": "string", "ownerId": 0, "careStart": "2023-12-18T17:35:10.417Z", "careEnd": "2023-12-18T17:35:10.417Z" }</pre>
Atsakymo kodas	201 (Created)
Galimi klaidų kodai	403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 400 – jei neteisingu formatu nurodomi užklausos duomenys

10 lentelė. Pets DELETE specifikacija

API metodas	DELETE
Paskirtis	Pašalinti nurodyto augintinio įrašą duomenų bazėje.
Kelias iki metodo (angl. route)	/api/pets/{petId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	204 (No content)
Galimi klaidų kodai	500 – jei įvyko klaida šalinant augintinį duomenų bazėje 404 – jei augintinis nerastas duomenų bazėje 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

5.3. AdsController

11 lentelė. Ads GET specifikacija

API metodas	GET
Paskirtis	Gauti nurodyto augintinio skelbimų sąrašą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads
Užklauskos struktūra	-
Užklauskos „Header“ dalis	-
Atsakymo struktūra	[<pre> { "id": 0, "meetingAddress": "string", "careStart": "2023-12-18T17:44:45.986Z", "careEnd": "2023-12-18T17:44:45.986Z", "pay": 0, "ownerName": "string", "petName": "string", "petId": 0 }] </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerastas augintinis duomenų bazėje.

12 lentelė. Ads GET specifikacija

API metodas	GET
Paskirtis	Gauti nurodytą augintinio skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads
Užklauskos struktūra	-
Užklauskos „Header“ dalis	-
Atsakymo struktūra	{ <pre> { "id": 0, "meetingAddress": "string", "careStart": "2023-12-18T17:44:45.986Z", "careEnd": "2023-12-18T17:44:45.986Z", "pay": 0, "ownerName": "string", "petName": "string", "petId": 0 } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerastas augintinis duomenų bazėje.

13 lentelė. Ads PUT specifikacija

API metodas	PUT
Paskirtis	Atnaujinti nurodytą augintinio skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads/{adId}
Užklauskos struktūra	{ <pre> { "meetingAddress": "string", "careStart": "2023-12-18T17:44:45.986Z", "careEnd": "2023-12-18T17:44:45.986Z", "pay": 0, } </pre>
Užklauskos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	{

	<pre> "id": 0, "meetingAddress": "string", "careStart": "2023-12-18T17:44:45.986Z", "careEnd": "2023-12-18T17:44:45.986Z", "pay": 0, "ownerName": "string", "petName": "string", "petId": 0 } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerastas augintinis duomenų bazėje. 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 400 – jei nurodomi neteisingu formatu užklausos duomenys

14 lentelė. Ads POST specifikacija

API metodas	POST
Paskirtis	Sukurti nurodytam augintiniui skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads
Užklausos struktūra	<pre> { "meetingAddress": "string", "careStart": "2023-12-18T17:44:45.986Z", "careEnd": "2023-12-18T17:44:45.986Z", "pay": 0, } </pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": 0, "meetingAddress": "string", "careStart": "2023-12-18T17:44:45.986Z", "careEnd": "2023-12-18T17:44:45.986Z", "pay": 0, "ownerName": "string", "petName": "string", "petId": 0 } </pre>
Atsakymo kodas	201 (Created)
Galimi klaidų kodai	404 – jei nerastas augintinis duomenų bazėje. 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 400 – jei nurodomi neteisingu formatu užklausos duomenys

15 lentelė. Ads DELETE specifikacija

API metodas	DELETE
Paskirtis	Pašalinti augintinio skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads/{adId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	204 (No content)
Galimi klaidų kodai	500 – jei įvyko klaida šalinant skelbimą duomenų bazėje 404 – jei nerastas augintinis arba skelbimas duomenų bazėje. 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

5.4. ApplicantsController

16 lentelė. Applicants GET specifikacija

API metodas	GET
Paskirtis	Gauti aplikantų sąrašą į nurodytą augintinio skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads/applicants
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	[<pre> { "id": 0, "name": "string", "description": "string", "email": "string", "phoneNumber": "string", "address": "string", "city": "string", "petId": 0, "petName": "string", "adId": 0 }] </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerasta aplikantų duomenų bazėje.

17 lentelė. Applicants GET specifikacija

API metodas	GET
Paskirtis	Gauti aplikantą į nurodytą augintinio skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads/applicants/{applicantId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	{ <pre> { "id": 0, "name": "string", "description": "string", "email": "string", "phoneNumber": "string", "address": "string", "city": "string", "petId": 0, "petName": "string", "adId": 0 } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerastas aplikantas duomenų bazėje.

18 lentelė. Applicants PUT specifikacija

API metodas	PUT
Paskirtis	Aplikantui atnaujinti savo pateiktą informaciją į skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads/applicants/{applicantId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	{ <pre> { "name": "string", } </pre>

	<pre> "description": "string", "email": "string", "phoneNumber": "string", "address": "string", "city": "string", } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	404 – jei nerastas aplikantas duomenų bazėje. 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 400 – jei nurodomi neteisingu formatu užklausos duomenys

19 lentelė. Applicants POST specifikacija

API metodas	POST
Paskirtis	Aplikantui pretenduoti į skelbimą.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads/applicants
Užklausos struktūra	<pre> { "name": "string", "description": "string", "email": "string", "phoneNumber": "string", "address": "string", "city": "string", } </pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "name": "string", "description": "string", "email": "string", "phoneNumber": "string", "address": "string", "city": "string", } </pre>
Atsakymo kodas	201 (Created)
Galimi klaidų kodai	404 – jei nerastas aplikantas duomenų bazėje. 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 400 – jei nurodomi neteisingu formatu užklausos duomenys

20 lentelė. Applicants DELETE specifikacija

API metodas	DELETE
Paskirtis	Pašalinti aplikantą iš skelbimo.
Kelias iki metodo (angl. route)	/api/pets/{petId}/ads/applicants/{applicantId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	204 (No content)
Galimi klaidų kodai	500 – jei įvyko klaida šalinant skelbimą duomenų bazėje 404 – jei nerastas aplikantas duomenų bazėje. 403 – jei naudotojas neturi teisės atlikti šį veiksmą 401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

6. Išvados

Skurta ir debesyje sėkmingai patalpinta augintinių priežiūros skelbimų svetainė, kuri išpildo nurodytus funkcinius reikalavimus. Darbo įgyvendinimui buvo pasitelkta C# programavimo kalba su .NET 8 karkasu. Naudojant šiuos įrankius buvo sukurti WebApi, Blazor WebAssembly ir Class library projektai, kurie leido atskirti verslo logiką nuo atvaizdavimo logikos ir išvengti priklausomybių tarp projektų turint atskirą projektą modeliams. Taip pat papildomai pasinaudota Entity Framework paprastesniam duomenų paėmimui iš Azure SQL Server duombazės ir MudBlazor biblioteka UI komponentams.

Nuoroda į projekto Github repozitoriją: <https://github.com/Ostinas/PetHelp>