

INFORMATIKOS FAKULTETAS

T120B165 Saityno taikomųjų programų projektavimas

Projekto „GURPS Helper“ ataskaita

Studentas:

Jūnius Vaitkus, IFF-0/5

Dėstytojai:

Tomas Blažauskas

Eligijus Kliudys

KAUNAS 2023

Turinys

1.	Sprendžiamo uždavinio aprašymas	3
1.1.	Sistemos paskirtis	3
1.2.	Funkciniai reikalavimai	3
2.	Sistemos architektūra	4
3.	API specifikacija	4
3.1.	Žaidimų (angl. Campaigns) API metodai	5
3.2.	Veikėjų API metodai	10
3.3.	Veikėjų įgūdžių API metodai	14
3.4.	Veikėjų bruožų API metodai	19
3.5.	Naudotojų API metodai	24
3.6.	Autentifikacijos API metodai	25
4.	Sistemos naudotojo sąsaja	27
4.1.	Sistemos internetinė aplikacija	27
5.	Išvados.....	41

1. Sprendžiamo uždavinio aprašymas

1.1. Sistemos paskirtis

Projekto tikslas – palengvinti GURPS žaidimui reikalingų elementų sukūrimą ir valdymą žaidėjams ir organizatoriams.

Veikimo principas - kuriamą platformą sudaro dvi dalys: internetinė aplikacija, kuria naudosis žaidėjai, administratorius bei aplikacijų programavimo sąsają (angl. API).

Žaidėjas, norėdamas naudotis šia platforma, prisiregistruos prie internetinės aplikacijos ir galės kurti savo veikėjus, jų bruožais ir įgūdžiais. Jo veikėjus galės matyti kiti žaidėjai, kai veikėjas bus pridėti prie sukurtų žaidimų. Kuriant žaidimus, tokiu pat principu žaidimo organizatorius galės kurti ne žaidėjų veikėjus (angl. NPC). Administratorius valdo užsiregistravusius vartotojus, ištrindamas juos arba suteikdamas kitą rolę, šalina netinkamus žaidimus ar veikėjus.

1.2. Funkciniai reikalavimai

Neregistruotas sistemos naudotojas galės:

1. Užsiregistruoti prie svetainės.

Registruotas sistemos vartotojas (žaidėjas) galės:

1. Atsijungti nuo internetinės aplikacijos;
2. Sukurti naują veikėją;
3. Sukurti naują bruožą;
4. Sukurti naują įgūdį;
5. Prisijungti prie organizuojamo žaidimo;

Registruotas sistemos vartotojas (organizatorius) galės:

1. Atsijungti nuo internetinės aplikacijos;
2. Sukurti naują žaidimą;
3. Sukurti naują veikėją;
4. Sukurti naują bruožą;
5. Sukurti naują įgūdį;
6. Valdyti veikėjus savo žaidime.

Administratorius galės:

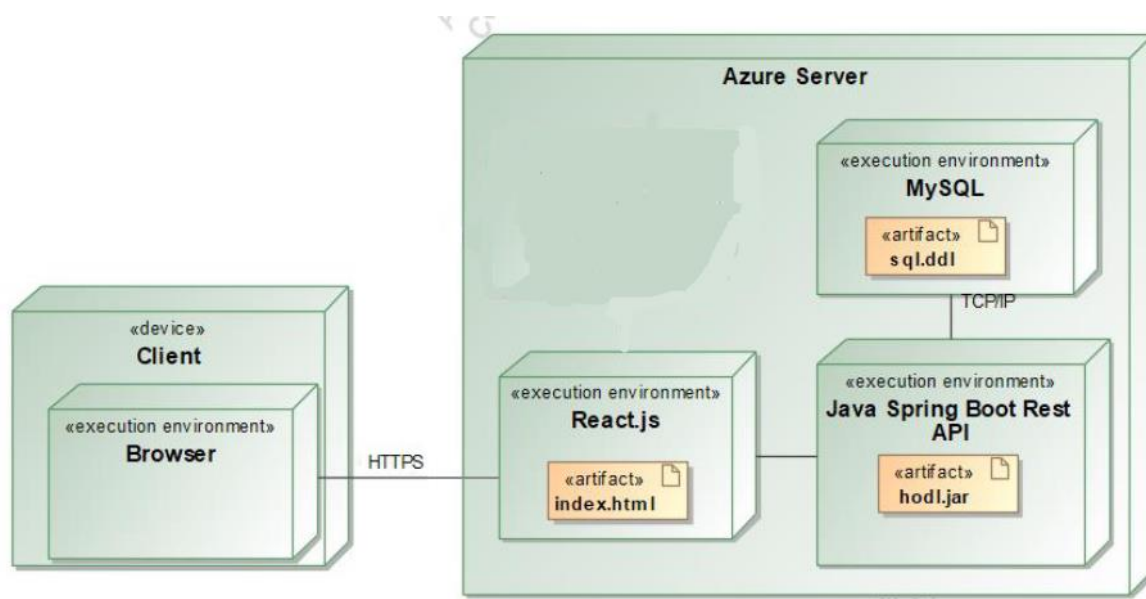
1. Šalinti naudotojus, redaguoti jų rolę;
2. Šalinti ir redaguoti netinkamus bruožus;
3. Šalinti ir redaguoti netinkamus įgūdžius;
4. Šalinti ir redaguoti netinkamus veikėjus;
5. Šalinti ir redaguoti netinkamus žaidimus.

2. Sistemos architektūra

Sistemos sudedamosios dalys:

- Kliento pusė (angl. Front-End) – naudojant React.js;
- Serverio pusė (angl. Back-End) – naudojant Java Spring Boot Rest API. Duomenų bazė – MySQL.

2.1 pav. vaizduojama kuriamos sistemos diegimo diagrama.. Sistema yra patalpinta Azure serveryje. Internetinė aplikacija yra pasiekiamą per HTTPS protokolą. Sistema manipuluojama per aplikacijų programavimo sąsają.



pav. 1 sistemos diegimo diagrama

3. API specifikacija

Tam, kad būtų išlaikomas nuoseklumas, sistemos API metodai bus aprašomi lentelėmis (žr. 1 lentelę). Jose pateikiama svarbiausia bei būtina informacija sėkmingam užklauskos vykdymui. Taip pat pateikiama informacija apie galimas klaidas, kurios gali iškilti apdorojant

tam tikrą API užklausą. Verta paminėti, kad kelias iki metodo yra nurodomas tik parašant URL galutinę dalį, nes domeną, akivaizdų, gali skirtis.

1 lentelė. API specifikacijos šablonas

API metodas	
Paskirtis	
Kelias iki metodo (angl. route)	
Užklausos struktūra	
Užklausos „Header“ dalis	
Atsakymo struktūra	
Atsakymo kodas	
Galimi klaidų kodai	

3.1. Žaidimų (angl. Campaigns) API metodai

2 lentelė. Campaigns index (GET) specifikacija

API metodas	Campaigns index (GET)
Paskirtis	Gauti visų žaidimų sąrašą.
Kelias iki metodo (angl. route)	/api/campaign
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>[{ "id": ..., "name": "...", "ptsLimit": ..., "characters": [{ "id": ..., "first_name": "...", "last_name": "...", "photo": "...", "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": {</pre>

	<pre> "id": ..., "username": "...", "email": "...", "password": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } }, ...], ... }] </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

3 lentelė. Campaigns get (GET) specifikacija

API metodas	Campaigns get (GET)
Paskirtis	Gauti vieno žaidimo informaciją.
Kelias iki metodo (angl. route)	/api/Campaigns/{campaignId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "ptsLimit": ..., "characters": [{ "id": ..., "first_name": "...", "last_name": "...", "photo": "...", "strength": ..., "dex": ..., "intelligence": ..., "health": ..., </pre>

	<pre> "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "password": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 404 – URL nurodytas žaidimas neegzistuoja.

4 lentelė. Campaigns add (POST) specifikacija

API metodas	Campaigns add (PUT)
Paskirtis	Pridėti naują žaidimą.
Kelias iki metodo (angl. route)	/api/Campaigns/
Užklauskos struktūra	<pre> { "name": "...", "ptsLimit": ... } </pre>
Užklauskos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "ptsLimit": ..., "characters": [{ "id": ..., "first_name": "...", "last_name": "..." }] } </pre>

	<pre> "photo": "...", "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "password": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } </pre>
Atsakymo kodas	201 (Created)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 422 – jei netinkamai suformuluota užklausa.

5 lentelė. Campaigns update (PUT) specifikacija

API metodas	Campaigns update (PUT)
Paskirtis	Pakeisti žaidimo informaciją.
Kelias iki metodo (angl. route)	/api/Campaigns/{campaignId}
Užklaustos struktūra	<pre> { "name": "...", "ptsLimit": ... } </pre>
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "ptsLimit": ..., </pre>

	<pre> "characters": [{ "id": ..., "first_name": "...", "last_name": "...", "photo": "...", "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "password": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } }] </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 422 – jei netinkamai suformuluota užklausa.

6 lentelė. Campaigns delete (DELETE) specifikacija

API metodas	Campaigns delete (DELETE)
Paskirtis	Ištrinti žaidimą
Kelias iki metodo (angl. route)	/api/Campaigns/{campaignId}
Užklaustos struktūra	-
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	204 (No Content)

Galimi klaidų kodai	<p>403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, jei žaidimas nepriklauso naudotojui.</p> <p>404 – URL nurodytas žaidimas neegzistuoja.</p>
---------------------	---

3.2. Veikėjų API metodai

7 lentelė. Characters index (GET) specifikacija

API metodas	Characters index (GET)
Paskirtis	Gauti veikėjų, esančių žaidime sąrašą.
Kelias iki metodo (angl. route)	/api/Campaigns/{campaignId}/Characters
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>[{ "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } }, ...]</pre>

Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

8 lentelė. Characters put (PUT) specifikacija

API metodas	Characters update (PUT)
Paskirtis	Pakeisti veikėjo informaciją.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}
Užklausos struktūra	<pre>{ "first_name": "...", "last_name": "...", "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden":... }</pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>{ "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [</pre>

	<pre> { "authority": "...", }], "credentialsNonExpired": ..., "accountNonExpired": ... } } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, jei negalima prieiga dėl rolių trūkumo. 422 – jei netinkamai suformuluota užklausa.

9 lentelė. Characters create (POST) specifikacija

API metodas	Characters create (POST)
Paskirtis	Sukurti naują veikėją.
Kelias iki metodo (angl. route)	/api/Campaigns/{campaignId}/Characters
Užklausos struktūra	<pre> { "first_name": "...", "last_name": "...", "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden":... } </pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., } </pre>

	<pre> "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } </pre>
Atsakymo kodas	201 (Created)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 422 – jei netinkamai suformuluota užklausa.

10 lentelė. Characters get (GET) specifikacija

API metodas	Characters get (GET)
Paskirtis	Gauti konkretaus veikėjo informaciją.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}
Užklaustos struktūra	-
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { </pre>

	<pre> "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	<p>403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, negalima prieiga dėl rolės.</p> <p>404 – Veikėjas nerastas.</p>

3.3. Veikėjų įgūdžių API metodai

11 lentelė. Character skills index (GET) specifikacija

API metodas	Character skills index (GET)
Paskirtis	Gauti veikėjo įgūdžių sąrašą.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/skills
Užklaustos struktūra	-
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> [{ "id": ..., "name": "...", "description": "...", "level": ..., "difficulty": "...", "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., </pre>

	<pre> "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } , ...] </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

12 lentelė. Character skill get (GET) specifikacija

API metodas	Character skills get (GET)
Paskirtis	Gauti veikėjo vieno įgūdžio informaciją.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/skills/{skillId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "description": "...", "level": ..., "difficulty": "...", "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., </pre>

	<pre> "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas. 404 – Jei veikėjas neegzistuoja.

13 lentelė. Character skills update (PUT) specifikacija

API metodas	Character skills update (PUT)
Paskirtis	Pakeisti vieno įgūdžio informaciją.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/skills/{skillId}
Užklausos struktūra	<pre> { "name": "...", "difficulty": "...", "description": "...", "level": "..." } </pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "description": "...", </pre>

	<pre> "level": ..., "difficulty": "...", "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, neužtenkamos privilegijos. 422 – netinkamai suformuluota užklausa.

14 lentelė. Character skills update (POST) specifikacija

API metodas	Character skills add (POST)
Paskirtis	Pridėti naują įgūdį.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/skills/
Užklaustos struktūra	<pre> { "name": "...", "difficulty": "...", </pre>

	<pre> "description": "...", "level": "..." } </pre>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "description": "...", "level": ..., "difficulty": "...", "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } } </pre>
Atsakymo kodas	201 (Created)
Galimi klaidų kodai	<p>403 – JWT žetonas netinkamas arba prieiga neleistina, neužtenkamos privilegijos.</p> <p>422 – netinkamai suformuluota užklausa.</p>

15 lentelė. Character skills delete (DELETE) specifikacija

API metodas	Character skills delete (DELETE)
Paskirtis	Pašalinti veikėjo įgūdį.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/skills/{skillId}
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	204 (No Content)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, neleidžiama prieiga prie veikėjo.

3.4. Veikėjų bruožų API metodai

16 lentelė. Character traits index (GET) specifikacija

API metodas	Character traits index (GET)
Paskirtis	Gauti veikėjo bruožų sąrašą.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/traits
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>[{ "id": ..., "name": "...", "description": "...", "price": ..., "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...",</pre>

	<pre> "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } ...]</pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	<p>401 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas neužtenkamos privilegijos.</p> <p>404 – URL nurodytas užsakymas neegzistuoja.</p>

17 lentelė. Character trait get (GET) specifikacija

API metodas	Character traits get (GET)
Paskirtis	Gauti veikėjo vieno bruožo informaciją.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/traits/{traitId}
Užklaustos struktūra	-
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "description": "...", "price": ..., "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., } }</pre>

	<pre> "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	<p>403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.</p> <p>404 – URL nurodytas užsakymas arba užsakymo prekė neegzistuoja.</p>

18 lentelė. Character traits update (POST) specifikacija

API metodas	Character traits add (POST)
Paskirtis	Pridėti naują bruožą.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/traits/
Užklaustos struktūra	<pre> { "name": "...", "price": ..., "description": "..." } </pre>
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", "description": "...", "price": ..., "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., </pre>

	<pre> "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } </pre>
Atsakymo kodas	201 (CREATED)
Galimi klaidų kodai	<p>403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, negalima prieiga prie duomenų.</p> <p>422 – netinkamai suformuluota užklausa.</p>

19 lentelė. Character traits update (PUT) specifikacija

API metodas	Character traits update (PUT)
Paskirtis	Pakeisti vieno bruožo informaciją.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/traits/{traitId}
Užklaustos struktūra	<pre> { "name": "...", "price": ..., "description": "..." } </pre>
Užklaustos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre> { "id": ..., "name": "...", </pre>

	<pre> "description": "...", "price": ..., "character": { "id": ..., "first_name": "...", "last_name": "...", "photo": ..., "strength": ..., "dex": ..., "intelligence": ..., "health": ..., "hitPoints": ..., "will": ..., "perception": ..., "basicSpeed": ..., "basicMove": ..., "hidden": ..., "user": { "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... } } </pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, negalima prieiga prie duomenų. 422 – netinkamai suformuluota užklausa.

20 lentelė. Character traits delete (DELETE) specifikacija

API metodas	Character traits delete (DELETE)
Paskirtis	Pašalinti veikėjo įgūdį.
Kelias iki metodo (angl. route)	/api/Characters/{characterId}/traits/{traitId}
Užklaustos struktūra	-
Užklaustos „Header“ dalis	Authorization: Bearer {token}

Atsakymo struktūra	-
Atsakymo kodas	204 (No Content)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas, jei negalima prieiga prie turinio.

3.5. Naudotojų API metodai

21 lentelė. Users index (GET) specifikacija

API metodas	Users index (GET)
Paskirtis	Gauti naudotojų sąrašą.
Kelias iki metodo (angl. route)	/api/users
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>[{ "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... }, ...]</pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – jei siunčiamas autorizacijos JWT žetonas yra neteisingas.

22 lentelė. Users update (PATCH) specifikacija

API metodas	Users update (PATCH)
Paskirtis	Redaguoti naudotojo rolę.
Kelias iki metodo (angl. route)	/api/users/{userId}
Užklausos struktūra	<pre>{ "role": "GAMEMASTER" }</pre>

	}
Užklauso „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<pre>{ "id": ..., "username": "...", "email": "...", "role": "...", "enabled": ..., "accountNonLocked": ..., "authorities": [{ "authority": "..." }], "credentialsNonExpired": ..., "accountNonExpired": ... }</pre>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – neteisingas JWT žetonas.

23 lentelė. Users delete (DELETE) specifikacija

API metodas	Users delete (DELETE)
Paskirtis	Ištrinti naudotoją.
Kelias iki metodo (angl. route)	/api/users/{userId}
Užklauso struktūra	-
Užklauso „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	204 (No Content)
Galimi klaidų kodai	403 – neteisingas JWT žetonas.

3.6. Autentifikacijos API metodai

24 lentelė Naujo naudotojo registracija

API metodas	Users create (POST)
Paskirtis	Naujo naudotojo registracija
Kelias iki metodo (angl. route)	/api/v1/auth/register
Užklauso struktūra	<pre>{ "username": "...", "email": "...", "password": "..." }</pre>

Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<code>{"token": "...", "refreshToken": "..."} </code>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	422 – naudotojas jau egzistuoja

25 lentelė. Users authenticate (POST) specifikacija

API metodas	Users create (POST)
Paskirtis	Naudotojo prisijungimas.
Kelias iki metodo (angl. route)	/api/v1/auth/authenticate
Užklausos struktūra	<code>{ "username": "...", "password": "..." }</code>
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	<code>{"token": "...", "refreshToken": "..."} </code>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	401 – neteisingi prisijungimo duomenys.

26 lentelė. Users logout (GET) specifikacija

API metodas	Logout (GET)
Paskirtis	Atsijungimas
Kelias iki metodo (angl. route)	/api/v1/auth/logout
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}
Atsakymo struktūra	-
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – neteisingas arba nėra JWT žetono.

27 lentelė. Refresh (GET) specifikacija

API metodas	Refresh (GET)
Paskirtis	JWT žetono atnaujinimas
Kelias iki metodo (angl. route)	/api/v1/auth/refresh
Užklausos struktūra	-
Užklausos „Header“ dalis	Authorization: Bearer {token}

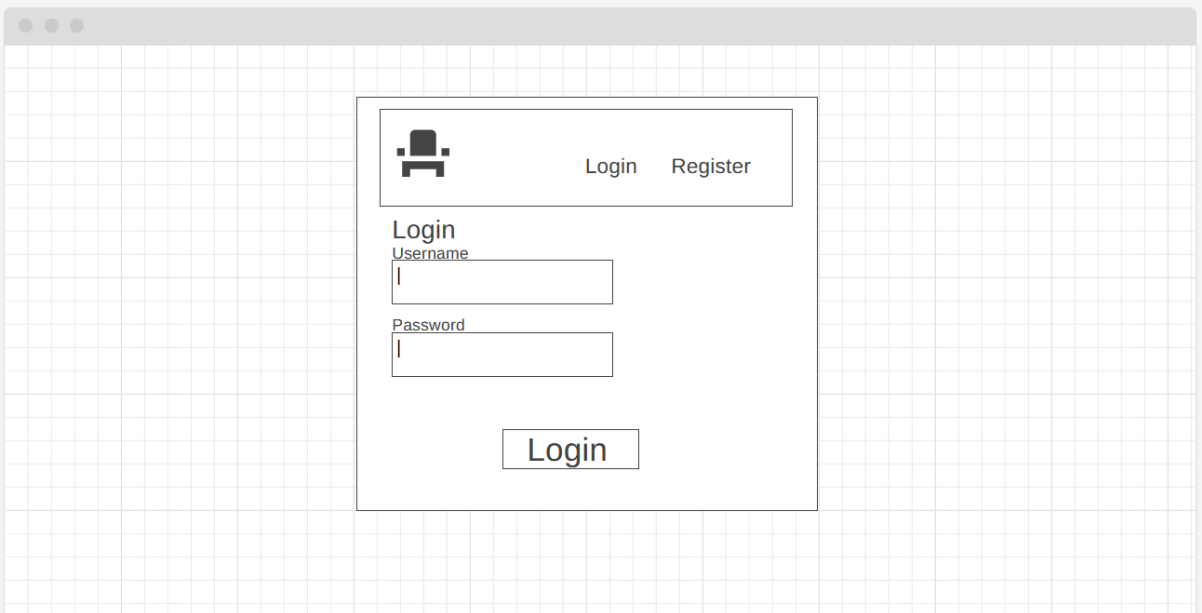
Atsakymo struktūra	<code>{"token":"...", "refreshToken":null}</code>
Atsakymo kodas	200 (OK)
Galimi klaidų kodai	403 – neteisingas JWT žetonas.


4. Sistemos naudotojo sąsaja

Priėjimas prie sistemos yra galimas tik prisiregistravus ir prisijungus.

4.1. Sistemos internetinė aplikacija

Kol kas sistema yra pasiekiamą tik privačiame tinkle. Registracijai prie sistemos reikalingas slap vardis, elektroninio pašto adresas ir slaptažodis. Serveriui užsikrovus yra matomas prisijungimo langas. Iš jo galima pereiti į registracijos langą.



 [Login](#) [Register](#)

Login

Username:

Password:

Login

Do not contact us

© 2023 My Company. All rights reserved.

Užsiregistravus ir prisijungus pereinama į /campaigns, kur surašyti visi žaidimai.



Campaigns Logout

Campaign List

New
campaign

Campaign name

Edit

Delete



Campaigns Logout

Campaign List

New Campaign

[I cry - 100](#)

Edit

Delete

[I cry hard - 100](#)

Edit

Delete

Do not contact us

© 2023 My Company. All rights reserved.

Norint sukurti naują žaidimą, ar redaguoti jau esamą yra paspaudžiamas atitinkamas mygtukas ir iššoka modalinis langas.

Create Campaign

Campaign name

Points Limit

Create

Close



Campaigns Logout

Create Campaign

Campaign Name:

Points Limit:



Create Campaign

Close

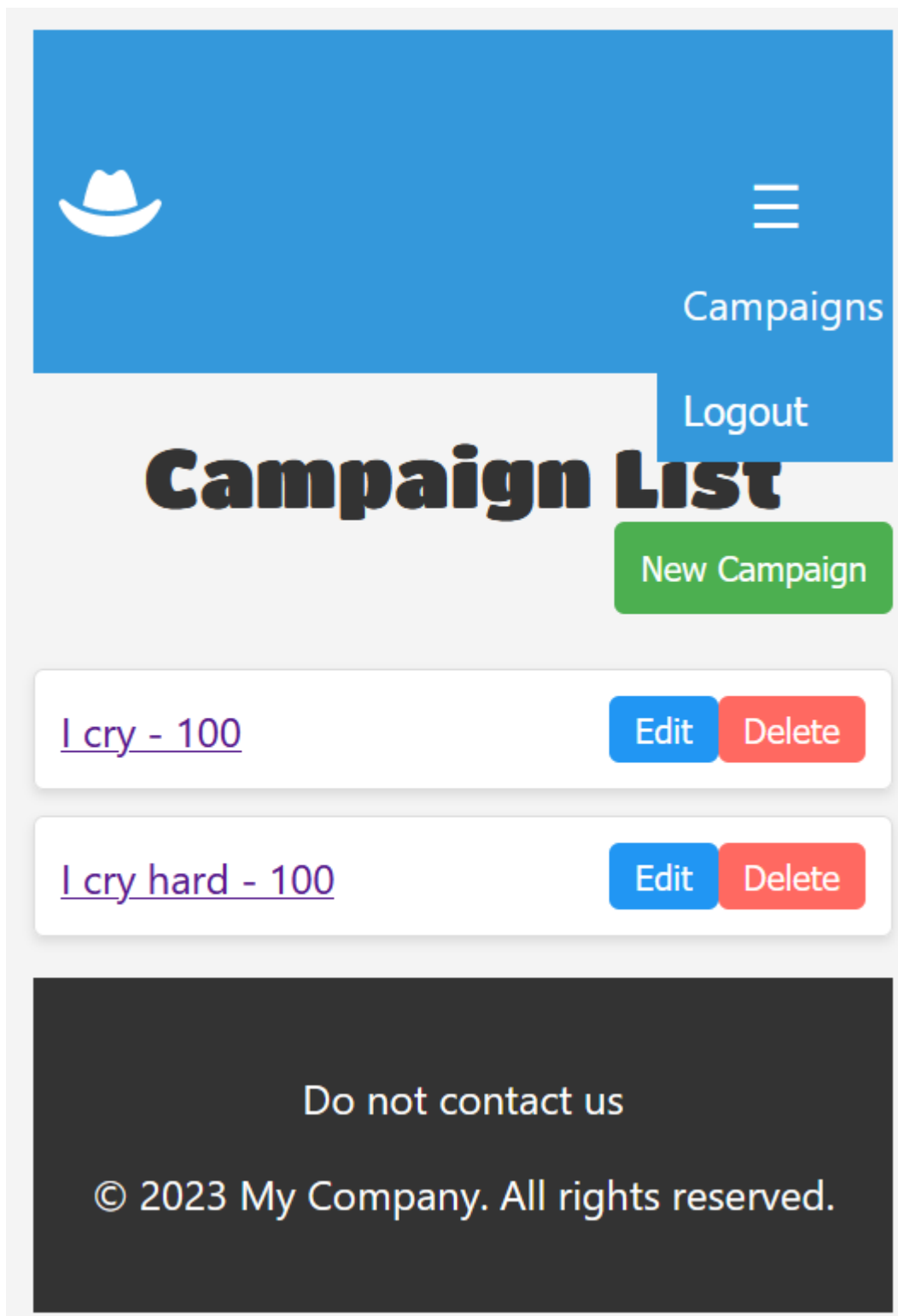
View Campaign

Delete

Delete

[I cry - 100](#)

[I cry hard](#)



Susiaurinus ekrana, atsiranda Hamburger menu.



Campaigns Logout

Character List

Back

Character

Skills

Traits


Delete

Add character

Name

|

Add

CampaignsLogout

Characters

[Back to Campaigns](#)

Kestis1 Kestuliukas

[Skills](#)[Traits](#)[Delete Character](#)

Add Character

Photo:

First Name:

Last Name:

Strength:

Dexterity:

Intelligence:

Health:

Hit Points:

Will:

Perception:

Basic Speed:

Basic Move:

Hidden:

☐

[Add Character](#)

Do not contact us

© 2023 My Company. All rights reserved.



Campaigns Logout

Skills

Back

Skill


Delete

Add skill

Name

|

Add

CampaignsLogout

Skills

Back to Character List

skill - Level 1Delete

Add Skill

Name:

Description:

Level:

1

Difficulty:

Easy

Add Skill

Do not contact us

© 2023 My Company. All rights reserved.

Toliau galima valdyti veikėjo įgūdžius ir bruožus.



Campaigns Logout

Traits

Back

Trait

Delete

Add trait

Name

|

Add



Campaigns Logout

Traits

[Back to Character List](#)

trait

Delete

Add Trait

Name:

Description:

Price:

0



Add Trait

Do not contact us

© 2023 My Company. All rights reserved.

Administratorius gali prieiti specialų langą, kur būtų valdomi visi naudotojai.



Campaigns Logout

Users

User

Delete



User List

why - ADMIN

Delete user

Kestis - GAMEMASTER

Delete user

kestis1 - PLAYER

Delete user

KestisBenamis - PLAYER

Delete user

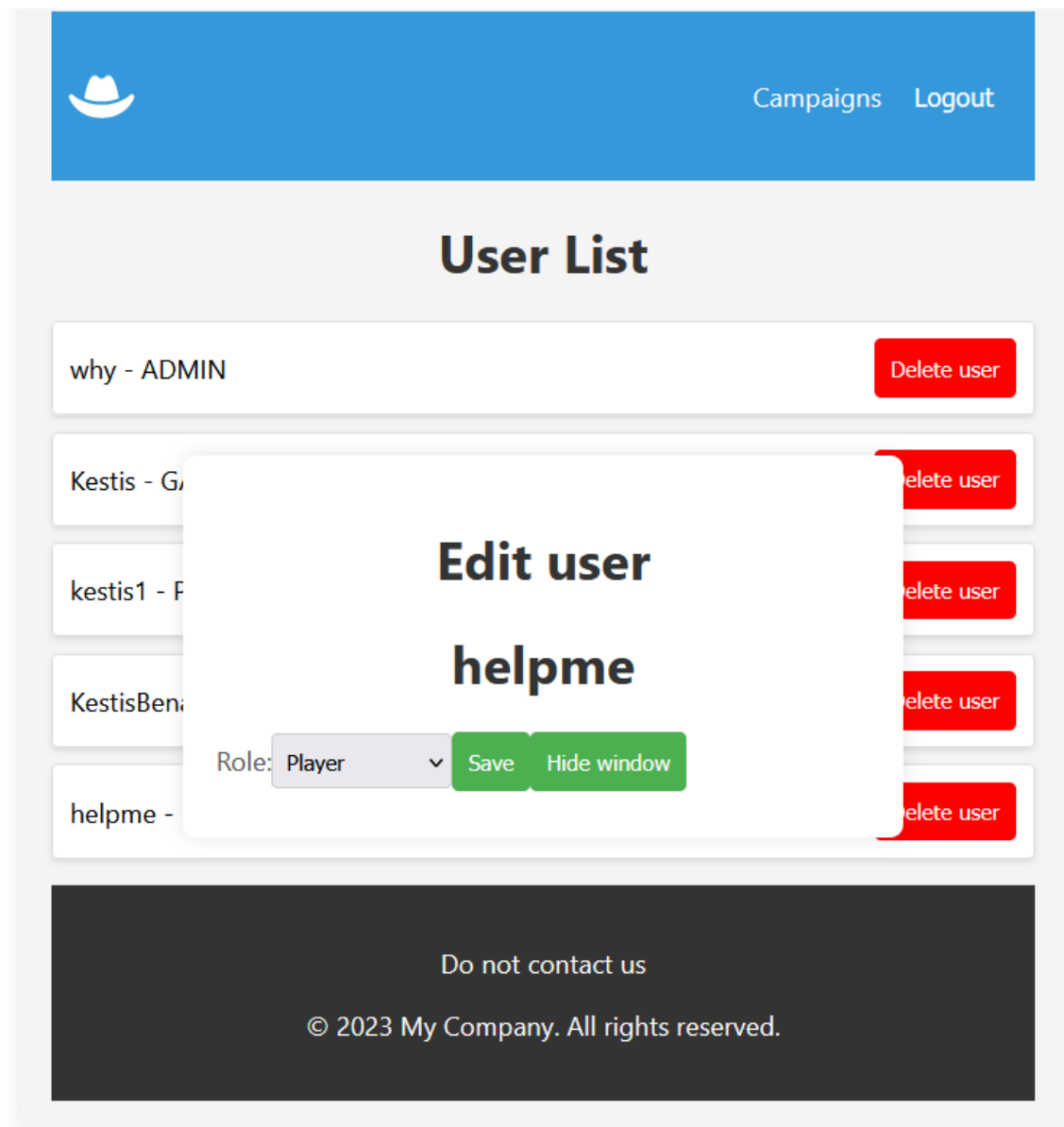
helpme - PLAYER

Delete user

Do not contact us

© 2023 My Company. All rights reserved.

Paspaudus ant naudotojo atsidaro modalinis langas, kuriame galima pakeisti naudotojo rolę.



5. Išvados

Projektas buvo įgyvendintas sėkmingai. Realizuotos trys rolės. Buvo įgyventi visi funkciniai reikalavimai. Back-end buvo relizuotas su Spring Boot Java, o Front-End su React.js, panaudota MySQL duombazė. Su Spring Boot sukurtas API realizuoja CRUD funkcijas žaidimams (angl. campaign), veikėjamas (angl. characters) žaidimuose, igūdžiams (angl. skills) ir bruožams (angl. traits). Administratorius dar turi galimybę manipuluoti naudotojais. API turi klaidą, jog kai prie turinio negalima prieiti dėl pasibaigusio JWT žetono vis tiek yra grąžinamas 403 klaidos pranešimas. Front-end dalis yra pritaikyta ir prie mobilių prietaisų ekranų.