

# **VIDZEMES TEHNOLOĢIJU UN DIZAINA TEHNIKUMS**

Profesionālās vidējās izglītības programma – Programmēšana

Kvalifikācija – Programmēšanas tehnikas

Izglītības ieguves forma – Klātie

Kvalifikācijas eksāmena praktiskās daļas dokumentācija

## **“PET FOOD” TĪMEKĻA VIETNE MĀJDZĪVNIEKU INFORMĀCIJAS UZMEKLĒŠANAI**

Autors

IP20 grupas audzēknis  
**Kārlis Ivars Braķis**

Cēsīs, 2024

# ANOTĀCIJA

**Tīmekļa vietne “Pet Food”.** Braķis, Kārlis Ivars. Kvalifikācijas darbs, 71 lappuse, 69 attēli, 6 tabulas, 17 literatūras avoti, 4 pielikumi.

Šī projekta mērķis ir izveidot lietotājam draudzīgu vietni, kurā mājdzīvnieku īpašnieki var viegli atrast saviem mājdzīvniekiem piemērotu barību. Mērķis ir nodrošināt visaptverošu resursu, kas vienkāršo mājdzīvnieku barības izvēles procesu un nodrošina mājdzīvnieku labklājību.

Šajā darbā es aprakstu vietnes izstrādes soļus un apspriežu izmantoto metodiku. Tas ietver līdzīgu vietņu izpēti, mājdzīvnieku īpašnieku vajadzību noteikšanu un tīmekļa vietnes izveidi, kas efektīvi atbilst šīm vajadzībām.

Tiek sniegts ieskats izstrādes procesā un apspriežu visas problēmas, ar kurām saskāros ceļā. Neraugoties uz dažiem šķēršļiem, projekts tika veiksmīgi pabeigts, un to var potenciāli uzlabot.

Dokumentā ir iekļauts lietotāja ceļvedis jeb rokasgrāmata, kas palīdzēs apmeklētājiem orientēties vietnē un maksimāli izmantot tās funkcijas.

Kopumā mājaslapas projekta "Mājdzīvnieku barība" mērķis ir sniegt mājdzīvnieku īpašniekiem vērtīgu informāciju, lai viņi varētu pieņemt pārdomātus lēmumus par savu mājdzīvnieku uzturu un labklājību.

**Saite uz koda glabātuvi:** <https://github.com/karluga/petfood>

**Vietnes saite:** <https://petfood.brakis.id.lv>

## ANNOTATION

**Website “Pet Food”.** Braķis, Kārlis Ivars. Qualification paper, 71 pages, 69 images, 6 tables, 17 literature sources, 4 appendices.

This project aims to create a user-friendly website where pet owners can easily find suitable food for their pets. We aim to provide a comprehensive resource that simplifies the process of selecting pet food and ensures the well-being of pets.

In this paper, I outline the steps taken to develop the website and discuss the methodology employed. This includes researching similar websites, identifying the needs of pet owners, and designing a website that meets those needs effectively.

Additionally, I provide insights into the development process and discuss any challenges faced along the way. Despite encountering some obstacles, the project was completed with room for further enhancement.

Furthermore, the paper includes a user guide to assist visitors in navigating the website and making the most of its features.

Overall, the "Pet Food" website project aims to empower pet owners with valuable information to make informed decisions about their pets' nutrition and well-being.

**Link to code repository:** <https://github.com/karluga/petfood>

**Link to access the website:** <https://petfood.brakis.id.lv>

# SATURA RĀDĪTĀJS

Anotācija.....	2
Annotation .....	3
Satura rādītājs .....	4
Jēdzieni un to skaidrojumi.....	5
Ievads.....	8
1. Literatūras apskats un variantu salīdzinājums.....	9
1.1 Līdzīgas programmas vai pakalpojumi .....	9
1.2 API izpēte .....	12
2. Prasību specifikācija.....	14
2.1 Funkcionālās prasības.....	15
2.2 Nefunkcionālās prasības.....	17
3. Mērķa sasniegšanas līdzekļu izvēle un pamatojums .....	19
4. Sistēmas un datu struktūras modelis .....	21
4.1 Lietošanas gadījumu diagramma.....	21
4.2 Klienta – servera modelis .....	22
4.3 Priekšgalsistēma .....	24
4.4 Aizmugursistēmas ER diagramma .....	27
5. Funkcionālais un dinamiskais sistēmas modelis .....	28
6. Reaģējoša tīmekļa vietne.....	30
7. Testēšana.....	33
7.1 Testēšanas nosacījumi .....	33
8. Lietotāja ceļvedis.....	52
8.1 Galvenā funkcionalitāte.....	52
8.2 Lietotāja skati .....	56
8.3 Administratora skati .....	59
Secinājumi un priekšlikumi .....	61
Izmantotā literatūra.....	63
Pielikumi.....	65
1. <i>pielikums</i> . Meklēšanas rezultātu aizmugursistēmas API metodes loģika (PHP) .....	66
2. <i>pielikums</i> . Meklēšanas rezultātu priekšgalsistēmas loģika (JavaScript) .....	67
3. <i>pielikums</i> . MVC programmēšanas pieejas piemērs .....	69
4. <i>pielikums</i> . Logotips .....	71

## JĒDZIENI UN TO SKAIDROJUMI

Atslēgas vārds	Skaidrojums
Kods	Kods ir jebkurā programmēšanas valodā rakstīts teksts, ko saprot dators.
Priekšgalsistēma (angļu v. <i>front-end</i> )	Priekšgalsistēma atbild par visu, kas ir redzams lietotāja pusē. Priekšgalsistēmas 3 galvenās tehnoloģijas jeb pamati ir HTML, JavaScript un CSS.
Aizmugursistēma (angļu v. <i>back-end</i> )	Aizmugursistēma atbild par darbībām, kuras veic lietotājs servera pusē. Tās galvenais uzdevums ir apstrādāt un saglabāt datus, pārvaldīt lietotāja pieprasījumus, un nodrošināt saskarni starp datu bāzi un klienta ierīci, lai sniegtu nepieciešamās informācijas atbildes.
Serveris	Serveris ir dators vai programmatūra, kas sniedz pakalpojumus vai atbild uz pieprasījumiem no citiem datoriem vai programmatūras klientiem.
Lietojumprogrammas saskarne jeb API (angļu v. <i>application programming interface</i> )	Tas ir veids, kādā divas programmas var sazināties, neietekmējot vienu otru. Šajā gadījumā Priekšgalsistēma sūta pieprasījumu uz aizmugursistēmu, kur tas tiek apstrādāts un atsūta atbildi uz priekšgalsistēmu masīva formātā. Šajā darbā izmantoju JSON formātu.
API galapunkts (angļu v. <i>endpoint</i> )	Līdzīgi kā direktorijas ceļš failu sistēmā, arī galapunkts norāda uz to, ko mēs vēlamies iegūt vai darīt. Šis saite piemēram ir API saite, un tajā ir redzams galapunkts "species/102084770", kas norāda uz konkrētu resursu vai darbību attiecīgajā API. Šeit "species" varētu būt kā resursa nosaukums, un "102084770" varētu būt identifikators konkrētajai sugai vai objektam. Katrs galapunkts piedāvā konkrētu informāciju vai darbību, kas pieejama attiecīgajā resursā vai sistēmā.
Pieprasījums jeb vaicājums (angļu v. <i>request</i> )	Tiešā nozīmē tas ir process, kurā tiek pieprasīti, izveidoti, izmainīti vai izdzēsti dati. Tas var būt starp lietotāja pusi un serveri, datu bāzi vai API.
Validācija	Validācija ir process, kurā pārbauda datu atbilstību noteiktām prasībām vai standartiem.
Saskarne jeb UI (angļu v. <i>user interface</i> )	Lietotājam draudzīgs veids, kā cilvēks mijiedarbotos ar datorprogrammu vai ierīci. Tas ietver visu, ko lietotājs redz un sajūt, tai skaitā ekrānu, ikonas, pogas un citus vizuālus elementus, kas atvieglo lietotāja mijiedarbību ar programmām.
Statisks (angļu v. <i>static</i> )	Tīmekļa vietne, kuras saturs nemainās.
Dinamisks (angļu v. <i>dynamic</i> )	Tīmekļa vietne, kuras saturs nav statisks jeb ir atkarīgs no lietotāja darbībām.
Reaģējošs (angļu v. <i>responsive</i> )	Tāda tīmekļa vietne, kura pielāgojas un izmaina satura izkārtojumu atkarībā no ekrāna lieluma.
Modālais logs (angļu v. <i>pop-up window</i> )	Neliels logs, kas satur kādu informāciju.

Ievadforma (angļu v. <i>input form</i> )	Ekrānā attēlojamo datu forma, kas atvieglo datu ievadīšanu un rediģēšanu.
Sesija	Sesija ir veids, kā saglabāt un uzturēt informāciju par lietotāju starp vairākiem pieprasījumiem vietnē.
Verifikācija jeb autorizācija	Šajā kontekstā verifikācija nozīmē lietotāja autorizēšanos sistēmā, kā arī API apstiprināšanas darbības.
Klienta jeb lietotāja puse (angļu v. <i>client-side</i> ) un servera puse (angļu v. <i>server-side</i> )	Šajā kontekstā vārdu savienojumi “klienta puse” un “servera puse” tiek lietoti lai raksturotu konkrētas darbības kurā konkrētā puse ir iesaistīta.
OAuth	OAuth ir standarts, kas ļauj lietotājiem piekļūt resursiem, neizpaužot savus autorizācijas datus, izmantojot trešās puses identifikācijas nodrošinātāju. Tas bieži tiek izmantots, lai autorizētu lietotājus piekļūt trešās puses lietotnēm vai pakalpojumiem, izmantojot citu pakalpojumu sniedzēja identifikācijas informāciju.
Lomas (angļu v. <i>roles</i> )	Lomas ir vienkāršs veids kā pārvaldīt sistēmas atļaujas, izmantojot lietotāja lomu sadalījumu.
HTTP / HTTPS (angļu v. HyperText Transfer Protocol)	Lietojumslāņa protokols, kas paredzēts datu apmaiņai starp tīmekļa serveriem un pārlūkprogrammām. HTTPS ir HTTP protokols ar papildus drošības slāni (S - secure).
Žetons (angļu v. <i>token</i> )	Žetonu izmanto, lai identificētu lietotāju uz konkrētu laika periodu un lietotājs var veikt atsevišķas noteiktas darbības, kad viņam ir piešķirts šis žetons. Tas tiek izveidots vienu reizi, izmantojot pierakstīšanās datus.
Noklusējums (angļu v. <i>default</i> )	Noklusējums attiecas uz kādu iestatījuma vērtību, kuru lietotājs var mainīt, bet ja to nemaina, tad tā paliek kā kādas iepriekš noteikta vērtība.
Pārvirzīšana (angļu v. <i>redirect</i> )	Satura vai lapas nomaiņa.
Vietrādis jeb URL	Tīmekļa adrese, kuru atverot nonāksiet konkrētā tīmekļa vietnē.
Ģenerēt (angļu v. <i>generate</i> )	Automātiski izveidot saturu.
Atvērtā koda programmatūra (angļu v. <i>open-source</i> )	Atvērtais kods ir tāds, kas ir brīvi pieejams aplūkošanai ikvienam un to var mainīt atbilstoši savām vajadzībām.
Vienas lapas aplikācija (single-page application)	Vienas lapas aplikācija ir tīmekļa lietojumprogramma, kas darbojas uz vienas HTML lapas, kura tiek dinamiski atjaunināta un mainās, neielādējot jaunas lapas no servera. Tā vietā, izmantojot JavaScript un citas modernās tīmekļa tehnoloģijas, ielādē sākotnējo lapu un pēc tam datu apmaiņas procesā dinamiski ielādē saturu un veic izmaiņas vietnē, reaģējot uz lietotāja darbībām.
Ietvars (angļu v. <i>framework</i> )	Programmēšanas kontekstā, ietvars ir konkrēts veids kādā tiek izveidota failu struktūra un pielietotas atkārtoti lietojamas komponentes.

Bibliotēka (angļu v. <i>library</i> )	Savukārt “bibliotēka” programmēšanas kontekstā ir koda bloks, kas satur atkārtoti lietojamās funkcijas, procedūras vai klases. Bibliotēkas var izmantot, lai paātrinātu programmatūras izstrādes procesu, jo tās nodrošina atkārtoti lietojamās funkcijas, kas var tikt izmantotas vairākos projektos.
---------------------------------------	--

# IEVADS

Mājdzīvnieku aprūpe ir ļoti svarīga un būtiska problēma mūsdienu sabiedrībā. Tā vienmēr būs aktuāla kamēr vien cilvēkiem būs mājdzīvnieki.

Lai gan ir daudz informācijas par šo tēmu internetā, ir grūti atrast konkrētu informāciju, nerakstot garāku aprakstu, ko tieši tu vēlies sameklēt. Šobrīd, ierakstot *Google* meklētājā konkrēti ko vēlies barot, tas apkopos informāciju no dažādiem avotiem.

Cita nozīmīga problēma ir būt taupīgam. Piemēram, ja tev paliek pāri kāds ēdiens, tā vietā lai to izmestu ārā, labāk ir pārbaudīt vai šo ēdienu var barot savam sunim. It īpaši, ja neesi pārliecināts vai droši drīkst barot, to var ātri pārbaudīt šajā tīmekļa vietnē.

## Mērķis

Kvalifikācijas darba mērķis ir izstrādāt intuitīvu, ērti un viegli saprotamu tīmekļa vietni, kur jebkurš var ievadīt sava mājdzīvnieka sugu un uzmeklēt informāciju par pārtikas produktu drošu patēriņu attiecīgās dzīvnieka sugas pārstāvim.

## Uzdevumi

1. Izveidot tīmekļa vietnes dizainu.
2. Izstrādāt priekšgalsistēmu (angļu v. *front-end*), kurā lietotāji varēs uzmeklēt dzīvnieku sugas un pārtikas produktus.
3. Izveidot datu bāzes arhitektūru ER diagrammas veidā un izstrādāt datu bāzi.
4. Izveidot aiz mugursistēmu (angļu v. *back-end*), kurā tiks saglabāti visu produktu dati par drošu izmantošanu, lietotāju profilu dati un dzīvnieku sugu dati.
5. Veikt tīmekļa vietnes testēšanu.
6. Novērst radušās problēmas un veikt labojumus.

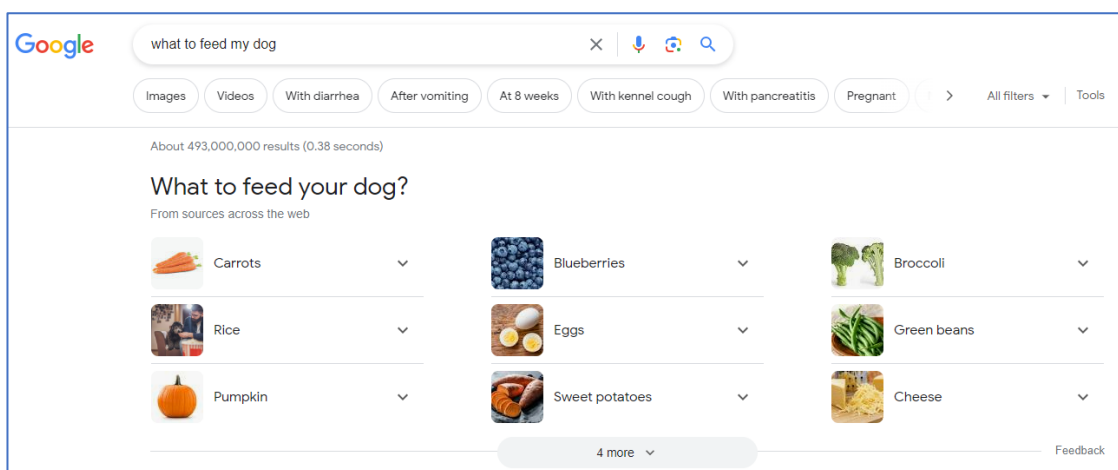


# 1. LITERATŪRAS APSKATS UN VARIANTU SALĪDZINĀJUMS

Šajā nodaļā tiks apskatīti citi, jau pastāvoši varianti apspriestās problēmas atrisināšanai. Ir svarīgi meklēt līdzīgus risinājumus, kuri jau ir izveidoti, lai izdomātu kādas ir atšķirības starp tavu risinājumu un citiem. Varbūt nemaz nebiji iedomājies, ka šāda ideja jau ir izdomāta un realizēta. Ja ideja ir oriģināla vai esi pamanījis nepilnības jau pastāvošajos risinājumos, vai lietas kas ir labāk izveidotas, tad realizēt pašam savu ideju būs daudz vieglāk un nozīmīgāk.

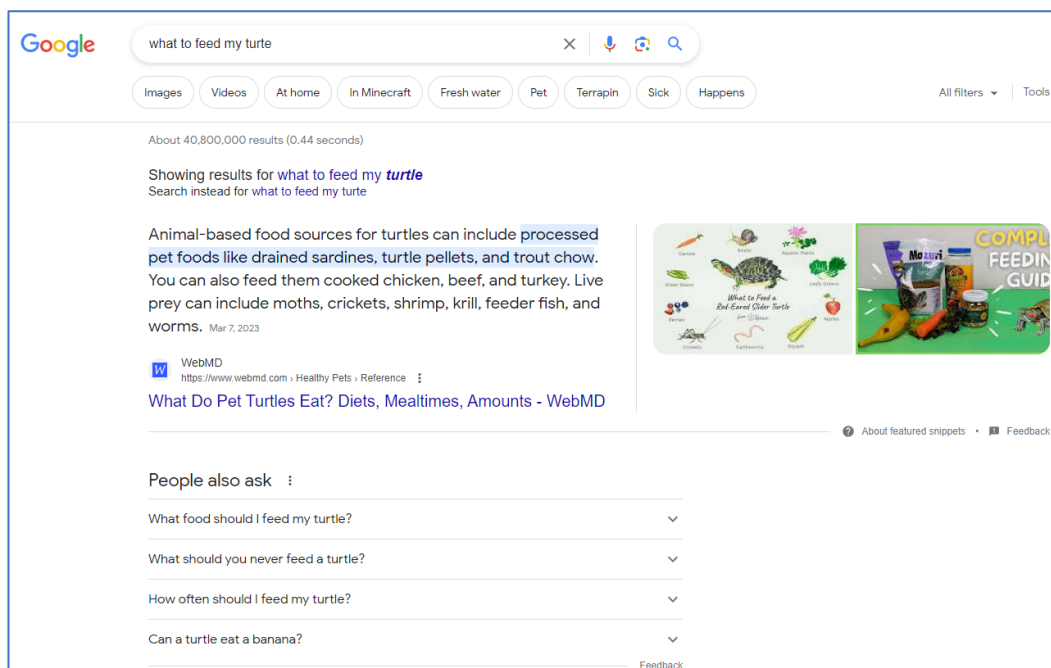
## 1.1 LĪDZĪGAS PROGRAMMAS VAI PAKALPOJUMI

Sobrīd *Google* meklētājā var atrast apkopotu informāciju, uzreiz ierakstot, ko meklēt, piemēram, “Ko barot savam sunim?” (skatīt 1. attēlu). Tomēr šī informācija tiek apkopota no dažādiem avotiem un nav pieejama vienuviet. Manas idejas mērķis ir informāciju apkopot vienā uzticamā resursā.



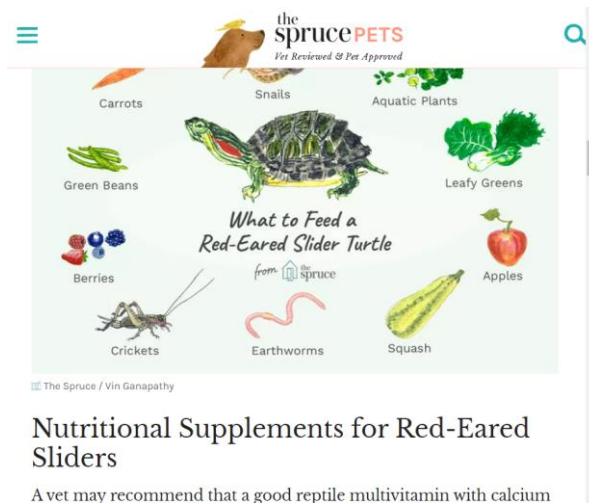
1. attēls. *Google* meklētājs

Kad vēlas sameklēt jebkādu citu konkrētāku informāciju, piemēram par bruņurupučiem, tad ir redzams, ka svarīgākā informācija tiek iezīmēta, bet pati informācija ir no atsevišķas tīmekļa vietnes.



2. attēls. Google meklētājs

Meklējot tālāk informāciju par bruņurupučiem, atradu vienu tīmekļa vietni (skatīt 2. pielikumu), kurā ir plašs informācijas klāsts par daudzām dzīvnieku klasēm (skatīt 4. attēlu). Konkrēti atradu pārskatāmu ilustrāciju par pārtikas produktiem, kurus ēd Sarkano ausu bruņurupuči (skatīt 3. attēlu).



3. attēls. The Spruce Pets raksts



4. attēls. *The Spruce Pets* sākuma lapa

*The Spruce Pets* tīmekļa vietnē ir specializēta un apkopota informācija no dažādiem speciālistiem, kur informācija arī tiek rūpīgi pārbaudīta. Kā arī ir iespēja apskatīt dažādus mājdzīvnieku saistītus mājsaimniecības produktus.

Viena negatīvā lieta ir tāda, ka cilvēki nevar uzrakstīt komentārus zem satura, ko viņi par to domā un kāda ir bijusi viņu pieredze, tomēr šajā mājaslapā ir daudz satura, kas varētu būt grūti pārskatāms, ja pievienotu komentēšanas iespēju. Vēl ko es pamanīju ir tas, ka nav iespējas nomainīt valodu, kas ir svarīga lapas sastāvdaļa, tomēr tas nozīmētu lielu darbu un laika ieguldījumu.

Vēl es ieraudzīju, ka nav iespējas izveidot “grāmatzīmes”, atzīmējot informāciju lasīšanai vēlāk. Kaut gan vienmēr ir iespēja izveidot grāmatzīmi savā interneta pārlūkprogrammā, ir daudz ērtāk, ka tām ir izveidota atsevišķa sadaļa iekš tīmekļa vietnes.

Salīdzinot ar manu taisīto tīmekļa vietni, *The Spruce Pets* ir pieejama pārbaudīta informācija no pieredzējušiem profesionāļiem šajā jomā. Manā lietotnē būs gan apstiprināta, gan neapstiprināta informācija no dažādiem cilvēkiem. Manā lietotnē būs pieejama bloga sadaļa, kur cilvēki varēs publicēt fotogrāfijas ar savu mājdzīvnieku. Gan kā pierakstīties lietotājs, gan anonīms. Pēdējā nozīmīgā atšķirība ir lokalizācija. Manā lietotnē būs pieejams saturs ne tikai vienā valodā. Ja saturs nav vēl iztulkots, tad tiks rādītas citas valodu iespējas un angļu valoda būs noklusējumā.

Secinu, ka salīdzinātā lietotne ir vairāk oficiāls informācijas avots, kur uzsvars ir uz dzīvnieku veselību un labklājību – medicīnas viedoklī, bet man būs kā neoficiāls sociālais medijs.

## 1.2 API IZPĒTE

Lai varētu ērtāk pārvaldīt sugu datus, tika meklēts veids kā iegūt patiesu dzīvnieku sugu informāciju. Tas tika panākts izmantojot GBIF API (Global Biodiversity Information Facility, 2024), kas atbilda šiem kritērijiem: pieejams par brīvu, detalizēta informācija par taksonomijas pakāpēm un ir vairāki API galapunkti (angļu v. *endpoint*), lai iegūtu konkrētākus datus.

Sakopju vienas sugas pārstāvja informāciju tabulā, lai izprastu cik daudz grupas ir dzīvnieku taksonomijā (skatīt 1. tabulu). Ievietoju arī atsauci uz izvēlētās sugas aprakstu (Wikipedia, 2024).

GBIF API lietojumprogrammas saskarnes pieprasījuma un atbildes piemērs:

HTTP pieprasījuma saite: <https://api.gbif.org/v1/species/102084770>

Atslēgas parametrs: 102084770

Daļa no API atbildes (angļu v. *response*):

```
"kingdomKey": 101683523,  
"phylumKey": 102017110,  
"classKey": 102074563,  
"orderKey": 102084090,  
"familyKey": 102084282,  
"genusKey": 102084770,  
"canonicalName": "Eurycea",  
"vernacularName": "Brook Salamanders",  
"rank": "GENUS", // SUGA
```

## GBIF lietojumprogrammas saskarnes sugu hierarhija

<b>Grupas (angļu v. rank) nosaukums angliski</b>	<b>Grupas nosaukums latviski</b>	<b>Vernakulārais jeb tautā lietotais nosaukums (angļu v. vernacular name)</b>	<b>Kanoniskais nosaukums (angļu v. canonical name), bieži sakrīt ar zinātnisko nosaukumu</b>	<b>GBIF datu bāzes sugu identifikācijas numurs</b>
SUBSPECIES	APAKŠSUGA	-	-	-
SPECIES	SUGA	-	-	-
GENUS	ĢINTS	Brook Salamanders	Eurycea	102084770
FAMILY	DZIMTA	Lungless Salamanders	Plethodontidae	102084282
ORDER	KĀRTA	Salamanders	Caudata	102084090
CLASS	KLASE	Amphibians	Amphibia	102074563
SUPERCLASS	VIRSKLASE	-	Tetrapoda	134090735
INFRAPHYLUM	INFRATIPS	-	Gnathostomata	134090735
SUBPHYLUM	APAKŠTIPS	vertebrates	Vertebrata	102018738
PHYLUM	TIPS	chordates	Chordata	102017110
INFRAKINGDOM	INFRAVALSTS	-	Deuterostomia	134090703
SUBKINGDOM	AKAKŠVALSTS	-	Bilateria	134090702
KINGDOM	VALSTS	animals	Animalia	101683523

## 2. PRASĪBU SPECIFIKĀCIJA

Prasību specifikācija ir nepieciešama, lai nodrošinātu skaidru un sīkstu izpratni par tam, kādas funkcionalitātes un īpašības paredzēts attīstīt konkrētajā programmā vai sistēmā. Turpmāk minēti ir daži iemesli, kāpēc prasību specifikācija ir svarīga.

1. **Skaidra saprašana:** Prasību specifikācija palīdz visiem projektā iesaistītajiem saprast, kādas mērķus un funkcionalitātes vajadzētu sasniegt. Tas nodrošina vienotu redzējumu un novērš neskaidrības attīstības procesā.
2. **Projekta plānošana:** Prasību specifikācija ir svarīgs instruments projekta plānošanai. Tā dod izstrādātājiem iespēju novērtēt nepieciešamo laiku un resursu daudzumu, lai izpildītu visus noteiktos uzdevumus.
3. **Kvalitātes nodrošināšana:** Prasību specifikācija apraksta kvalitātes standartus un prasības, kas jāievēro izstrādes procesā. Tas ļauj nodrošināt, ka gala rezultāts atbilst klienta vai lietotāja vajadzībām un standartiem.
4. **Komunikācija:** Tas kalpo kā komunikācijas instruments starp dažādām projektā iesaistītajām pusēm, piemēram, izstrādātājiem, projektu vadītājiem un klientiem. Tas palīdz izvairīties no neskaidrībām un neprecizitātēm.
5. **Izmaiņu pārvaldība:** Prasību specifikācija var būt pamats izmaiņu pārvaldībai. Ja rodas nepieciešamība mainīt vai papildināt funkcionalitāti, specifikācija nodrošina struktūru, lai veiktu šīs izmaiņas, nemainot kopējo projektu nevajadzīgā veidā.
6. **Klienta apstiprinājums:** Prasību specifikācija ir svarīgs dokuments klientam vai pasūtītājam, lai viņi varētu izteikt atzinumu par plānoto darbu un sniegt atpakaļ saistošu atbildi par to, vai visi viņu prasītie nosacījumi ir ietverti.

Kopumā, prasību specifikācija ir būtisks instruments, lai nodrošinātu veiksmīgu un efektīvu programmatūras izstrādi, vienlaikus samazinot riskus un neskaidrības projekta gaitā.

## 2.1 FUNKCIONĀLĀS PRASĪBAS

Pirms darba sākšanas ir nepieciešams noskaidrot kādas funkcionālās prasības būs nepieciešamas, kādu funkcionalitāti pildīs programma un kāds labums no tā būs. Funkcionālās prasības ir nepieciešamas, lai aprakstītu, kādus konkrētus darbības un funkcionalitātes aspektus izstrādājamajā sistēmā vai lietotnē vajadzētu iekļaut. Tās raksturo ar aprakstu un nosacījumiem (skatīt 2. tabulu).

2. tabula

Funkcionālās prasības

Nr.	Nosaukums	Apraksts
1.	Domēna vārda iegūšana	Viens no svarīgākajiem soļiem ir iegūt domēna vārdu, zem kura tavs pakalpojums būs pieejams tiešsaistē.
2.	Tīmekļa vietņu mitināšanas servisa izvēle	Lai varētu publicēt savu programmu, ir nepieciešams kāds serveris uz kura var uzturēt savu pakalpojumu un tas būtu pieejams publiski.
3.	Navigācija jeb galvene tīmekļa vietnes augšpusē	Navigācija ir svarīgākais vietnes elements, kurā var redzēt svarīgākās darbības kādas var veikt vai skatus, kurus var apmeklēt.
4.	Logotips	Vienkāršam un atpazīstamam logotipam ir izšķiroša nozīme labiem panākumiem. Tā tiks novietota lapas galvenē, kreisajā pusē. Uzspiežot uz tās, lietotājs tiks pārvirzīts uz sākuma lapu.
5.	Konta izveide jeb reģistrācija	Reģistrētiem lietotājiem ir iespēja uzmeklēt produktu informāciju daudz ērtāk, kā arī var piedalīties vietnes satura uzlabošanā un dalīties ar savu mājdzīvnieka turēšanas pieredzi.
6.	Iespēja reģistrēties ar populāriem sociālajiem tīkliem kā <i>Google</i> un <i>Facebook</i>	Modernākajās tīmekļa vietnēs var daudz ērtāk reģistrēties izmantojot kādu populāru sociālo tīklu. Šis tiks realizēts izmantojot standarta autorizācijas protokolu <i>OAuth</i> (Google Developers, 2023)
7.	Paroles maiņa	Jābūt iespējai nomainīt paroli ar e-pasta palīdzību, ja gadījumā nepieciešams atkopt savu kontu vai vēlās savādāku paroli.
8.	E-pasta verifikācija	E-pasta verifikācija ir nepieciešama, lai varētu nomainīt paroli vai arī saziņai par “lomus” iegūšanu.
9.	Pieslēgšanās sistēmā - “Ienākt” poga	Šī poga atklāj pieslēgšanās ievadformu, kur lietotājs var pieslēgties sistēmā, kā arī piereģistrēties.

10.	Izrakstīšanās - “Iziet” poga	Atļauj izrakstīties no sistēmas.
11.	Profilu izdzēšana	Iespēja izdzēst profilu ar noteiktu laika periodu, kurā lietotājs var pārdomāt savu izvēli.
12.	Sugu meklēšana (ievadforma)	Šeit lietotājs var meklēt dzīvnieku sugas, rakstot arī saņemot visatbilstošākos ieteikumus, balstoties uz ierakstiem datu bāzē un API.
13.	Ēdiena jeb pārtikas produktu meklēšana (ievadforma)	Līdzīgi kā sugu meklēšanā, rakstot ēdienus, tiks parādīti ieteikumi ātrākai meklēšanai.
14.	Kategoriju izvēle	Var izvēlēties lielākās, populārākās dzīvnieku kategorijas un meklēt informāciju balstoties uz tām.
15.	Mājdzīvnieku pievienošana profilam (ievadforma)	Var pievienot vienu vai vairākas savu mājdzīvnieku sugas. Var pievienot arī mājdzīvnieka vārdu, lai meklējot ātrāk atpazītu.
16.	Piekļuves līmeņu sistēma izmantojot lietotāju lomas	Ierobežo piekļuvi dažādām sistēmas darbībām, piemēram, satura izstrādi un rediģēšanu.



## 2.2 NEFUNKCIONĀLĀS PRASĪBAS

Nefunkcionālās prasības ir svarīgas, lai definētu kvalitātes standartus un ierobežojumus, piemēram, dizaina aspektus, veiktspēju, drošību un citus faktorus, kas neietilpst tieši funkcijas aprakstā (skatīt 3. tabulu).

3. tabula

Nefunkcionālās prasības

Nr.	Nosaukums	Apraksts
1.	Tīmekļa vietnes stils	Elementu izvietojumam jābūt skaidram, saprotamam un salasāmam. Vietnes stils ir veidots pēc brūna toņa krāsu paletes.
2.	Informācija un saturs	Saturam jābūt atbilstošam un informatīvam.
3.	Lokalizācija	Lokalizācija ir nepieciešama, lai lietotāji no dažādām pasaules valstīm var izmantot vietni un saprast to.
4.	Šifrēšana	Paroles un cita sensitīva informācija ir jāšifrē pirms tā tiek pieglabāta datubāzē, lai to nevar vienkārši nolasīt trešās puses personas.
5.	Aizsardzība pret servera pārslogošanu	Jānovērš iespēja atļaut vienam lietotājam vai jebkādam ļaunprogrammatūrai sūtīt lielu daudzumu pieprasījumu uz serveri.
6.	Lietotāja datu aizsardzība	Ir dažādi uzbrukumu veidi ar kuriem var manipulēt ar lietotāja datiem, piemēram, CSRF (Cross-Site Request Forgery), ka formas iesūtīšana var notikt no citas tīmekļa vietnes un ne no lietotāja datora. Cits piemērs ir XSS (Cross-Site Scripting), kurā uzbrucējs var izmantot tīmekļa vietnes nepilnības, lai izpildītu savu koda fragmentu, potenciāli pakļaujot datus neautorizētai piekļuvei.
7.	CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart)	Pirms ievadformas nosūtīšanas jāpārliedz vai sūtītājs ir īsts cilvēks.
8.	Burģera izvēlne (angļu v. <i>hamburger menu</i> )	Mazākiem ekrāna izmēriem navigācija tiks pārvērsta par ikonu ar trīs svītrām, ko sauc par burgeri. Uzspiežot uz tā var atvērt un savērst izvēlni, lai tā neaizņem daudz vietas uz ekrāna.
9.	Reaģējoša tīmekļa vietne (angļu v. <i>responsivity</i> )	Saturam jāmaina izvietojums atkarībā no ekrāna lieluma, vieglākai lasāmībai.
10.	Pieejamība (angļu v. <i>accessibility</i> )	Saistās ar SEO (Search Engine Optimization), kas ir nepieciešams lai tava tīmekļa vietne būtu viegli atrodama meklētājprogrammās.

11.	Ātrdarbība (angļu v. <i>performance</i> )	Ātrdarbība ir svarīga, jo tā nosaka to, cik laba pieredze būs lietotājiem, izmantojot lietotni. Lietotnes darbība nedrīkst atvairīt esošus lietotājus.
12.	Veiktspēja un mērogojamība ( <i>efficiency &amp; scalability</i> )	Tīmekļa vietnei jābūt efektīvai, lai tā būtu spējīga apstrādāt pieaugošu datu apjomu un lietotāju skaitu.
13.	Uzturamība (angļu v. <i>maintainability</i> )	Uzturamība attiecas uz to, cik viegli un efektīvi var uzturēt un atjaunināt tīmekļa vietni. Laba uzturamība ietver skaidru un saprotamu kodu, regulārus atjauninājumus, kā arī labu dokumentāciju, lai citas personas varētu turpināt darbu pie projekta.
14.	Atbilstība (angļu v. <i>compliance</i> )	Tīmekļa vietnei jāatbilst noteikumiem, standartiem un likumdošanai, piemēram, datu aizsardzības noteikumiem vai pieejamības standartiem. Tas nodrošina, ka vietne ir juridiski un ētiski atbilstoša.
15.	Atkārtota lietojamība (angļu v. <i>reusability</i> )	Atkārtota lietojamība attiecas uz kodu un komponentēm, kuras jebkurš, ne tikai tu pats vari izmantot citās vietnēs vai projektos.

### 3. MĒRĶA SASNIEGŠANAS LĪDZEKĻU IZVĒLE UN PAMATOJUMS

#### **Izstrādei un produktivitātei:**

*Visual Studio Code* (VS Code) – koda redaktors

*PhpMyAdmin* (PMA) – datu bāzes pārvaldības sistēma

*Figma* – dizaina izstrādes rīks tiešsaistē

*GitHub* – tiešsaistes platforma un pakalpojums, kas nodrošina versiju kontroles un sadarbības iespējas programmatūras izstrādei, izmantojot Git versiju kontroles sistēmu. *GitHub* piedāvā arī funkcionalitāti izsekošanai, problēmu ziņošanai, sadarbībai ar citiem izstrādātājiem un daudz citu rīku, kas veicina efektīvu programmatūras izstrādi un sadarbību.

#### **Programmā izmantotās tehnoloģijas:**

HTML (HyperText Markup Language) – tīmekļa lapu struktūras valoda, kas nosaka dokumenta loģisko uzbūvi, izmantojot marķēšanas elementus un atribūtus. Tas ir galvenais instruments informācijas attēlošanai tīmeklī, veidojot virknes elementu, kas ietver tekstu, attēlus, saites un citus resursus.

CSS (Cascading Style Sheets) – stila lapas valoda, kas paredzēta HTML dokumentiem, lai definētu to izskatu un izvietojumu. Tas nodrošina iespēju veidot vizuālu stilu, piemēram, krāsu shēmas, fontus, izkārtojumu un animācijas, atdalot izskatu no saturu un ļaujot veidot vienotu un viegli uzturamu dizainu.

*JavaScript* – tīmekļa lapu programmēšanas valoda, ko izmanto, lai piešķirtu dinamisku un interaktīvu uzvedību tīmekļa lapām. Tas palīdz veikt darbības ar aizmugursistēmu bez lapas pārlādēšanas.

PHP (Hypertext Preprocessor) – servera puses programmēšanas valoda, ko izmanto, lai izstrādātu dinamiskas tīmekļa lapas un lietojumprogrammas. Tas ļauj veikt datu apstrādi, veikt datu bāzu mijiedarbību un ģenerēt dinamisku saturu, padarot tīmekļa lietotnes elastīgas un funkciju bagātas.

*MySQL* – atvērta koda relāciju datu bāzu pārvaldības sistēma, ko izmanto, lai saglabātu un pārvaldītu strukturētus datus. Tas piedāvā efektīvu veidu, kā organizēt un atkopt informāciju datu bāzēs, nodrošinot uzticamu datu glabāšanu un piekļuvi tīmekļa lietotnēm.

*jQuery* – starpplatformu JavaScript bibliotēka, kas vienkāršo tīmekļa lapu izstrādi un manipulāciju ar *DOM* (Dokumenta Objekta Modelis). Tas piedāvā dažādas funkcijas, piemēram, animācijas, notikumu apstrādi un *AJAX* pieprasījumu izpildi, ļaujot izstrādāt efektīvu un intuitīvu saskarni (OpenJS Foundation, 2024).

AJAX (Asynchronous JavaScript and XML) – tehnoloģijas komplekts, kas ļauj veikt asinhronas datu apmaiņas operācijas ar serveri, neuztraucoties par tīmekļa lapas pilnīgu pārlādi. Tas ļauj dinamiski atjaunināt tīmekļa lapu, pārsūtīt un saņemt datus bez nepieciešamības pilnībā pārlādēt lapu. Tas bieži tiek izmantots, lai uzlabotu lietotāja pieredzi un optimizētu datu pārsūtīšanu starp klienta pusi un serveri.

*Laravel* – augsta līmeņa tīmekļa ietvars, kas izveidots, izmantojot PHP programmēšanas valodu. Tas nodrošina iespējami vienkāršu un efektīvu veidu, kā attīstīt tīmekļa lietojumprogrammas, izmantojot modernas programmatūras izstrādes principus un piedāvājot plašu funkciju klāstu, kas atvieglo sarežģītu uzdevumu risināšanu, piemēram, datu bāzu mijiedarbību, maršrutēšanu un lietotāja autentifikāciju (T. Otwell, 2021).

*Laravel UI Auth* iesācēja pakete (angļu v. *scaffolding*) – šis bija ātrākais veids kā uztaisīt funkcionālu reģistrācijas un pieslēgšanās sistēmu, toties bija nepieciešams to savienot ar tulkojumiem un pārrakstīt esošo loģiku, kā arī pielāgot priekšgalskatu, jo vēlējos lai reģistrācija būtu pieejama visās vietnes sadaļās.

*Composer* – Nepieciešama programmatūra pilnvērtīgai *Laravel* funkciju izmantošanai, izmantojot komandrindu (angļu v. *terminal*), tādām kā – migrāciju veidošana un palaišana, modeļu un kontrolieru automātiska ģenerēšana, kešatmiņas (angļu v. *cache*) iztīrīšanai piemēram, maršrutiem, skatiem, uzstādījumiem un atsevišķi saglabātiem datiem (N. Adermann, J. Boggiano, 2024).

*Font Awesome* – ikonu un simbolu bibliotēka, kas nodrošina plašu klāstu noformējuma elementu un ikonu, kuras var ievietot tīmekļa lapās. Tā piedāvā lietotājiem iespēju viegli pievienot stilizētus simbolus, uzlabojot tīmekļa lietotņu izskatu un piedāvājot estētisku pievienoto vērtību (Fonticons, Inc., 2024).

*Bootstrap* – tīmekļa izstrādes ietvars, kas piedāvā komplektu priekšdefinētu stila klases, JavaScript komponentes un citus resursus, lai paātrinātu un vienkāršotu responsīvo tīmekļa lapu izveidi. Tas ir izstrādāts, lai palīdzētu veidot modernas, mobilajām ierīcēm draudzīgas lapas, izmantojot HTML, CSS un JavaScript. Bootstrap ietver gan izskata, gan interakcijas komponentes, tāpēc jums nav nepieciešams no jauna izstrādāt visu no nulles (The Bootstrap team, 2024).

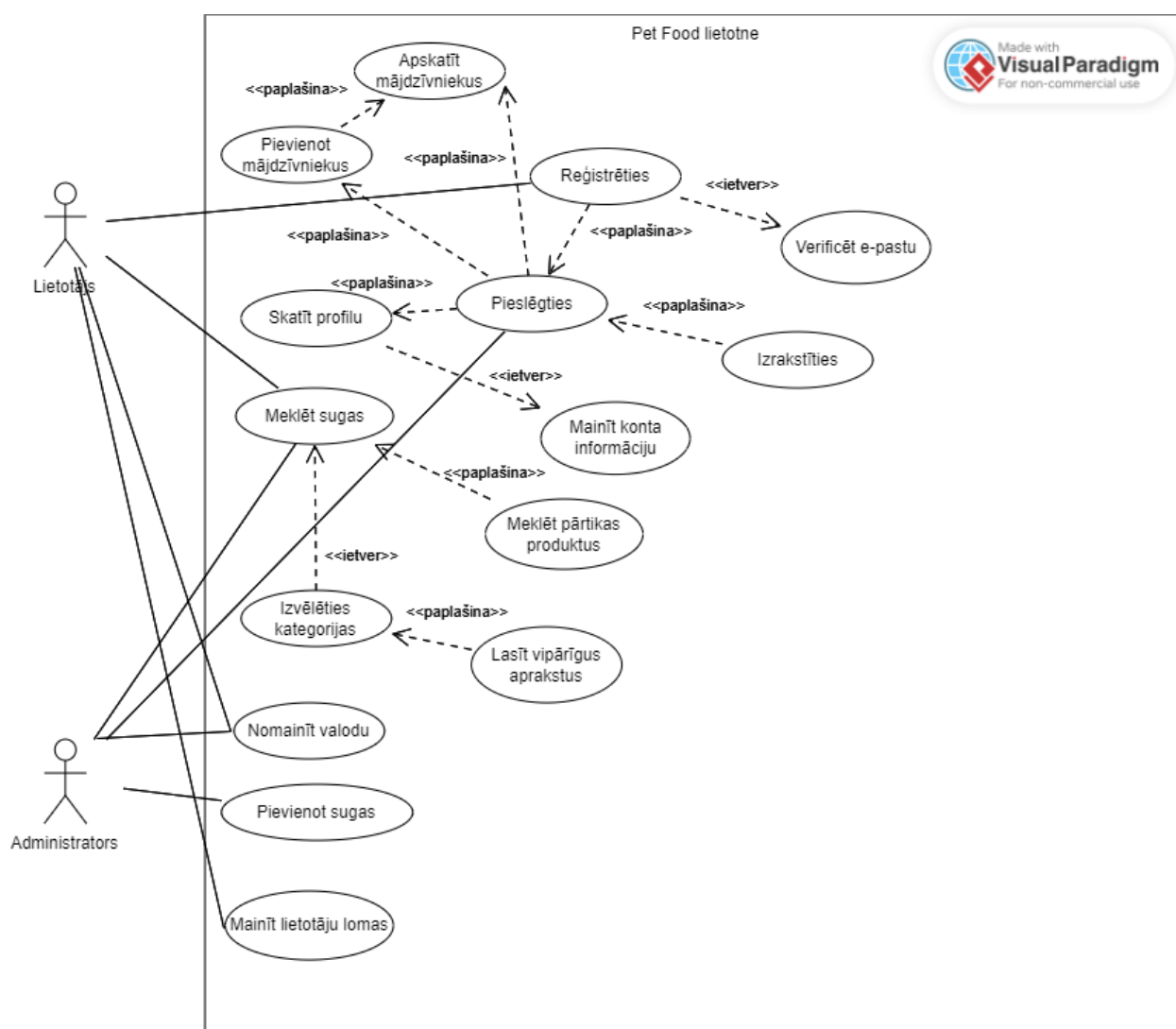
*DataTables* – jQuery paplašinājums, kas ļauj veikt tabulu datu parādīšanu un apstrādi interaktīvā veidā. Tas ļauj pārvērst parastās HTML tabulas dinamiskos, interaktīvos elementus ar meklēšanas, kārtotāšanas un lappušu numerācijas (angļu v. *pagination*) funkcijām. Šeit svarīgākais pielietojums būs tabulas savēršana mazākiem ekrāna izmēriem (SpryMedia Ltd., 2023).

## 4. SISTĒMAS UN DATU STRUKTŪRAS MODELIS

Kad sistēmas funkcionālās un nefunkcionālās prasības ir definētas, pirms pašas tīmekļa vietnes izstrādes, nepieciešams izveidot tās struktūras modeli jeb arhitektūru. Šajā nodaļā tiek prezentēta tīmekļa vietnes *Pet Food* lietošanas gadījumu diagramma, klienta – servera modelis, izmantojot REST API, priekšgalsistēmas komponentu modelis un aizmugursistēmas ER diagramma.

### 4.1 LIETOŠANAS GADĪJUMU DIAGRAMMA

Lietošanas gadījumu diagramma sniedz pārskatu par lietotnes funkcionalitāti un norāda nepieciešamās darbības, lai sasniegtu citas funkcijas. Šī diagramma izceļ lietotāja iespējas un precīzi norāda soļus, kas jāveic, lai efektīvi pārvietotos starp dažādām lietotnes darbībām. Diagrammu skatīt attēlā zemāk.

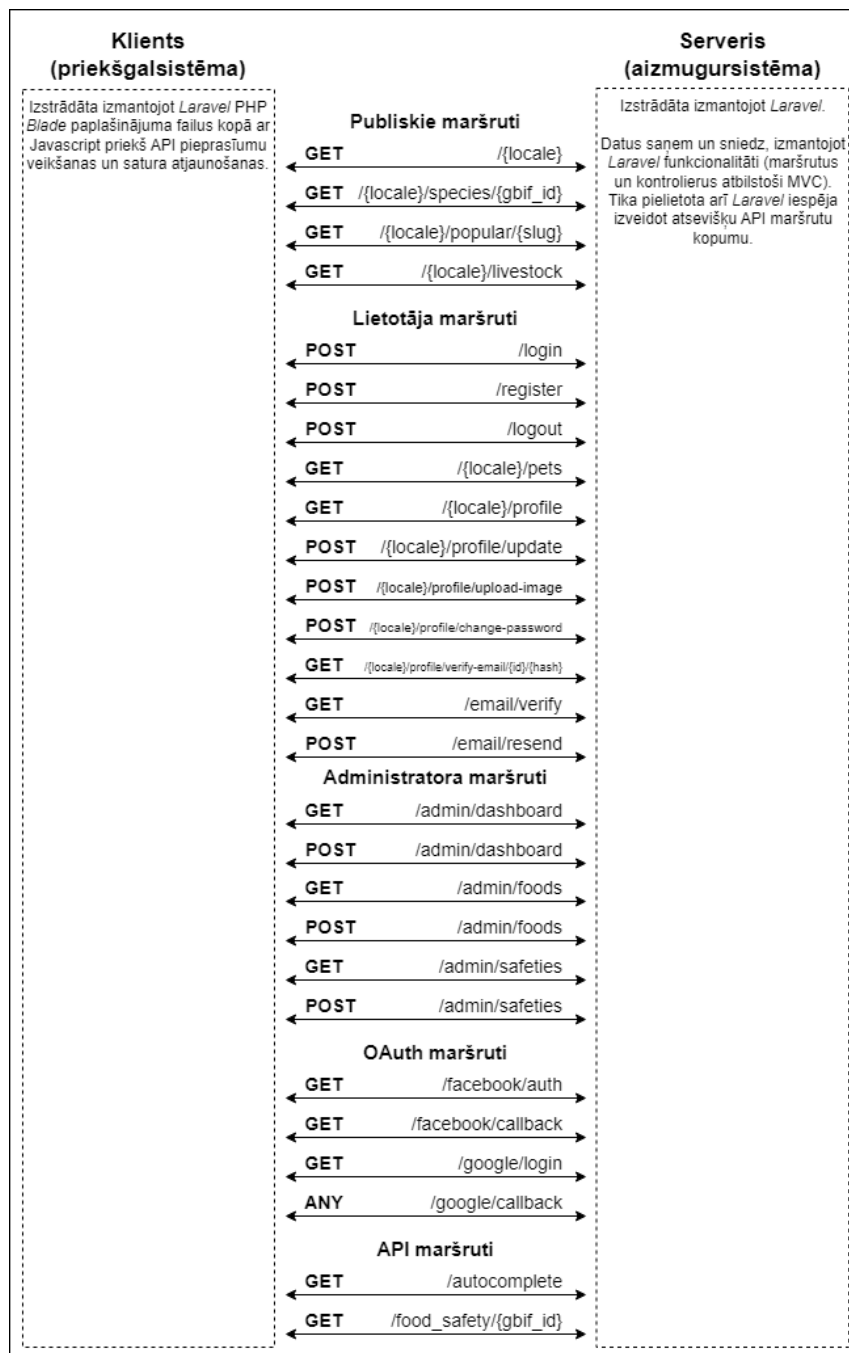


5. attēls. Lietošanas gadījumu diagramma

## 4.2 KLIENTA – SERVERA MODELIS

Tīmekļa vietne *Pet Food* balstās uz klienta – servera modeli, kurā komunikācija tiek nodrošināta, izmantojot HTTP protokolu. Turklāt priekšgalsistēma un aizmugursistēma ir savstarpēji atdalītas, izmantojot REST API, lai nodrošinātu vieglāku izstrādi un uzturamību.

REST API pieeja iekļauj sevī maršrutu izveidi, kuros priekšgalsistēma un aizmugursistēma savstarpēji apmainās ar datiem. *Laravel* arī ir pieejama noderīga komanda, ar kuras palīdzību var apskatīt visus vietnes atbalstītos maršrutus – “php artisan route:list”. Maršruti apkopoti un klienta – servera modelis attēlots zemāk.



6. attēls. Klienta – servera modelis

Kā redzams 6. attēlā, visvairāk tika izmantotas HTTP GET un POST metodes. Metode “ANY” nav standarta metode, bet gan veids kā *Laravel* var atļaut jebkādu HTTP pieprasījumu. Dinamiski aizvietojamie maršruta parametri ir pierakstīti ar viettura sintaksi, piemēram, “{locale}” nozīmē valodas 2 ciparu ISO standarta kodu (Wikipedia, 2024). Tas var būt arī savādāks citiem reģioniem, piemēram “en-gb” (Great Britain) vai “en-us” (United States). Visas HTTP metodes un to funkcijas redzamas tabulā zemāk.

4. tabula

HTTP metodes	
HTTP metode	Apraksts
GET	Bieži izmanto, lai iegūtu tīmekļa lapas, datus vai resursus.
POST	Izmanto, lai izveidotu jaunus resursus. Parasti to izmanto, iesniedzot veidlapu serverī. Piemēram šajā gadījumā tiek ievietoti dati datubāzē.
PUT	Parasti tiek izmantots, lai atjaunotu esošos resursus. Tomēr HTML veidlapās parasti izmanto POST un iekļauj paslēptu ievades lauku ar nosaukumu “_method” un vērtību “PUT”, lai imitētu PUT pieprasījumu.
PATCH	Līdzīgi kā PUT, to izmanto, lai atjaunotu esošos resursus. Tomēr to parasti izmanto daļējiem atjauninājumiem, kad tiek rediģētas tikai dažas vērtības.
DELETE	Izmanto, lai dzēstu resursus.
OPTIONS	Šī metode tiek izmantota, lai aprakstītu komunikācijas iespējas mērķa resursam. To var izmantot, lai noskaidrotu URL atbalstītās metodes vai servera iespējas.
HEAD	Šī metode ir līdzīga GET, bet tā neatgriež ziņojuma ķermeni atbildē. To bieži izmanto, lai iegūtu meta-informāciju (informācija, kas ir domāta kā priekšstats, daloties ar saiti) par resursu, nevis pārsūtot visus datus par vietni.

### 4.3 PRIEKŠGALSISTĒMA

Priekšgalsistēma tika izstrādāta izmantojot PHP paplašinājuma *Blade* sintakses failus (skatīt 5. tabulu). Vienam no tiem devu nosaukumu “app.blade.php”, kurā ievietoju visu struktūru priekš kopīgo skatu informācijas. Tajā ietilpst navigācija un autorizācija.

Stils tika rakstīts vienā CSS failā, izņemot stilus priekš meklēšanas ievades. Tas nav ļoti pārskatāmi, toties bija jāiekļauj tikai divi faili iekš galvenā skata, kā arī tas uzlabo ielādes ātrumu. Ir ieteicama SCSS failu struktūra, kurus var sakopot vienā CSS failā (S. Sarker, 2022).

Javascript tika pielietots, lai sugu un ēdienu meklētājā rādītos atbilstošākie rezultāti pēc datiem un lietotājs varētu uzreiz uzspiest uz konkrētās sugas un atrast ēdienus. Failu nosaukumi – “autocomplete.js” un “load\_foods.js”.

5. tabula

Priekšgalsistēmas skatu lomas

Mapes struktūra	Skata nosaukums	Loma
layouts\	app.blade.php	Atsevišķs skats, kas atrodas savā mapē, kurš iekļauj sevī citus skatus.
-	welcome.blade.php	Sākuma skats (angļu v. <i>landing page</i> ).
-	animals.blade.php	Skats, kurā var redzēt populārākos dzīvniekus vai to kategorijas. Tas iekļauj gan “popular”, gan “livestock” skatus.
-	popular.blade.php	Paplašina “animals” skatu.
-	livestock.blade.php	Paplašina “animals” skatu.
-	home.blade.php	Skats, uz kuru lietotājs tiek pārvirzīts, kad lietotājs autentificējas. Šeit var redzēt savu mājdzīvnieku sugas, ja tie ir pievienoti iepriekš.
-	profile.blade.php	Lietotāja skats, kurā var rediģēt savu personīgo informāciju
-	species.blade.php	Publiskais skats, kurā var meklēt konkrētas sugas ēdienus.
vendor\notifications\	email.blade.php	UI Auth iesācēja pakā pārrakstītais (angļu v. <i>override</i> ) fails, kurā es



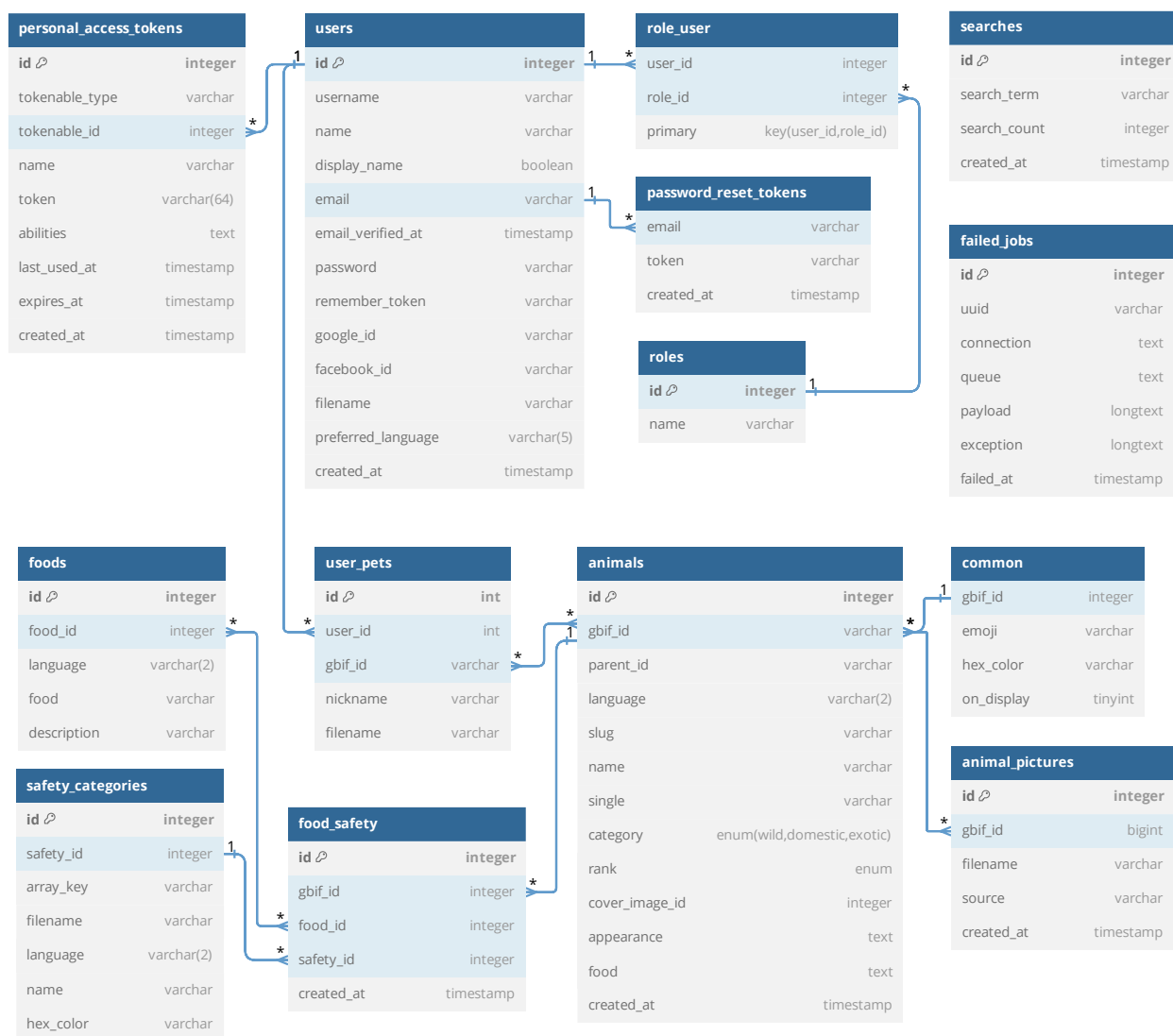
		varēju personalizēt izsūtītā e-pasta saturu un izskatu (piemēram, sūtot paroles atjaunošanas vai e-pasta verifikācijas e-pastus).
errors\	404.blade.php	<i>Laravel</i> piedāvā iespēju personalizēt kļūdu skatus, ja ievieto failu tieši šajā manierē.
admin\layouts\	dashboard.blade.php	Atsevišķs skats, kas atrodas savā mapē, kurš iekļauj sevī citus skatus.
admin\	animals.blade.php	Administrators skats, kurā var pievienot dzīvnieku sugas, iekļaujot to tulkojumus citās valodās.
admin\	foods.blade.php	Administrators skats, kurā var pievienot ēdiena nosaukumus, iekļaujot to tulkojumus citās valodās.
admin\	safeties.blade.php	Administrators skats, kurā var sasaistīt dzīvnieka sugu ar ēdiena nosaukumu un pievienot tā drošības kategoriju.
resources\views\auth\passwords\	confirm.blade.php	Netiek primāri izmantots, bet var divkārtoti pārbaudīt lietotāja identitāti, prasot vēlreiz ievadīt paroli veicot kādu darbību.
resources\views\auth\passwords\	email.blade.php	Pieteikšanās formā, kura ir pieejama visās vietās iekš vietnes, ja lietotājs nav autentificējies, tur rādās saite uz šo skatu (“Aizmirsi savu paroli?”), lai var atjaunot paroli.
resources\views\auth\passwords\	reset.blade.php	Pēc paroles atjaunošanas e-pasta saņemšanas, lietotājs tiek pie ievadformas, kur var nomainīt paroli vienas stundas laikā.
resources\views\auth\	login.blade.php	Netiek primāri izmantots, bet ir pieejams.
resources\views\auth\	register.blade.php	Netiek primāri izmantots, bet ir pieejams.

resources\views\auth\	verify.blade.php	Skats, kuram var piekļūt apskatot savu profilu, ja vēlas verificēt e-pastu.
-----------------------	------------------	---

## 4.4 AIZMUGURSISTĒMAS ER DIAGRAMMA

Aizmugursistēma tika veidota, izmantojot programmēšanas valodas PHP aizmugursistēmu ietvaru Laravel. Tas balstās uz pieeju Modelis – skats – kontrolieris un tajā tiek izmantota objektrelāciju kartēšana. Tā rezultātā tieša mijiedarbība ar datu bāzi nav nepieciešama, bet datu struktūra tiek definēta ar Laravel rīku Eloquent, kam ir savi datu tipi.

Diagrammas izveidošanai izmantoju <https://dbdiagram.io/> saitei specifisko DBML sintaksi. Izveidotā ER diagramma ir redzama attēlā.



7. attēls. ER diagramma

Ikkatram lietotājam var būt neviens, viens vai vairāki mājdzīvnieki vai lietotāja lomas.

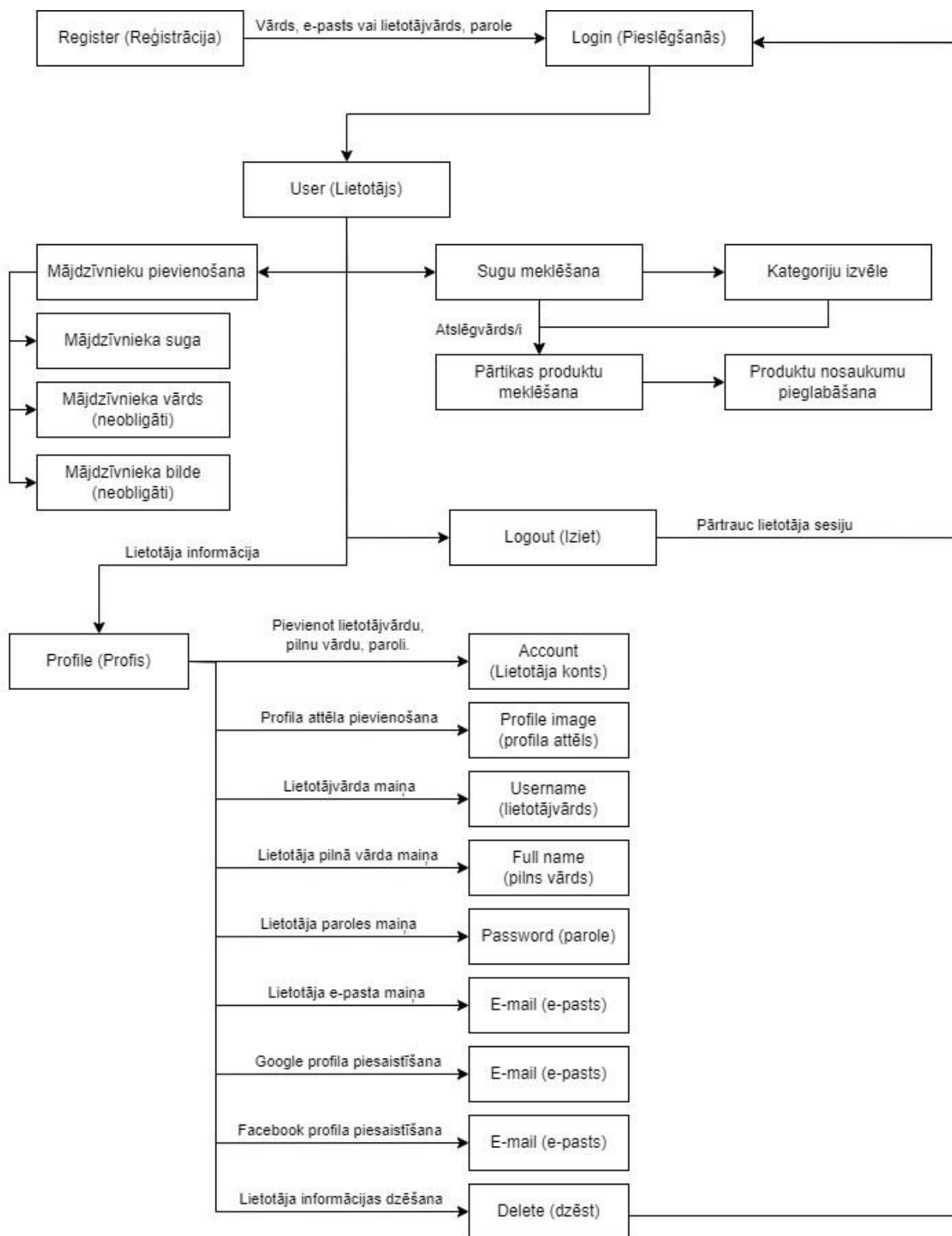
Ikkatrai dzīvnieka sugai var būt neviena, viena vai vairāk bilžu un ēdiena nosaukumu saistījumu.

## 5. FUNKCIONĀLAIS UN DINAMISKAIS SISTĒMAS MODELIS

Datu plūsmas shēmā pirmā lietotāja darbība ir reģistrācija, kur ir jāievada nepieciešamie dati, lai izveidotu lietotāju – pilns vārds, e-pasts, lietotājvārds un parole. Ir iespējams arī reģistrēties izmantojot kādu no populārākajām sociālajiem tīkliem – *Google* un *Facebook*. Kad lietotājs ir veiksmīgi reģistrējies, viņu pārvirza uz sākuma lapu. Nākamajā reizē, atverot lietotni, varēs izmantot savu izveidoto profilu.

Kad lietotājs ir pieslēdzies, tam ir pieeja vienai jaunai funkcijai, kuru nevar veikt bez profila izveidošanas – mājdzīvnieku pievienošana. Ja lietotājam ir administratora, loma tad viņš var pievienot dzīvnieku sugas, to aprakstus, ēdienus un sugām pielikt ēdienu drošības kategorijas. Galvenā funkcionalitāte, kas ir pieejama visiem lietotnes apmeklētājiem, ir dzīvnieku sugu un to ēdienu meklēšana.

Lietotāja profila sekcijā ir iespējams nomainīt dažādu informāciju par sevi – vārdu, lietotājvārdu, paroli un e-pastu, kā arī pievienot sevi aprakstošu profila attēlu. Lietotāja datu plūsmas diagrammu skatīt zemāk.



8. attēls. Lietotāja datu plūsma

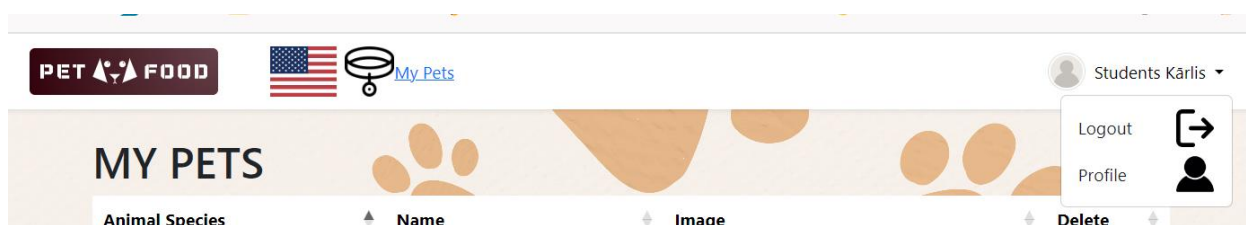
## 6. REAGĒJOŠA TĪMEKĻA VIETNE

Reagējoša tīmekļa vietne ir tieši saistīta ar pieejamību. Tas nozīmē, ka vietnei ir jābūt pieejamai dažādās ierīcēs ar dažādiem ekrāna izmēriem. Dizainam ir jābūt tādām, lai uz lielākiem ekrāniem pieejamā vieta tiktu lietderīgi izmantota un uz mazākiem ekrāniem saturs būtu nespīests un salasāms. Lai ātri un efektīvi izveidotu reagējošus stilus, šajā gadījumā es visvairāk izmantoju *Bootstrap* klases. Pielietojot tās pareizi, var panākt, ka vietne uzreiz ir reagējoša, izņemot gadījumu kad vēlas citādāku izkārtojumu.

Pirmais reagējošais stils tika uztaisīts vietnes navigācijai, kas ir pati svarīgākā vietnes sadaļa. Skatīt 9., 10. un 11. attēlu.

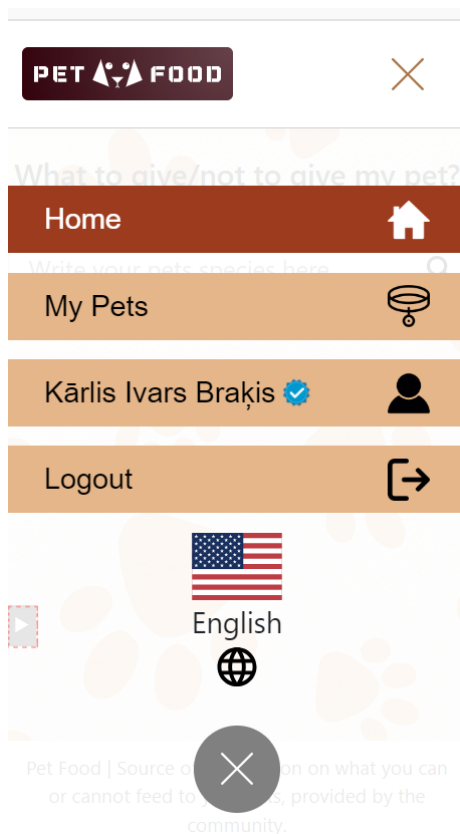


9. attēls. Datora navigācija

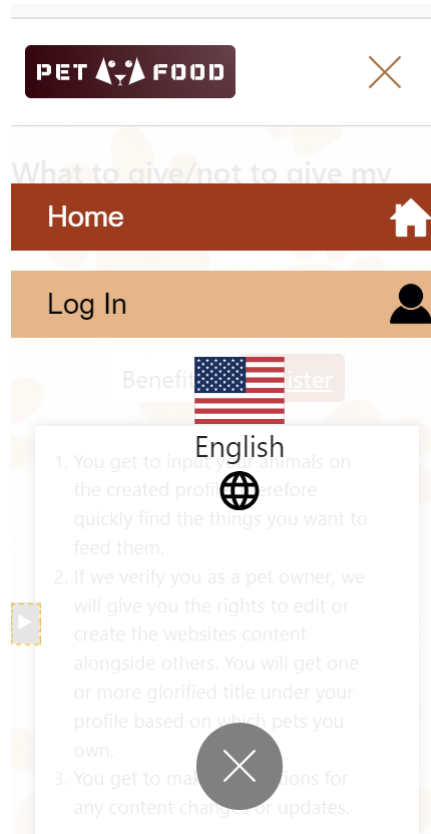


10. attēls. Datora navigācija kad lietotājs ir autorizējies

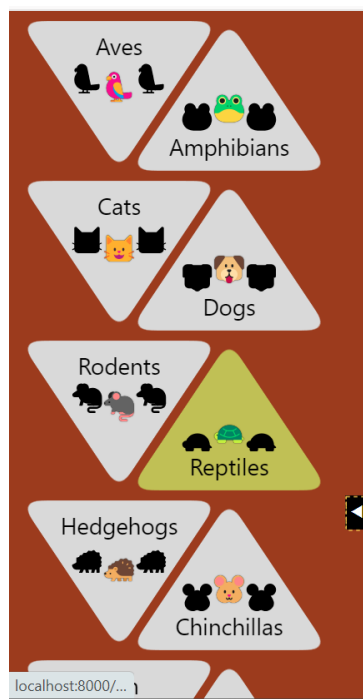
A.



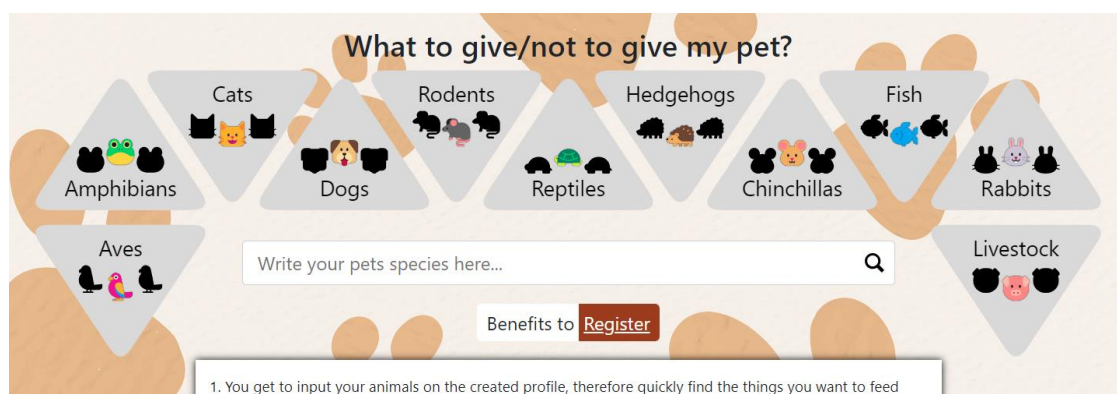
B.



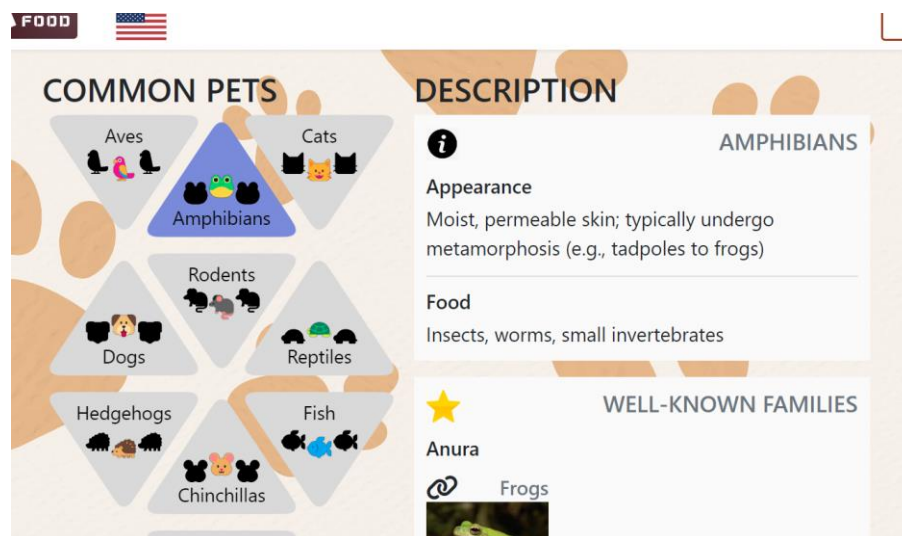
11. attēls. Mazāku ekrāna izmēru navigācija A) kad lietotājs ir autorizējies, B) kad lietotājs nav autorizējies



12. attēls. Populārāko sugu kategoriju izvēle mazākos ekrānos



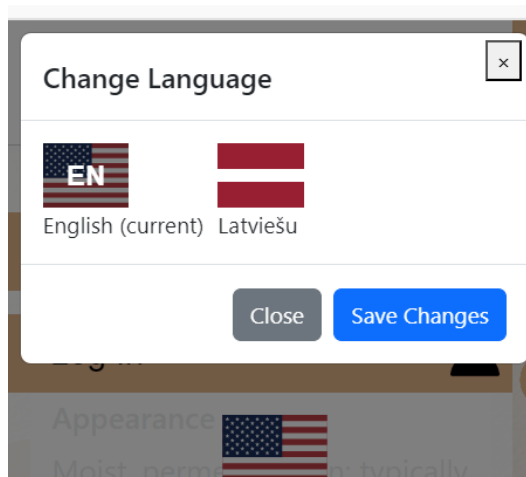
13. attēls. Populārāko sugu kategoriju izvēle uz datora (sākuma skatā)



14. attēls. Konkrētās kategorijas skats uz datora



15. attēls. Valodas izvēlne uz datora



16. attēls. Valodas izvēlne mazākos ekrānos



## 7. TESTĒŠANA

Testēšana ir būtisks solis tīmekļa izstrādē, pārbaudot lietojumprogrammas funkcionalitāti un uzticamību. Tas nodrošina gan funkcionālo (piemēram, reģistrācija, pieteikšanās), gan nefunkcionālo (piemēram, dizains, šifrēšana) prasību izpildi. Testēšana ir neaizstājama, lai identificētu un atrisinātu iespējamās problēmas, uzlabotu lietojumprogrammas vispārējo kvalitāti un nodrošinātu nevainojamu lietotāja pieredzi. Tiek ieteikts izveidot automātiskos testus, piemēram PHPUnit (S. Bergmann, 2024). Tos var palaist izmantojot komandrindu. Tas ir ātrs un efektīvs veids kā pārbaudīt vai vispārējās vietnes funkcijas pienācīgi darbojas.

### 7.1 TESTĒŠANAS NOSACĪJUMI

- Funkcionālā testēšana:** Pārbaudīt, vai visas funkcijas darbojas pareizi, piemēram, vai datu bāze pareizi glabā informāciju, vai lietotāji var meklēt sugas, vai autentifikācija un autorizācija darbojas pareizi utt.
- Lietotāja saskarnes testēšana:** Pārbaudīt, vai lietotāja saskarne ir intuitīva un viegli lietojama. Vai lietotāji var viegli atrast un izmantot visas funkcijas?
- Veiktspējas testēšana:** Pārbaudīt, kā tīmekļa vietne darbojas pie lielas slodzes. Vai tā joprojām darbojas pareizi, ja vienlaikus tiek veiktas daudzas rezervācijas?
- Drošības testēšana:** Pārbaudīt, vai jūsu tīmekļa vietne ir droša. Vai dati tiek pareizi šifrēti? Vai lietotāji nevar veikt neautorizētas darbības?
- Kļūdu pārvaldības testēšana:** Pārbaudīt, kā tīmekļa vietne reaģē uz kļūdām. Vai lietotāji saņem saprotamus kļūdu paziņojumus? Vai kļūdas tiek pareizi reģistrētas?
- Pārlūkošanas savietojamības testēšana:** Pārbaudīt, kā tīmekļa vietne darbojas dažādos pārlūkos un ierīcēs. Vai tā izskatās un darbojas pareizi visos atbalstītos pārlūkos?

6. tabula

Autorizācijas testēšana

N. p. k.	Priekšnosacījums	Scenārijs	Vēlamais rezultāts	Iegūtais rezultāts
1.	Lietotāja reģistrācija.	Letotājs uzspiež uz navigācijas pogas "Ienākt".	Parādās ielogošanās modālais logs.	Pozitīvs.
2.	Lietotāja reģistrācija. » Lietotājs ir atvēris ielogošanās modālo logu.	Letotājs uzspiež uz teksta "Reģistrēties".	Ielogošanās modālais logs tiek nomainīts uz reģistrāciju.	Pozitīvs.

3.	Lietotāja reģistrācija.	Lietotājs aizpilda reģistrācijas ievadlaukus bez kļūdām.	Tiek izveidots lietotāja profils un lietotāju pārvirza uz sākuma lapu.	Pozitīvs (skatīt 17., 18. attēlus).
4.	Lietotāja reģistrācija ar nepareizu e-pastu.	Lietotājs ieraksta e-pastu bez “@” zīmes.	Zem e-pasta ievadlauka parādās kļūdas ziņojums “Nepareizs e-pasta formāts.” (angļu v. <i>Invalid email format.</i> )	Pozitīvs (skatīt 19. attēlu).
5.	Lietotāja reģistrācija ar tukšiem ievadlaukiem.	Lietotājs neaizpilda nevienu reģistrācijas ievadlauku reģistrācijas ievadē.	Zem katra tukšā lauka parādās kļūdas ziņojums “Ievadlauks nevar būt tukšs.” (angļu v. <i>This field is required.</i> )	Pozitīvs (skatīt 19. attēlu).
6.	Lietotāja reģistrācija ar pārāk garu ievadi.	Lietotājs ievadlaukā ieraksta vairāk par 64 rakstzīmēm.	Zem ievadlauka parādās kļūdas ziņojums “Ievade nedrīkst pārsniegt 64 rakstzīmes.” (angļu v. <i>The name must not exceed 64 characters.</i> )	Pozitīvs (skatīt 20. attēlu).
7.	Lietotāja reģistrācija ar nedrošu paroli.	Lietotājs ievada paroli, kura ir īsāka par astoņiem cipariem.	Zem paroles ievadlauka parādās kļūdas ziņojums “Parolē jābūt vismaz 8 rakstzīmēm.” (angļu v. <i>The password must be at least 8 characters long.</i> )	Pozitīvs (skatīt 21. attēlu).
8.	Lietotāja reģistrācija ar dažādām parolēm.	Lietotājs ievada citādāku paroli apstiprināšanas ievadlaukā.	Zem paroles ievadlauka parādās kļūdas ziņojums “Paroles nesakrīt.” (angļu v. <i>The password confirmation does not match.</i> )	Pozitīvs (skatīt 22. attēlu).
9.	Reģistrācija vai ienākšana caur pogām “Sign-up with Google” vai “Login with Google” (darbojas arī otrādāk, ja mēģina reģistrēties ar eksistējošu profilu).	Lietotājs izvēlas savu google profilu autorizācijai vietnē.	Lietotājs tiek pārvirzīts uz sākuma skatu “/home”.	Pozitīvs.

10.	Jebkāda veida reģistrācija ar aizņemtu un verificētu e-pastu.	Lietotājs izvēlas aizņemtu e-pastu un mēģina reģistrēties.	Zem e-pasta ievadlauka parādās kļūdas ziņojums “Šis e-pasts jau ir aizņemts” (angļu v. <i>This email is taken.</i> )	Pozitīvs (skatīt 23. attēlu).
11.	Lietotājs vēlas ielogoties.	Lietotājs korekti ievada savus akreditācijas datus un uzspiež pogu “Ienākt”.	Lietotājs tiek pārvirzīts uz sākuma skatu “/home”.	Pozitīvs.
12.	Lietotājs vēlas ielogoties.	Lietotājs vai ļaunprātīgs lietotājs ieraksta nekorektus akreditācijas datus un uzspiež pogu “Ienākt”.	Lietotājs tiek pārvirzīts uz to pašu skatu ar kļūdas ziņojumu “Šie akreditācijas dati neatbilst mūsu ierakstiem.” (angļu v. <i>These credentials do not match our records.</i> )	Pozitīvs (skatīt 24. attēlu).
13.	Lietotājs ir aizmirsis savu izveidoto paroli. Tika uzspiests uz teksta “Aizmirsi paroli?” un lietotājs ir pārvirzīts uz paroles atjaunošanas skatu.	Lietotājs ievada savu e-pastu un nosūta pieprasījumu atjaunot paroli, uzspiežot pogu “Send Password Reset Link”.	Uz ievadīto e-pastu tiek atsūtīta gan poga, gan saite uz paroles nomaiņas skatu.	Pozitīvs (skatīt 27., 28. attēlus).
14.	Lietotājs atrodas paroles atjaunošanas skatā.	Lietotājs ievada savu e-pastu un nosūta vairākus pieprasījumus atjaunot paroli.	Tiek aizsūtīts tikai viens e-pasts un pārējie tiek ignorēti. Parādās kļūdas ziņojums “Lūdzu uzgaidiet, pirms mēģiniet vēlreiz” (angļu v. <i>Please wait before retrying.</i> )	Pozitīvs (skatīt 29. attēlu).
15.	Lietotājs ir saņēmis e-pastu par paroles nomaiņu.	Lietotājs uzspiež uz norādītās pogas vai saites.	Lietotājs tiek pārvirzīts uz paroles nomaiņas skatu.	Pozitīvs.
16.	Lietotājs atrodas paroles nomaiņas skatā pēc e-pasta saņemšanas.	Lietotājs izdomā un ievada savu jauno paroli	Lietotājs tiek pārvirzīts uz sākuma skatu “/home” un turpmāk var izmantot jauno paroli.	Pozitīvs.

17.	Lietotājs ir ielogojies un vēlas nomainīt paroli. Ir atvērts paroles atjaunošanas modālais logs profila skatā.	Lietotājs korekti ievada savu pašreizējo paroli, jauno paroli, atkārtoti jauno paroli un uzspiež pogu "Change Password"	Lietotājs tiek pārvirzīts uz profila skatu "/profile" ar paziņojumu "Parole ir nomainīta veiksmīgi." (angļu v. <i>Password updated successfully.</i> ) un turpmāk var izmantot jauno paroli.	Pozitīvs (skatīt 25. attēlu).
18.	Lietotājs vēlas izlogoties.	Lietotājs atrod pogu "Logout" iekš navigācijas.	Lietotājs tiek pārvirzīts uz sākuma skatu.	Pozitīvs.

7. tabula

Galvenās funkcionalitātes testēšana

N. p. k.	Priekšnosacījums	Scenārijs	Vēlamais rezultāts	Iegūtais rezultāts
1.	Sugas meklēšana.	Lietotājs ieraksta atslēgvārdu "varde".	Tiek parādīti atbilstošākie rezultāti no datubāzes. Uzspiežot uz vēlamā rezultāta lietotājs tiek pārvirzīts uz ēdienu meklēšanas skatu.	Pozitīvs (skatīt 30. attēlu).
2.	Ēdienu drošības pakāpes vai nekaitīguma meklēšana.	Lietotājs meklētājā ieraksta atslēgvārdu "muša".	Tiek parādīts tieši mušas vai citi līdzīgi ieraksti no datu bāzes un atbilstošā drošības kategorija – "Droši barot". Papildus tam, parādās krustiņa ikona, ar kuru var notīrīt ievadlauku.	Pozitīvs (skatīt 31. attēlu).
3.	Ēdienu drošības pakāpes vai nekaitīguma meklēšana. » Lietotājs meklētājā ierakstījis jebko.	Lietotājs uzspiež krustiņa ikonu, lai notīrītu ievadi.	Tiek izdzēsts viss ievadē ierakstītais un atjaunojas visi meklēšanas rezultāti.	Pozitīvs.
4.	Ēdienu drošības pakāpes vai nekaitīguma meklēšana. » Lietotājs vēlas redzēt tikai droši barojamos	Lietotājs atzīmē "Droši barot"	Atjaunojas visi meklēšanas rezultāti ar pievienoto filtra opciju. Kamēr tiek saņemta atbilde no aizmugursistēmas, rādās ielādes animācija	Pozitīvs.

	pārtikas produktus.		un ievadlauks tiek atspējots, lai novērstu servera noslodzi un netiktu izpildīti lieki API vaicājumi.	
5.	Ēdienu drošības pakāpes vai nekaitīguma meklēšana.	Lietotājs ierasta tikai vienu burtu meklētājā.	Atjaunojas visi meklēšanas rezultāti, tiek atlasīti dati, kuri sākas ar norādīto burtu.	Pozitīvs (skatīt 32. attēlu).
6.	Ēdienu drošības pakāpes vai nekaitīguma meklēšana.	Lietotājs ieraksta frāzi kura neatrodas datu bāzē - “frāze kura neatrodas datubāzē”.	Rezultātu vietā parādās ziņojums “Pēc frāzes <b>frāze kura neatrodas datubāzē</b> , nav atrasts neviens rezultāts.” (angļu v. <i>No data found for phrase: frāze, kura neatrodas datubāzē.</i> )	Pozitīvs (skatīt 33. attēlu).
7.	Ēdienu drošības pakāpes vai nekaitīguma meklēšana. » Lietotājs vēlas redzēt vairāk datu.	Lietotājs uzspiež pogu “Redzēt vēl”.	Tiek atlasīti papildus dati, izmantojot paginācijas principu.	Pozitīvs (skatīt 34. attēlu).

8. tabula

Autorizēta lietotāja funkciju testēšana

N. p. k.	Priekšnosacījums	Scenārijs	Vēlamais rezultāts	Iegūtais rezultāts
1.	Lietotāja datu rediģēšana. Lietotājs vēlas nomainīt savu lietotājvārdu vai vārdu.	Lietotājs ievada citādāku lietotājvārdu vai pilno vārdu un uzspiež pogu “Saglabāt izmaiņas”.	Tiek atjaunināti profila dati datubāzē un parādās ziņojums “Profila dati ir atjaunoti.” (angļu v. <i>Profile updated successfully.</i> )	Pozitīvs (skatīt 35. attēlu).
2.	Lietotāja datu rediģēšana. Lietotājs vēlas iestatīt savu vārdu kā publiski redzamu.	Lietotājs atzīmē ievades lauku “Iestatīt par publiski redzamu” un uzspiež pogu “Saglabāt izmaiņas”.	Tiek atjaunināti profila dati datubāzē un parādās ziņojums “Profila dati ir atjaunoti.” (angļu v. <i>Profile updated successfully.</i> ), kā arī skata labajā augšējā stūrī ir redzams savs pilnais vārds.	Pozitīvs (skatīt 36. attēlu).

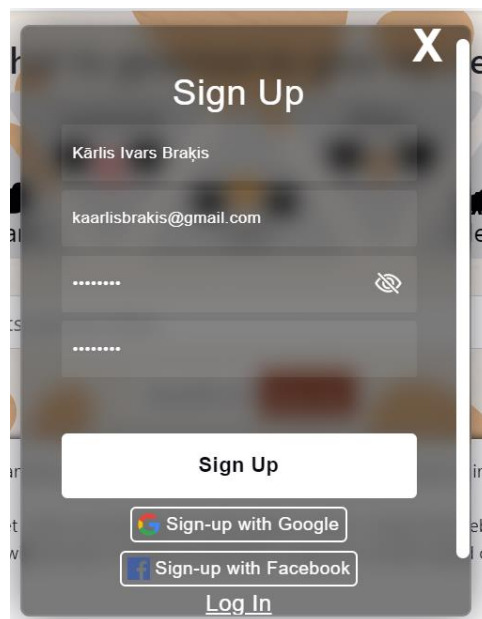
3.	Lietotāja datu rediģēšana. Lietotājs vēlas nomainīt e-pastu. Lietotāja e-pasts jau ir bijis verificēts.	Lietotājs ieraksta citādāku e-pastu un uzspiež pogu “Saglabāt izmaiņas”.	E-pasts tiek veiksmīgi nomainīts un parādās ziņojums “Profila dati ir atjaunoti.” (angļu v. <i>Profile updated successfully.</i> ). Zem e-pasta lauka “Verificēts” teksta vietā parādās saite, uz kuras uzspiežot var verificēt e-pastu.	Pozitīvs (skatīt 37. attēlu).
4.	Lietotāja datu rediģēšana. Lietotājs vēlas nomainīt profila bildi.	Lietotājs izvēlas attēlu un uzspiež pogu “Augšupielādēt”.	Tiek ievietots attēls failu sistēmā un faila nosaukums datubāzē. Parādās ziņojums “Profila bilde ir veiksmīgi nomainīta.” (angļu v. <i>Profile picture updated successfully.</i> ).	Pozitīvs (skatīt 38., 39. attēlus).
5.	Mājdzīvnieku pievienošana profilam.	Lietotājs ievadē izvēlas vienu no pieejamajām sugām, ievada mājdzīvnieka vārdu, izvēlas bildi un uzspiež pogu “Add Pet”.	Tiek ievietoti mājdzīvnieka dati datubāzē un attēls failu sistēmā. Parādās ziņojums “Mājdzīvnieks ir pievienots!” (angļu v. <i>Pet added successfully!</i> ).	Pozitīvs (skatīt 40., 41. attēlus).
6.	Mājdzīvnieku dzēšana no profila.	Lietotājs uzspiež pogu “Delete” vienam no sarakstā redzamajiem mājdzīvniekiem un pēc brīdinājuma apstiprināšanas uzspiež “Ok”.	Tiek izdzēsts mājdzīvnieka ieraksts no datubāzes un . parādās ziņojums “Mājdzīvnieka ieraksts ir izdzēsts!” (angļu v. <i>Pet deleted successfully!</i> ).	Pozitīvs (skatīt 42. attēlu).

9. tabula

#### Administratora funkciju testēšana

N. p. k.	Priekšnosacījums	Scenārijs	Vēlamais rezultāts	Iegūtais rezultāts
1.	Dzīvnieka sugas ievietošana.	Lietotājs ieraksta sameklēto GBIF ID pirmajā ievadlaukā un	Tiek automātiski aizpildīta daļa no ievadformas.	Pozitīvs (skatīt 43. attēlu).

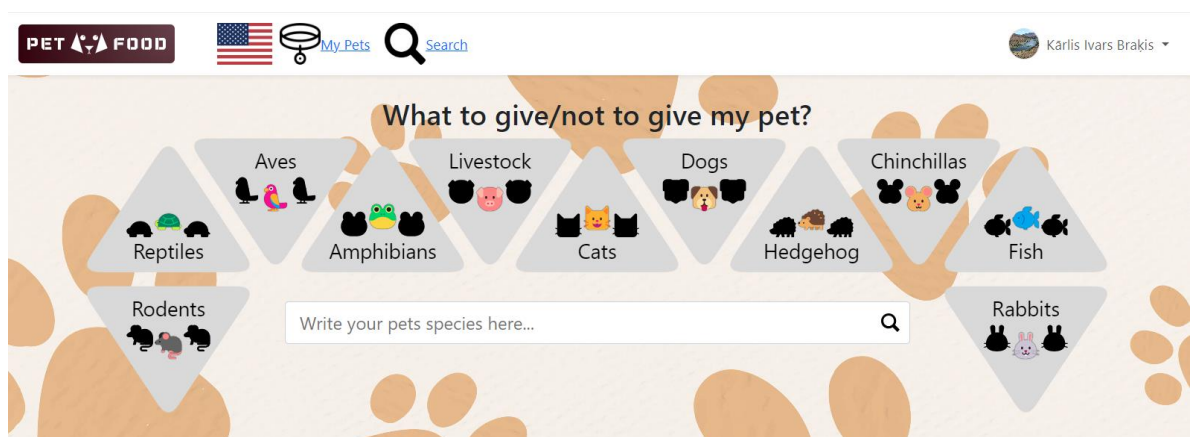
		uzspiež pogu "Autofill".		
2.	Dzīvnieka sugas ievietošana.	Lietotājs ir ierakstījis vismaz obligātos ievadlaukus un uzspiedis pogu "Submit"	Tiek augšupielādētas visas bildes un ievietots ieraksts datubāzē. Parādās ziņojums "Mājdzīvnieks ir pievienots!" (angļu v. <i>Pet added successfully!</i> ).	Pozitīvs (skatīt 44., 45. attēlus).
3.	Dzīvnieka sugas ievietošana.	Lietotājs ir ierakstījis GBIF ID ievadlaukā jau eksistējošu numuru un izvēlējis valodu, kurai jau ir šis numurs.	Parādās kļūdas ziņojums "Mājdzīvnieks ar šādu numuru un lokalizāciju jau eksistē." (angļu v. <i>A pet with the same GBIF ID and locale already exists.</i> )	Pozitīvs (skatīt 46. attēlu).
4.	Ēdiena vai pārtikas produkta nosaukuma ievietošana.	Lietotājs ir izvēlējis valodu un ierakstījis ēdiena nosaukumu.	Tiek ievietots ieraksts datubāzē. Parādās ziņojums "Produkta nosaukums ir pievienots!" (angļu v. <i>Food record created successfully!</i> ).	Pozitīvs (skatīt 47., 48. attēlus).
5.	Ēdiena nosaukuma un dzīvnieku sugu sasaiste ar drošības kategoriju.	Lietotājs ir izvēlējis dzīvnieka sugu, ēdiena nosaukumu un drošības kategoriju.	Tiek ievietots ieraksts datubāzē. Parādās ziņojums "Drošības kategorijas ieraksts ir pievienots!" (angļu v. <i>Safety record created successfully!</i> ).	Pozitīvs (skatīt 49., 50. attēlus).
6.	Ēdiena nosaukuma un dzīvnieku sugu sasaiste ar drošības kategoriju.	Lietotājs ir aizpildījis visus ievadlaukus, bet visi sakrīt ar eksistējošu ierakstu datubāzē.	Parādās kļūdas ziņojums "Tieši šāds ieraksts jau pastāv!" (angļu v. <i>A safety record with the same entry already exists!</i> )	Pozitīvs (skatīt 51. attēlu).



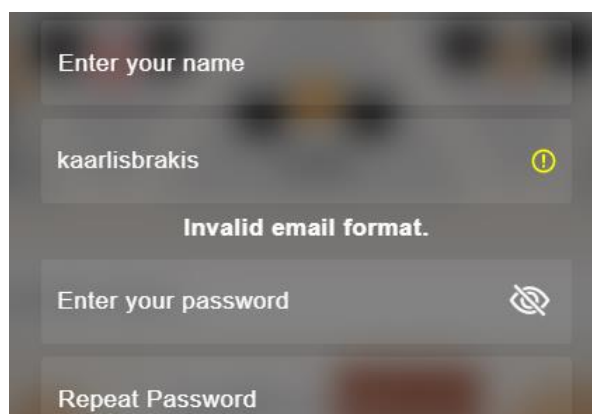
A dark-themed modal window titled "Sign Up" with a close button (X) in the top right corner. It contains the following fields and buttons:

- Text input: Kārlis Ivars Brāķis
- Text input: kaarlisbrakis@gmail.com
- Password input: ..... (with an eye icon for visibility toggle)
- Repeat password input: .....
- Primary button: Sign Up
- Social login buttons: Sign-up with Google, Sign-up with Facebook
- Link: [Log In](#)

17. attēls. Reģistrācija



18. attēls. Sākumlapa



A dark-themed sign-up form with the following elements:

- Text input: Enter your name
- Text input: kaarlisbrakis (with a yellow warning icon)
- Error message: Invalid email format.
- Password input: Enter your password (with an eye icon)
- Text input: Repeat Password

19. attēls. Reģistrācija ar nepareizu e-pastu



A screenshot of a mobile application's 'Sign Up' screen. The screen has a dark background with a white 'X' in the top right corner. The title 'Sign Up' is centered at the top. Below it are four input fields: 'Enter your name', 'Enter your email', 'Enter your password', and 'Repeat Password'. Each of the first three fields has a yellow warning icon (an exclamation mark inside a circle) to its right. Below the 'Enter your name' field, the text 'The name field is required.' is displayed. Below the 'Enter your email' field, the text 'The email field is required.' is displayed. Below the 'Enter your password' field, the text 'The password field is required.' is displayed. At the bottom of the form is a white button with the text 'Sign Up'.

19. attēls. Reģistrācija ar tukšiem ievadlaukiem

A screenshot of the 'Sign Up' screen, showing the 'Enter your name' field. The field contains a long string of characters. To the right of the field is a yellow warning icon. Below the field, the text 'The name must not exceed 64 characters.' is displayed. The other fields and the 'Sign Up' button are not visible in this crop.

20. attēls. Reģistrācija ar nesakarīgi garu ievadi

A screenshot of the 'Sign Up' screen. The 'Enter your name' field contains the text 'Kārlis Ivars Braķis'. The 'Enter your email' field contains the text 'kaarlisbrakis@gmail.com'. The 'Enter your password' field contains a password (represented by dots) and has a yellow warning icon to its right. Below this field, the text 'The password must be at least 8 characters long.' is displayed. The 'Repeat Password' field is empty. The 'Sign Up' button is at the bottom.

21. attēls. Reģistrācija ar nedrošu paroli

A screenshot of a 'Sign Up' form. At the top right is a close button 'X'. The form contains three input fields: the first contains 'Kārlis Ivars Braķis', the second contains 'kaarlisbrakis@gmail.com', and the third is labeled 'Enter your password' with a toggle icon. Below the password field, a message states 'The password confirmation does not match.' Below this is a 'Repeat Password' field.

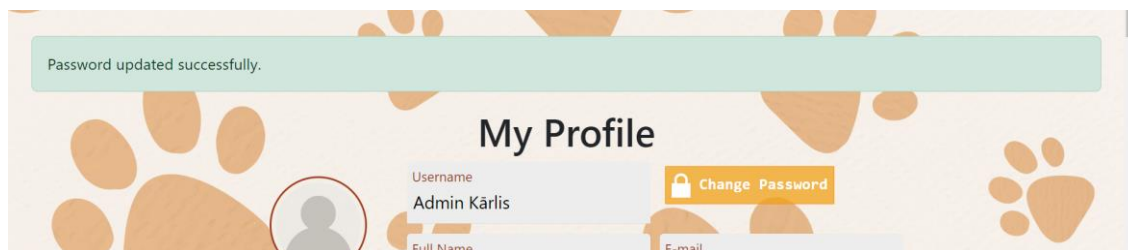
22. attēls. Reģistrācija ar dažādām parolēm

A screenshot of a 'Sign Up' form. It has three input fields: 'Enter your name', 'Enter your email' (with a yellow warning icon), and 'Enter your password' (with a toggle icon). A message below the email field says 'This email is already taken.'

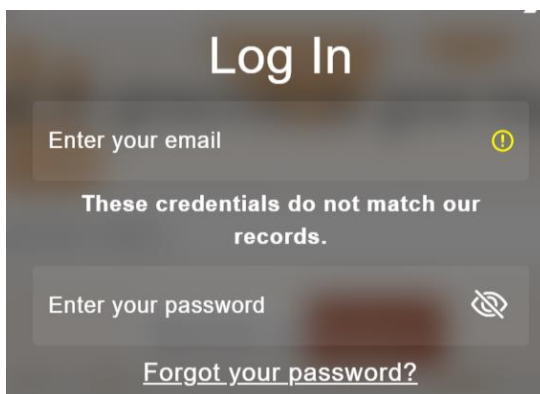
23. attēls. Reģistrācija ar aizņemtu e-pastu

A screenshot of a 'Log In' form. It has two input fields: the first contains 'kaarlisbrakis@gmail.com' (with a yellow warning icon) and the second is labeled 'Enter your password' (with a toggle icon). A message between the fields states 'These credentials do not match our records.' At the bottom is a link that says 'Forgot your password?'.

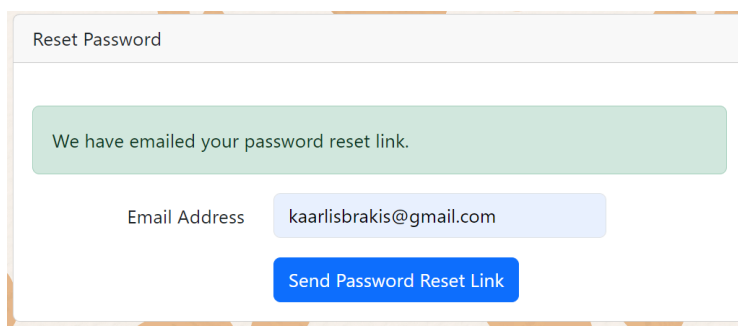
24. attēls. Ielogošanās ar nekorektiem akreditācijas datiem



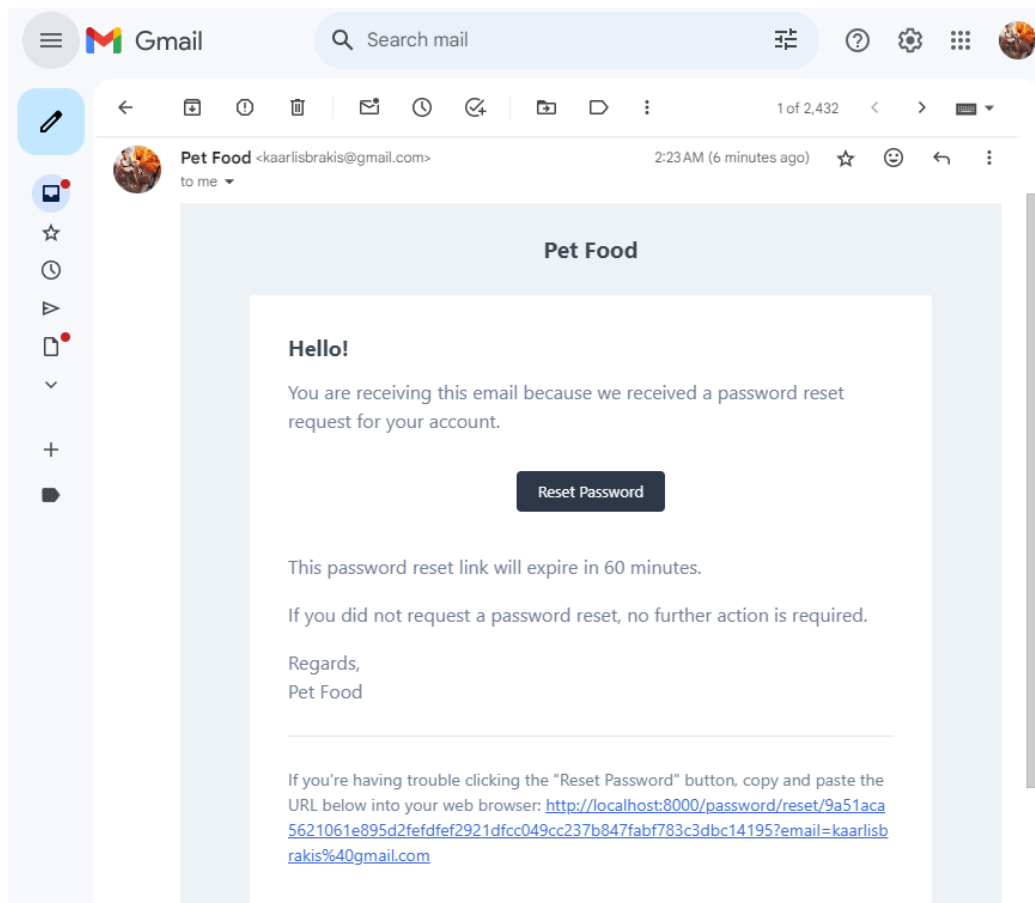
25. attēls. Parole ir veiksmīgi nomainīta.



26. attēls. Ielogošanās mēģinājums ar nekorektiem akreditācijas datiem



27. attēls. Paroles atjaunošanas pieprasījuma nosūtīšana



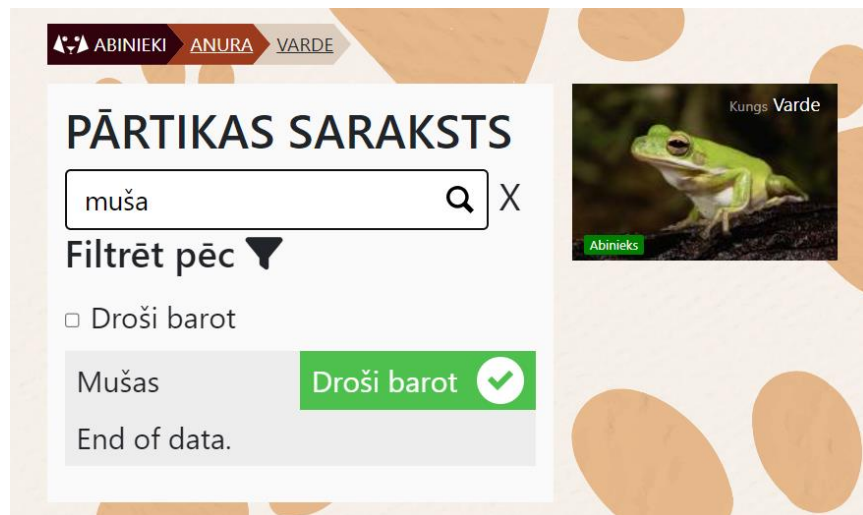
28. attēls. Paroles atjaunošanas e-pasts

The screenshot shows a 'Reset Password' form. The 'Email Address' field contains 'kaarlisbrakis@gmail.com' and has a yellow warning icon. Below the field, the text 'Please wait before retrying.' is displayed in red. A blue button labeled 'Send Password Reset Link' is positioned below the warning message.

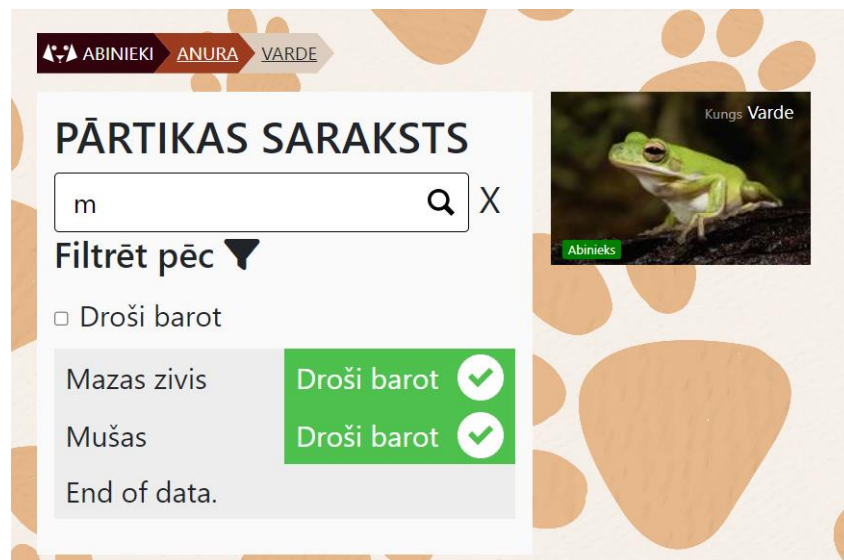
29. attēls. Atkārtotu e-pasta sūtījumu kļūdas ziņojums.

The screenshot shows a search bar with the word 'varde' entered. Below the search bar, the word 'Vardes' is displayed, suggesting a search result or suggestion.

30. attēls Sugu meklēšana



31. attēls. Ēdienu drošības kategoriju meklēšana, ierakstot atslēgvārdu



32. attēls. Ēdienu drošības kategoriju meklēšana, ierakstot vienu burtu



33. attēls Neatrasta frāze

A.

Zemes tārpi	Droši barot	✓
Mazas zivis	Droši barot	✓
Kurkuļi	Droši barot	✓
Zirnekļi	Droši barot	✓
Skorpioni	Droši barot	✓
Čūskas	Droši barot	✓

▼ RĀDĪT VĒL ▼

B.

Kurkuļi	Droši barot	✓
Zirnekļi	Droši barot	✓
Skorpioni	Droši barot	✓
Čūskas	Droši barot	✓
Gliemji	Droši barot	✓
Peles	Droši barot	✓
End of data.		

34. attēls. Papildus datu pieprasīšana A) Redzami dati pirms pogas uzspiešanas. B) Redzami dati pēc pogas uzspiešanas.

PET FOOD My Pets Admin Kārlis

Profile updated successfully.

### Mans profils

Vietturis

Izvēlēties attēlu

Lietotājsvārds  
Admin Kārlis

Pilns vārds  
Kārlis Ivars Brāķis

Iestatīt par publiski redzamu

Saglabāt izmaiņas

Mainīt paroli

E-pasts  
kaarlisbrakis@gmail.com

Verificēts

35. attēls. Profila datu rediģēšana

PET FOOD My Pets Kārlis Ivars Brāķis

Profile updated successfully.

### Mans profils

Vietturis

Izvēlēties attēlu

Lietotājsvārds  
Admin Kārlis

Pilns vārds  
Kārlis Ivars Brāķis

✓ Iestatīt par publiski redzamu

Saglabāt izmaiņas

Mainīt paroli

E-pasts  
kaarlisbrakis@gmail.com

Verificēts

36. attēls. Profila vārda iestatīšana publiski

Profile updated successfully.

### My Profile

Placeholder

Choose Image

Username  
Admin Kārlis

Full Name  
Kārlis Ivars Brāķis

✓ Set as display name

Save Changes

Change Password

E-mail  
cits\_epasts@gmail.com

Why/How?



37. attēls. E-pasta maiņa

The screenshot shows the 'My Profile' page with a light orange background and paw print patterns. On the left, there are two image placeholders: 'OLD IMAGE' (a grey silhouette) and 'NEW IMAGE' (a cow's face). Below the old image is a 'Placeholder' label and a 'Choose Image' button. Below the new image is a 'Close-up-cow-nose...' label and an 'Upload' button. To the right of the images, the profile information is displayed: Username 'Admin Kārlis', Full Name 'Kārlis Ivars Braķis', and E-mail 'cits\_epasts@gmail.com'. There is a 'Change Password' button with a lock icon. A checkbox 'Set as display name' is checked. At the bottom right, there are two buttons: 'Not verified!' (blue) and 'How/Why?' (green). At the bottom center, there is a 'Save Changes' button.

38. attēls. Lietotājs ir izvēlējies profila bildi

This screenshot shows the 'My Profile' page after a successful profile picture update. A green notification bar at the top states 'Profile picture updated successfully.' The profile information remains the same: Username 'Admin Kārlis', Full Name 'Kārlis Ivars Braķis', and E-mail 'cits\_epasts@gmail.com'. The 'NEW IMAGE' now shows the cow's face. The 'Choose Image' button is now labeled 'profile\_picture\_1\_'. The 'Save Changes' button is still present at the bottom.

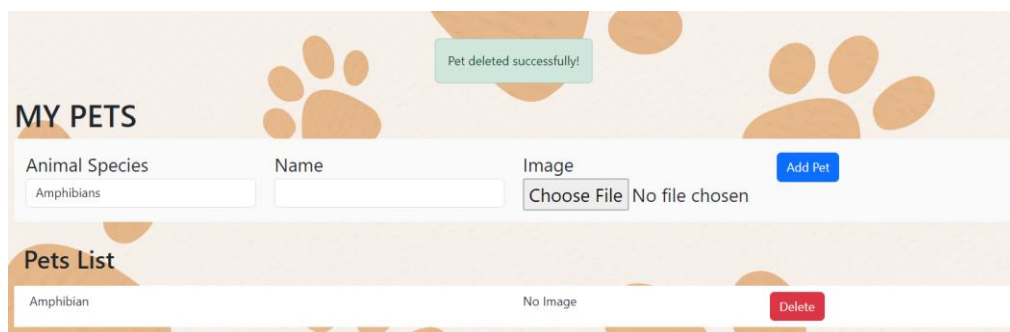
39. attēls. Jaunā profila bilde ir augšupielādēta

The screenshot shows the 'MY PETS' page. It has a form with three fields: 'Animal Species' (a dropdown menu with 'Cattle|Bos' selected), 'Name' (a text input with 'Gotiņa' entered), and 'Image' (a file upload area with a 'Choose File' button and a preview of the cow's face). There is an 'Add Pet' button. Below the form, a message says 'No pets added yet.'

40. attēls. Lietotājs ir ievadījis sava mājdzīvnieka datus

This screenshot shows the 'MY PETS' page after a pet has been added. A green notification bar at the top states 'Pet added successfully!'. The form is now empty, with 'Animal Species' set to 'Amphibians' and 'Name' empty. The 'Image' field shows 'No file chosen'. Below the form, there is a 'Pets List' table. The table has two rows: one for 'Amphibian' with 'No Image' and a 'Delete' button, and one for 'Cattle|Bos' with 'Gotiņa' as the name, a preview of the cow's face, and a 'Delete' button.

41. attēls. Lietotājs ir saglabājis sava mājdzīvnieka datus



Pet deleted successfully!

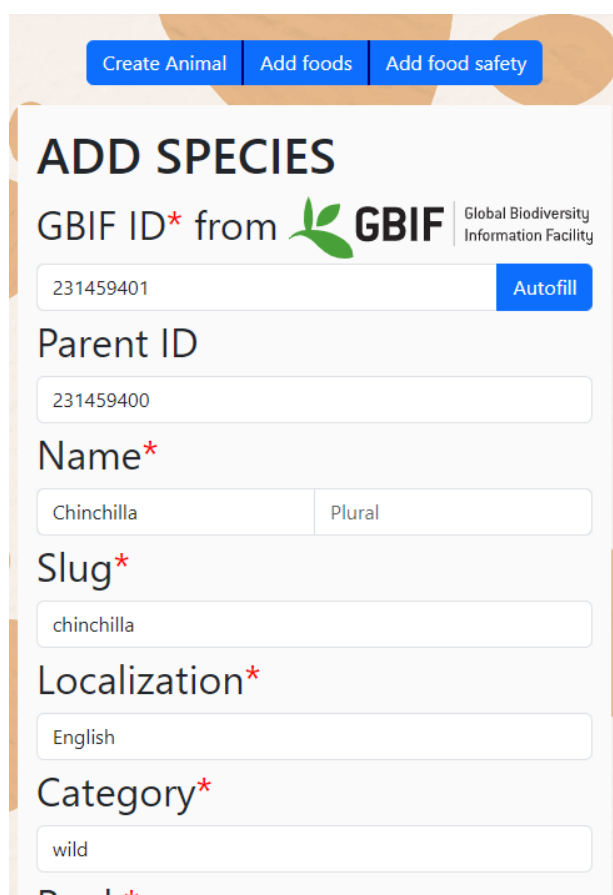
### MY PETS

Animal Species: Amphibians | Name: | Image: Choose File | No file chosen | Add Pet

### Pets List

Amphibian	No Image	Delete
-----------	----------	--------

42. attēls. Lietotājs ir izdzēsis kādu no saviem mājdzīvnieka ierakstiem



Create Animal | Add foods | Add food safety

## ADD SPECIES

GBIF ID\* from GBIF Global Biodiversity Information Facility

231459401 | Autofill

Parent ID

231459400

Name\*

Chinchilla | Plural

Slug\*

chinchilla

Localization\*

English

Category\*


wild

Parent\*

43. attēls. Administrators izmanto automātiskās ievades funkciju



## ADD SPECIES

GBIF ID\* from  Global Biodiversity Information Facility

Parent ID

Name\*

Slug\*

Localization\*

Category\*

Rank\*

Appearance


Food


Images

Allowed: jpeg, png, jpg, and max 2 MB

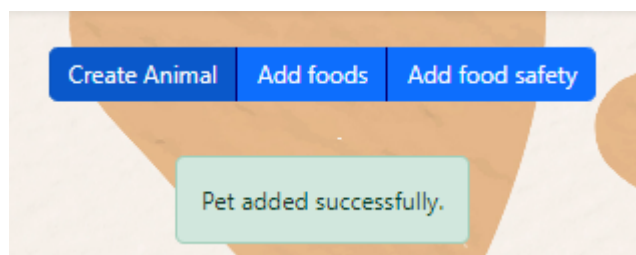
### Preview

Click on any image to set it as the cover image.

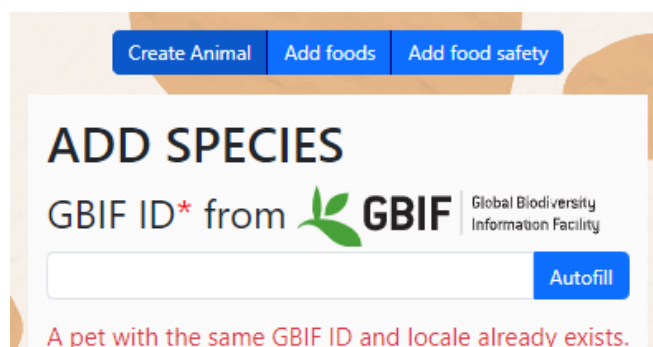




44. attēls. Administrators ir ievadījis kādas sugas datus



45. attēls. Administrators ir pievienojis sugas datus



46. attēls. Aizņemta sugas ieraksta kļūdas ziņojums

Create Animal Add foods Add food safety

## ADD FOODS

Localization\*

English

Food name\*

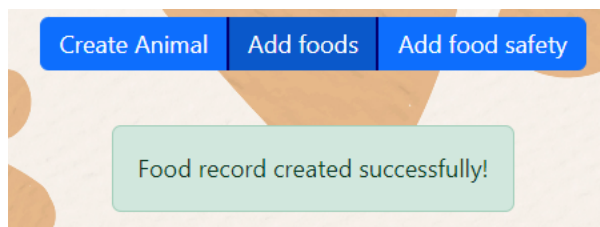
Chocolate

Description

Not suitable for any animal.

Submit

47. attels. Administrators ir ievadījis pārtikas produkta nosaukumu



48. attēls. Administrators ir pievienojis produkta datus

Create Animal Add foods Add food safety

## ADD SAFETIES

Animal Name

Amphibians

Food

Flies

Safety Category

Safe to feed

Safe to feed

Dangerous

Be careful

Unknown

49. attēls. Ēdienu drošības kategoriju pievienošana



50. attēls. Administrators ir pievienojis produkta drošības kategoriju

A screenshot of a web application interface. At the top, there are three blue buttons: "Create Animal", "Add foods", and "Add food safety". Below these buttons, a pink rounded rectangle contains the text "A safety record with the same entry already exists!". Below the error message, there is a white form titled "ADD SAFETIES". The form contains three input fields: "Animal Name" with the text "Amphibians", "Food" with the text "Flies", and "Safety Category" with the text "Safe to feed". At the bottom of the form is a blue "Submit" button. The background is a light beige color with faint orange circular patterns.

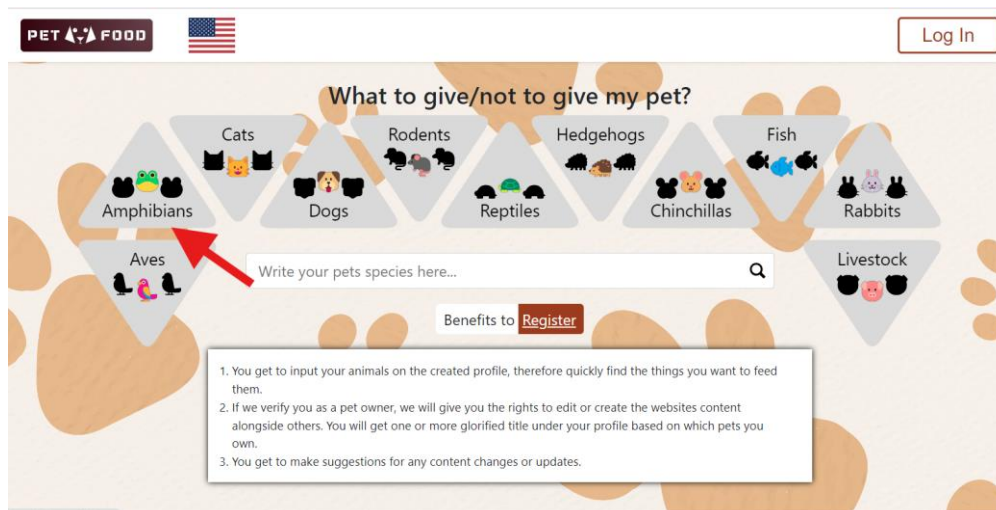
51. attēls. Drošības kategorijas ieraksts jau eksistē

## 8. LIETOTĀJA CEĻVEDIS

Lai izveidotu lietotāja ceļvedi, jāapraksta katra darbība, sākot no reģistrēšanās un pieslēgšanās, līdz pat navigācijai starp lapām. Ir svarīgi, lai katrs solis būtu skaidri un precīzi aprakstīts, ietverot arī ekrānuzņēmumus, kas palīdzēs lietotājiem labāk saprast funkcionalitāti un izmantot tīmekļa vietni pilnvērtīgi.

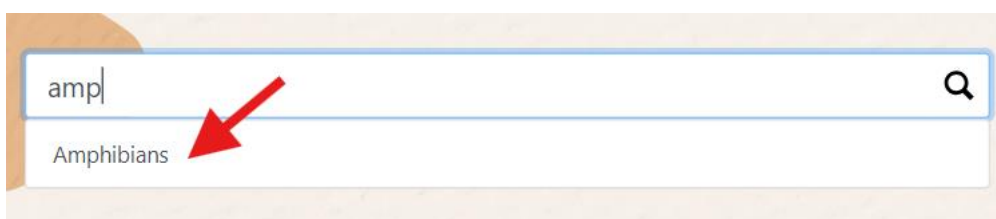
### 8.1 GALVENĀ FUNKCIONALITĀTE

Sākuma skatā var izvēlēties zināmākās sugu kategorijas (skatīt 52. attēlu).



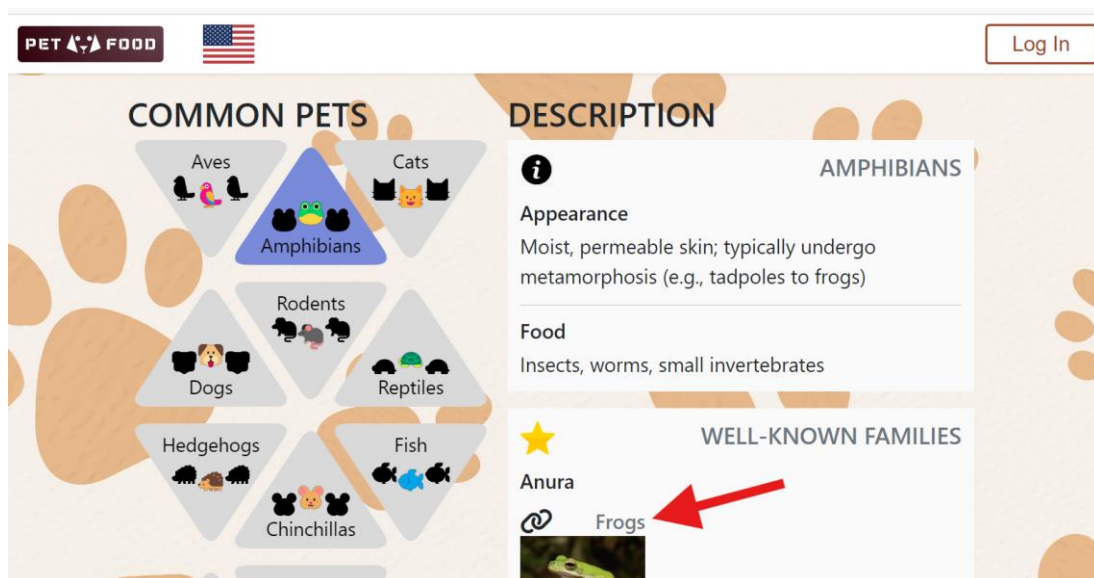
52. attēls. Sākuma skats

Galvenais veids kā meklēt sugas ir rakstīt atslēgvārdu meklētājā un uzspiest uz vēlamā rezultāta (skatīt 53. attēlu).



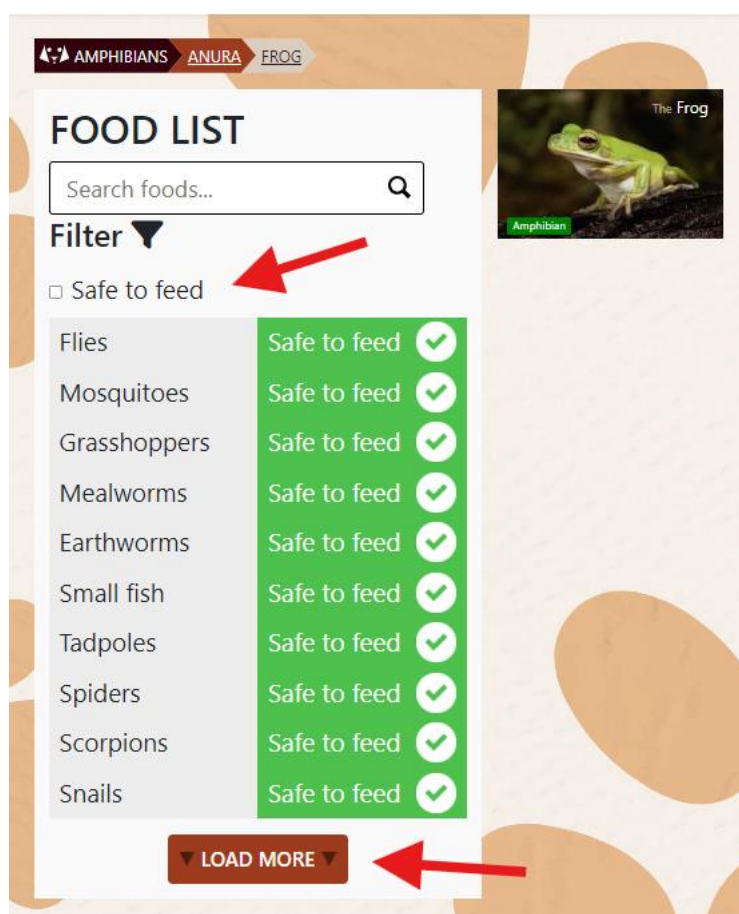
53. attēls. Atslēgvārda ierakstīšana meklētājā

Izvēloties vienu no kategorijām var redzēt vispārīgu aprakstu un konkrētākas sugas. (skatīt 54. attēlu).



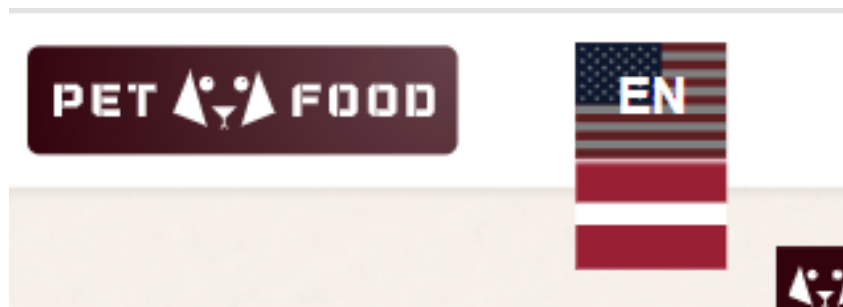
54. attēls. Sugas izvēle

Konkrētās sugas skatā var meklēt ēdienus pēc atslēgvārdiem, kā arī atzīmēt izvēles rūtiņu un filtrēt tikai “droši barojamus” pārtikas produktus (skatīt 55. attēlu).



55. attēls. Ēdienu meklēšana

Vietnes navigācijā ir pieejama valodu izvēlne (skatīt 56. attēlu).



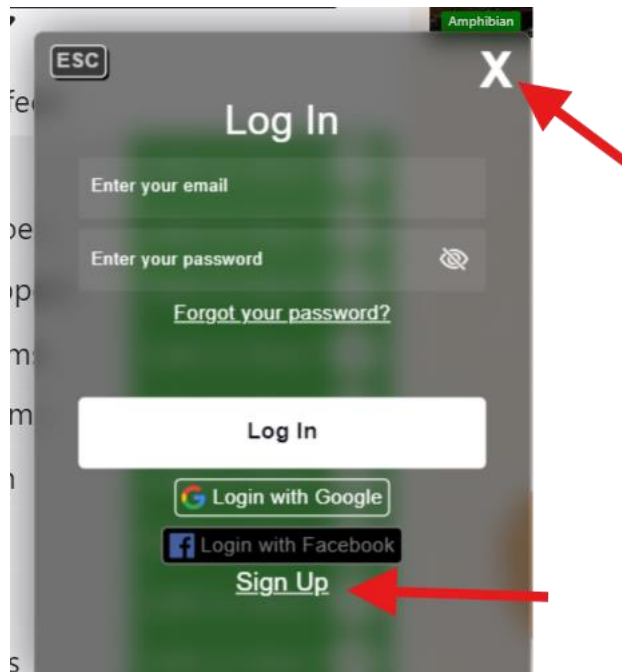
56. attēls. Valodas maiņa navigācijā

Vietnes navigācijā ir pieejama arī ienākšanas poga, kura atver ievadformas modālo logu (skatīt 57. attēlu).



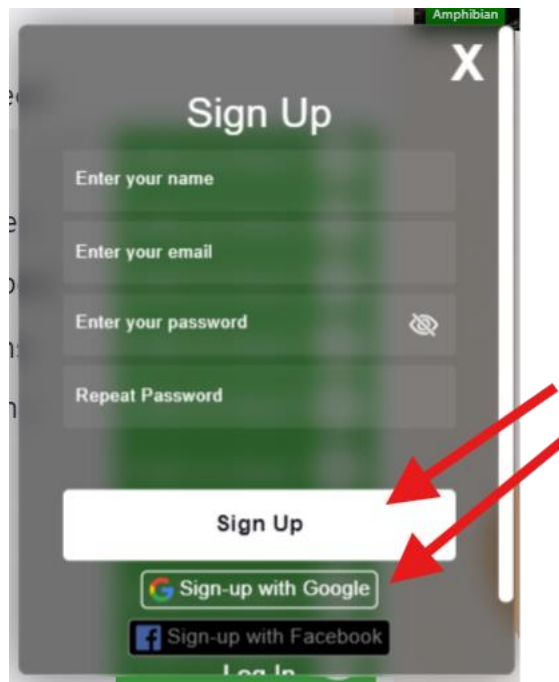
57. attēls. Ienākšanas poga

Ienākšanas ievadformā ir jāievada sava iepriekš uztaisītā profila akreditācijas dati – e-pasts un parole. Pēc tam jānospiež poga “Log In” (skatīt 58. attēlu).



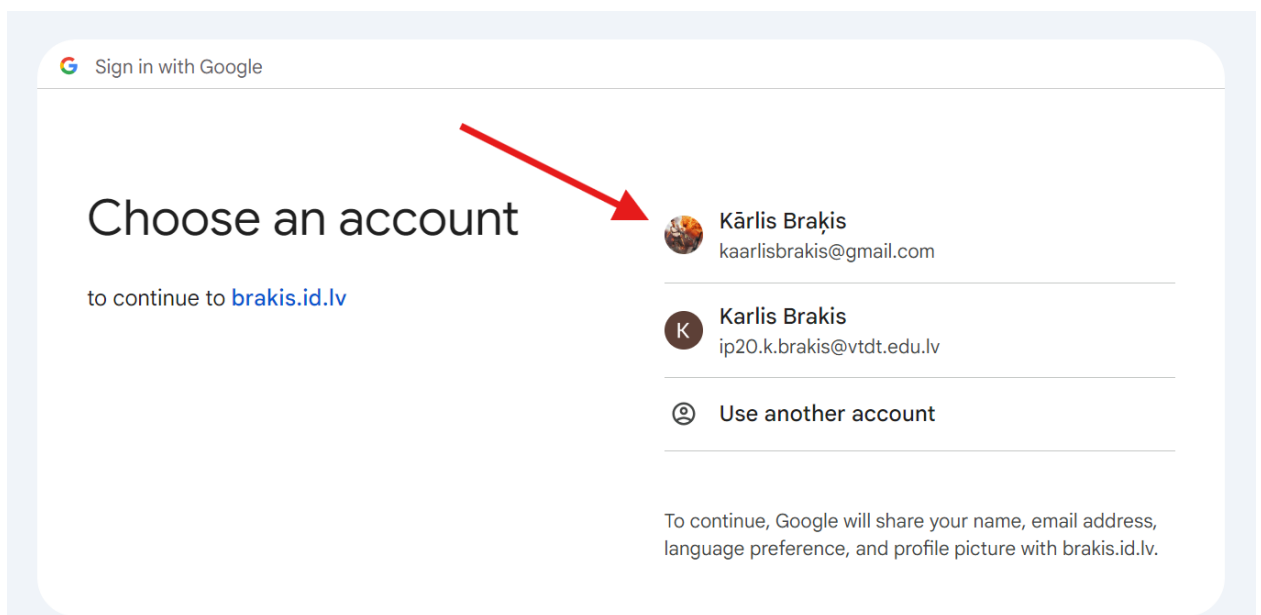
58. attēls. Ienākšanas ievadforma

Reģistrācijas ievadformā ir jāievada savs vārds, e-pasts un parole un parole atkārtoti. Pēc tam jānospiež poga “Sign Up”. Ir iespēja arī izmantot savu Google profilu (skatīt 59. attēlu).



59. attēls. Reģistrācijas ievadforma

Ja nospiež pogu “Log-in with google” vai “Sign-up with google”, tad tiks pārvirzīts uz *Google* autorizācijas skatu. Izvēloties savu *Google* profilu, tiks pārvirzīts uz vietnes sākuma skatu (skatīt 60. attēlu).



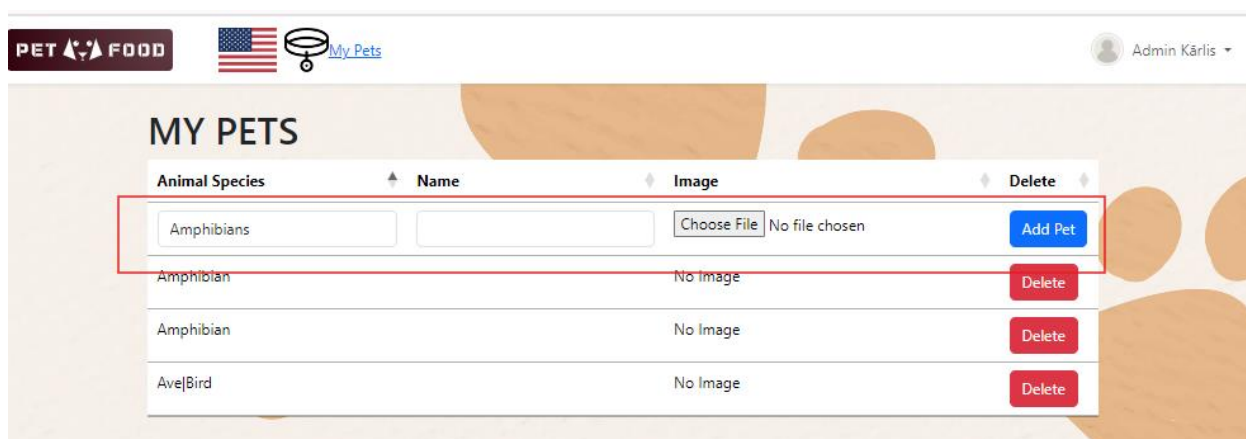
60. attēls. Google autorizācijas skats

## 8.2 LIETOTĀJA SKATI

Ja lietotājs ir ielogojies, tad navigācijā parādās papildus saite uz savu mājdzīvnieku skatu. Pirmajā reizē ir redzama tikai ievadforma bez mājdzīvnieku datiem (skatīt 61., 62. attēlus).



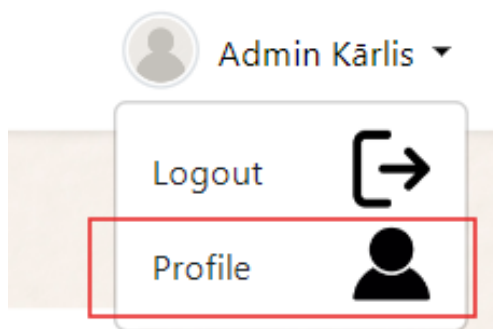
61. attēls. Autorizēta lietotāja navigācijas elements



62. attēls. Autorizēta lietotāja skats

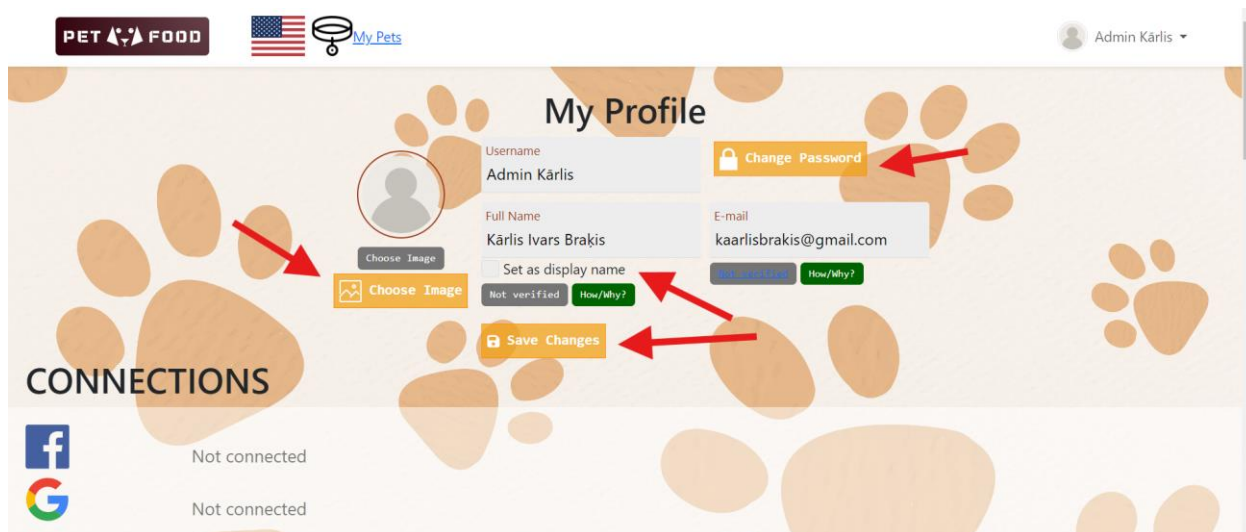


Kad lietotājs ir ielogojies, navigācijā “Log In” pogas vietā atrodas tavs lietotājvārds ar atveramu izvēlni, kur var izlogoties un apskatīt savu profilu (skatīt 63. attēlu).



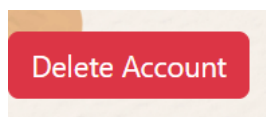
63. attēls. Navigācijas pogas – iziet un apskatīt profilu

Kad tiek apmeklēts profila skats, ir iespējams nomainīt savu personīgo informāciju – lietotājvārdu, pilno vārdu, e-pastu un profila bildi (skatīt 64. attēlu).



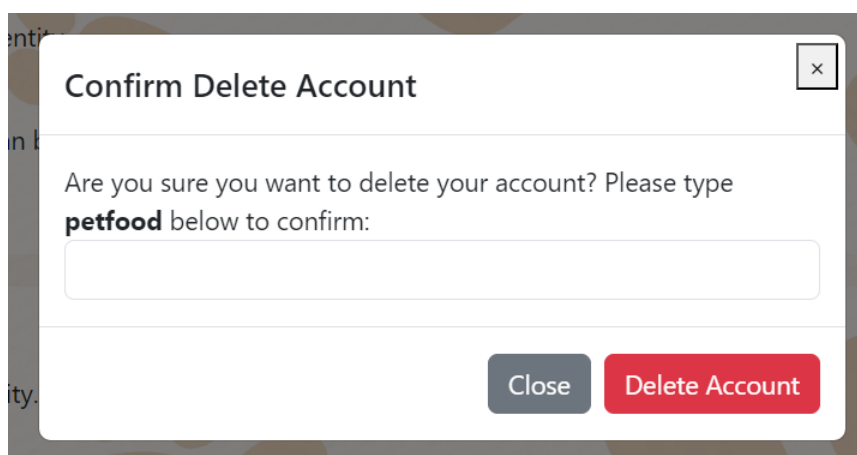
64. attēls. Profila rediģēšanas skats

Ritinot uz leju, var ieraudzīt profila dzēšanas pogu, ja tiešām lietotājs vēlas izdzēst savu profilu. Šī darbība ir neatgriezeniska (skatīt 65. attēlu).



65. attēls. Profila dzēšanas poga

Kad vēlas izdzēst profilu, tiek atvērts modālais logs un prasīts ievadīt apstiprinājuma frāzi, lai netīšām netiktu izdzēsts profils (skatīt 66. attēlu).



66. attēls Profila dzēšanas apstiprinājuma modālais logs

### 8.3 ADMINISTRATORA SKATI

Kad ielogojas lietotājs, kuram ir piešķirta administratora loma, viņš tiek pārvirzīts uz administratora skata pirmo iespējamo veicamo darbību, kas ir pievienot dzīvnieka sugu (skatīt 67. attēlu). Šeit ir iespējams pievienot dažādu skaitu ar attēliem vienai sugai. Ja uzspiež uz vienas no pievienotajām bildēm, tad to var iestatīt kā priekšstata bildi, kas būs redzama ēdienu meklēšanas skatā šai dzīvnieka sugai. Poga “Autofill” iegūst datus no GBIF API datubāzes un daļēji aizpilda ievadformu, ja ir jau atrasts un zināma sugas atslēga. Poga “Submit” ievietos datus, ja tie atbilst validācijas noteiktajiem kritērijiem. Aizpildāmie lauki: GBIF ID; sugas nosaukums vienskaitlī un daudzskaitlī; maršruta parametrs “slug”; valoda; dzīvnieka kategorija; sugu taksonomijas pakāpe; izskats; vispārīgs ēdienu saraksts; bildes un to nosaukumi. Bildēm atļautie paplašinājumi ir JPEG, JPG un PNG.

The screenshot shows the 'ADD SPECIES' form in the Pet Food Admin interface. The form is divided into several sections: 'GBIF ID' with an 'Autofill' button, 'Parent ID', 'Name' (with 'Single' and 'Plural' options), 'Slug', 'Localization' (with 'English' selected), 'Category' (with 'wild' selected), 'Rank' (with 'SUBSPECIES' selected), 'Appearance', 'Food', and 'Images'. The 'Images' section has a note: 'Allowed: jpeg, png, jpg, and max 2 MB' and a 'Choose Files' button. To the right of the form is a 'Preview' section with the instruction: 'Click on any image to set it as the cover image.' It displays three images of cows, each with a 'Remove' button below it. Red arrows point from the 'Autofill' button, the 'Images' section, and the 'Submit' button to the preview images. A red arrow also points to the 'Submit' button.

67. attēls. Administratora skata sugas ievadforma

Administratora skata veicamā darbība - pievienot ēdiena nosaukumu (skatīt 68. attēlu). Poga “Submit” ievietos datus, ja tie atbilst validācijas noteiktajiem kritērijiem. Aizpildāmie lauki: valoda; ēdiena nosaukums; apraksts.

68. attēls. Administratora skata ēdienu nosaukumu ievadforma

Administratora skata veicamā darbība - pievienot ēdiena drošības kategorijas (skatīt 69. attēlu). Poga “Submit” ievietos datus, ja tie atbilst validācijas noteiktajiem kritērijiem. Aizpildāmie lauki: sugas nosaukums; ēdiena nosaukums; drošības kategorija.

69 attēls. Administratora skata sugu un ēdienu drošības kategoriju sasaistes ievadforma

## SECINĀJUMI UN PRIEKŠLIKUMI

1. Darba mērķis tika sasniegts – izstrādāt intuitīvu, ērti un viegli saprotamu tīmekļa vietni, kur jebkurš var ievadīt sava mājdzīvnieka sugu un uzmeklēt informāciju par pārtikas produktu drošu patēriņu attiecīgās dzīvnieka sugas pārstāvim.
2. Tīmekļa vietnei tika izveidots pievilcīgs dizains un reaģējošs saturs.
3. Priekšgalsistēma tika veiksmīgi izmantota ar visiem ievadlaukiem un pogām – ienākšana, reģistrācija, sugu meklēšana, profila rediģēšana un dzēšana, mājdzīvnieku pievienošana un dzēšana. Administratora skatā – sugu, ēdienu un drošības kategoriju pievienošana.
4. Veiksmīgi tika izveidota ER diagramma un datu bāzes struktūra, izmantojot *Laravel* migrācijas, kā arī dinamiski savienota ar aizmugursistēmu un priekšgalsistēmu.
5. Aizmugursistēma tika veiksmīgi izveidota, izmantojot MVC modeli pārskatāmam kodam (sk. 3. pielikumu).
6. Veiksmīgi tika testēta vietnes darbība, pārbaudot lielāko daļu funkcionalitātes un izmēģinot visas ievades kombinācijas – tukši ievadlauki, pārāk gari teksti, nedroša parole, neatrasti dati vai maršruti.
7. Katra problēma tika atrisināta ar korektu un precīzu pieeju, meklējot problēmas cēloni un izcelsmi.

Visas ieplānotās funkcionalitātes netika izveidotas, bet ar laiku var uzlabot programmu.

### Ieteikumi nākotnei:

1. Pievienot ēdienu sarakstam papildus filtrēšanas opcijas, nevis tikai atlasīt visus “drošos” datus.
2. Pievienot papildus valodas (šo var izdarīt jebkurā brīdī, izveidojot jaunus tulkojumus failu sistēmā). Parādītās sugas un ēdieni ir atkarīgi no tā vai konkrētajā valodā ir pieejami dati, vai ir pieejama tikai daļa no datiem. Šobrīd katrai sugai un ēdienam jābūt attiecīgi ierakstam visās vietnē pieejamajās valodās, savādāk tiks parādīta tikai daļa no visiem datiem. *Laravel* ir tā, ja vietnes sadaļas tulkojums nav ierakstīts failu sistēmā tad tiek izmantots angliskais tulkojums pēc noklusējuma (angļu v. *fallback*). Tas nozīmē ka obligāti ir jābūt visiem angliskajiem tulkojumiem, savādāk tiks parādīta tulkojuma masīva atslēga.
3. Turpinot par tulkojumiem, būtu vēlams uztaisīt atsevišķu sadaļu administratora skatā, kur var mainīt un pievienot tulkojumus, kā arī jaunas valodas un publicēt tās failu sistēmā, kad tulkojumi ir pabeigti.
4. Pievienot papildus validāciju visām ievadformām gan priekšgalsistēmā, gan aizmugursistēmā, lai novērstu neparedzētus gadījumus.

5. Ideja izveidot atsevišķu emuāru sadaļu vietnē, kur cilvēkiem būtu iespēja pievienot bildes saistībā ar dzīvniekiem, un komentēt zem ierakstiem.
6. Pievienot papildus lietotāja lomas, piemēram, “Satura veidotājs”, kuram atļauts pievienot dzīvnieku sugu, ēdienu un drošības kategoriju ierakstus, kurus pēc tam pārbauda un publicē administrators. Loma “Mājdzīvnieka īpašnieks”, kas pierāda, ka tev pieder mājdzīvnieks. Loma “Eksperts”, kas pierāda ka cilvēks specializējas vienā no dzīvnieku saistītām profesijām. Lai iegūtu šo lomu, būtu jāapstiprina sava identitāte vietnē.
7. Pievienot CAPTCHA ievades rūtiņu (angļu v. *checkbox*) no *Google* piedāvātā reCAPTCHA servisa tām ievades formām, kur vajag pārliecināties ka darbību veic reāls cilvēks nevis ļaunprogrammatūra (angļu v. *malware*). Piemēram, reģistrācijas ievadformai.
8. Izveidot opciju, ka lietotājs paliek ielogojies, bet nevar veikt atsevišķas darbības pēc noteikta laika perioda. Piemēram, savs mājdzīvnieku saraksts nav īpaši sensitīva informācija, tāpēc apmeklējot saiti, uzreiz var parādīt “/home” skatu, šādi ietaupot nedaudz laika, kas būtu iztērēts ielogojoties tikai dēļ vienas lietas. Šo var panākt izmantojot JWT (JSON web token), kas ir īpašs žetons, kurš tiek glabāts datubāzē un pārbaudīts ar katru HTTP pieprasījumu uz serveri, vai lietotājs ir autentificējies veikt konkrētas darbības (LeanM, 2024).
9. Tumšais motīvs. Tā ir standarta funkcionalitāte modernās mājaslapās, kas uzlabo pieejamību.

## IZMANTOTĀ LITERATŪRA

1. *Bootstrap Documentation* [tiešsaiste]. The Bootstrap team, 2024 [skatīts 11.02.2024].  
Pieejams: <https://getbootstrap.com/docs/5.0/getting-started/introduction/>
2. *Brook Salamander* [tiešsaiste]. Wikipedia, 21.11.2023 [skatīts 16.1.2024]. Pieejams:  
[https://en.wikipedia.org/wiki/Brook\\_salamander](https://en.wikipedia.org/wiki/Brook_salamander)
3. *DataTables* [tiešsaiste]. SpryMedia Ltd., 2023 [skatīts 11.02.2024]. Pieejams:  
<https://datatables.net/extensions/rowreorder/examples/initialisation/responsive.html>
4. *Fontawesome Documentation* [tiešsaiste]. Fonticons, Inc., 2024 [skatīts 11.02.2024].  
Pieejams: <https://docs.fontawesome.com/web/setup/get-started>
5. *GBIF Technical Documentation* [tiešsaiste]. Global Biodiversity Information Facility, 2024  
[skatīts 11.02.2024]. Pieejams: <https://techdocs.gbif.org/en/openapi/v1/species>
6. *How CAPTCHAs work?* [tiešsaiste]. Cloudflare, Inc., 2024 [skatīts 11.02.2024]. Pieejams:  
<https://www.cloudflare.com/learning/bots/how-captchas-work/>
7. *jQuery* [tiešsaiste]. OpenJS Foundation, 2024 [skatīts 11.02.2024]. Pieejams:  
<https://jquery.com/>
8. LeanM, *Laravel REST API with JWT Authentication for Login and Role-based permissions*  
[tiešsaiste], 08.02.2023 [skatīts 09.01.2024]. Pieejams: <https://dev.to/leanm/laravel-rest-api-with-jwt-authentication-for-login-and-role-based-permissions-2p56>
9. *List of ISO 3166 country codes* [tiešsaiste]. Wikipedia, 20.03.2024 [skatīts 11.02.2024].  
Pieejams: [https://en.wikipedia.org/wiki/List\\_of\\_ISO\\_3166\\_country\\_codes](https://en.wikipedia.org/wiki/List_of_ISO_3166_country_codes)
10. McLeod, Lianne. *What Should I Feed My Red-Eared Slider Turtle?* [tiešsaiste], 07.12.2021  
[skatīts 09.01.2024]. Pieejams: <https://www.thesprucepets.com/what-should-i-feed-my-red-eared-slider-1238363>
11. Nils Adermann, Jordi Boggiano. *Composer Documentation* [tiešsaiste], 2024 [skatīts  
9.1.2024]. Pieejams: <https://getcomposer.org/doc/00-intro.md>
12. *OAuth 2.0 for Client-side Web Applications* [tiešsaiste]. Google Developers, 07.11.2023  
[skatīts 09.01.2024]. Pieejams:  
<https://developers.google.com/identity/protocols/oauth2/javascript-implicit-flow>
13. *PHPDoc Documentation* [tiešsaiste]. 2mdc, 2024 [skatīts 11.02.2024]. Pieejams:  
<https://docs.phpdoc.org/guide/references/phpdoc/basic-syntax.html>
14. Sampad Sarker, *5 Quick Steps Transpiling SCSS to CSS Using Node* [tiešsaiste], 20.09.2022  
[skatīts 11.02.2024]. Pieejams: <https://dev.to/sampadsarker/5-quick-steps-transpiling-scss-to-css-using-node-with-watch-minify-mode-2mm>

15. Sebastian Bergmann, *PHPUnit Documentation* [tiešsaiste], 2024 [skatīts 09.01.2024].  
Pieejams: <https://getcomposer.org/doc/00-intro.md>
16. Taylor Otwell, *Laravel* [tiešsaiste]. Laravel LLC, 07.12.2021 [skatīts 11.02.2024]. Pieejams:  
<https://laravel.com/>
17. *What is SSL?* [tiešsaiste]. Cloudflare, Inc., 2024 [skatīts 11.02.2024]. Pieejams:  
<https://www.cloudflare.com/learning/ssl/what-is-ssl/>



## **PIELIKUMI**

## 1. pielikums. Meklēšanas rezultātu aizmugursistēmas API metodes loģika (PHP)

```
1. public function autocomplete(Request $request)
2. {
3.     $searchQuery = $request->input('q');
4.     $locale = $request->input('locale', app()->getLocale());
5.
6.     $results = \DB::table('animals')
7.         ->where('language', $locale)
8.         ->where(function ($query) use ($searchQuery) {
9.             $query->where('name', 'like', "%$searchQuery%")
10.             ->orWhere('slug', 'like', "%$searchQuery%");
11.         })
12.         ->get();
13.
14.     $data = [];
15.     foreach ($results as $result) {
16.         $name = explode('|', $result->name)[0];
17.         $data[] = [
18.             'gbif_id' => $result->gbif_id,
19.             'rank' => $result->rank,
20.             'name' => $name,
21.             'category' => $result->category,
22.         ];
23.     }
24.
25.     return response()->json([
26.         'data' => $data,
27.         'locale' => $locale,
28.     ]);
29. }
```

Lai lietotājam varētu parādīt atbilstošus rezultātus, tiek atlasīti dati, kas atbilst ierakstītajai meklēšanas frāzei un izvēlētajai vietnes valodai.

## 2. pielikums. Meklēšanas rezultātu priekšgalsistēmas loģika (JavaScript)

```
1. function load_data(query) {
2.     if (query.length > 2) {
3.         // Parādīt lādēšanās animāciju kamēr dati tiek iegūti
4.         document.getElementById('search-loader').style.display = 'block';
5.
6.         fetch("/api/autocomplete?q=" + query + "&locale=" +
7. document.documentElement.lang, {
8.             method: "GET",
9.             headers: {
10.                 'Content-type': 'application/json; charset=UTF-8'
11.             }
12.         }).then(function(response) {
13.             if (!response.ok) {
14.                 throw new Error('Network response was not ok');
15.             }
16.             return response.json();
17.         }).then(function(response) {
18.             // Paslēpt lādēšanās animāciju kad dati ir iegūti
19.             document.getElementById('search-loader').style.display = 'none';
20.
21.             var data = response.data;
22.
23.             var html = '<div class="list-group">';
24.             if (data.length > 0) {
25.                 for (var count = 0; count < data.length; count++) {
26.                     /* Viena rezultāta HTML struktūra */
27.
28.                 } else {
29.                     html += '<a href="#" class="list-group-item list-group-item-
30. action disabled">' + t.no_data_found + '</a>';
31.                 }
32.                 html += '</div>';
33.                 document.getElementById('search_result').innerHTML = html;
34.             }).catch(function(error) {
35.                 // Paslēpt lādēšanās animāciju un parādīt kļūdas ziņojumu
36.                 document.getElementById('search-loader').style.display = 'none';
37.                 document.getElementById('search_result').innerHTML = '<p>' +
38. t.something_went_wrong + '</p>';
39.                 console.error('Error:', error);
40.             });
41.         } else {
42.             document.getElementById('search_result').innerHTML = '';
43.         }
44.     }
```

Priekšgalsistēmā bija noderīgi izmantot *Javascript* kodu, jo ar to var tieši ietekmēt HTML struktūru kā izveidot elementus, pievienot klases un saturu. API “Fetch” darbībā tiek iekļauta gan

meklēšanas frāze, gan 2 ciparu valodas kods. Kods var arī nebūt norādīts, jo pēc noklusējuma ir izveidots lai datus atgriež angļu valodā. Lai vizualizētu, ka dati tiek iegūti no datu bāzes, tiek parādīta lādēšanas animācija. Ja nav datu, tad tiek atgriezts kļūdas ziņojums attiecīgajā vietnes valodā. Darbību secība “fetch”, “then” un visbeidzot “catch” ir standarta pieceja API vaicājuma izsaukšanai.

Svarīgi pieminēt ka implementēju API pieprasījumu limitēšanas mehānismu, ja lietotājs raksta atslēgvārdu meklētājā, pēc pēdējā taustiņa nospiešanas, tiek gaidīta viena pilna sekunde pirms tiek nosūtīts pieprasījums no priekšgalsistēmas uz aizmugursistēmu. Kā arī ja lietotājs uzspiež “Enter” taustiņu uz klaviatūras, pieprasījums tiek izdarīts bez aizkavēšanās, toties ja tiek atklāts, ka lietotājs ir spiedis šo taustiņu 3 spiedienus sekundē, tad uz sekundi parādās brīdinājums un laika atskaite.

### 3. pielikums. MVC programmēšanas pieejas piemērs

Daļa no “admin\foods.blade.php” skata struktūras. “old()” *Laravel* funkcija ir noderīga, lai saglabātu ievadformā iepriekš ievadītās vērtības, ja tās bija nepilnīgas.

```
1. @if (session('success'))
2. <div class="alert alert-success mx-4">{{ session('success') }}</div>
3. @endif
4. @if (session('error'))
5. <div class="alert alert-danger mx-4">{{ session('error') }}</div>
6. @endif
7. {{-- Citi ievadlauki --}}
8. <div class="form-group required">
9.     <label for="language" class="mb-1">Localization</label>
10.    <select class="form-control" id="language" name="language">
11.        @foreach($supportedLanguages as $code => $language)
12.            <option value="{{ $code }}" {{ old('language') == $code ? 'selected'
13.            : '' }}>{{ $language['name'] }}</option>
14.        @endforeach
15.    </select>
16.    @error('language')
17.        <span class="text-danger fs-5">{{ $message }}</span>
18.    @enderror
19. </div>
```

Maršruti iekš “web.php” faila. Var izveidot arī citus HTTP metožu maršrutus, nevis tikai GET un POST, bet šie ir visbiežāk lietotie. Nosaukumi atbilst metodēm un maršruta nosaukumu var izdomāt kādu vēlas.

```
1. Route::get('/foods', [
2.     AdminController::class, 'showFoods'
3. ])->name('admin.foods.index');
4. Route::post('/foods', [
5.     AdminController::class, 'createFoods'
6. ])->name('admin.foods.create');
```

Daļa no “AdminController” kontroliera. Kontrolierī parasti notiek vispārīga fatu apstrāde un tie tiek padoti uz vienu no “.blade.php” paplašinājuma skatiem. Sīkāka datu apstrāde notiek Modeļos, ko izmanto iekš kontroliera, piemēram “Animal” modelis.

```
1. <?php
2.
3. namespace App\Http\Controllers;
4.
5. use Illuminate\Http\Request;
6. use App\Models\Animal;
7.
8. class AdminController extends Controller
9. {
10.     public function showFoods()
11.     {
12.         $supportedLanguages = config('languages');
13.         return view('admin.foods', compact('supportedLanguages'));
14.     }
15.     public function createFoods(Request $request)
16.     {
17.         $request->validate([
18.             'language' => 'required|string|size:2',
19.             'food' => 'required|string',
20.             'description' => 'nullable|string',
21.         ]);
22.         try {
23.             $maxFoodId = \DB::table('foods')
24.                 ->where('language', $request->language)
25.                 ->max('food_id');
26.             $foodId = $maxFoodId + 1;
27.
28.             \DB::table('foods')->insert([
29.                 'language' => $request->language,
30.                 'food_id' => $foodId,
31.                 'food' => $request->food,
32.                 'description' => $request->description,
33.             ]);
34.
35.             return back()->with('success', 'Food record created successfully!');
36.         } catch (\Exception $e) {
37.             return back()->with('error', 'Error creating food record: ' . $e-
38.                 >getMessage());
39.         }
```



Logotips tika izveidots izmantot tiešsaistes rīku <https://looka.com>, kas ir tam ir paredzēta. Tas ir ievietots vietnē SVG paplašinājuma formātā, kas paātrina gan ielādi, gan kvalitāti, jo attēls ir radīts pamatojoties uz vektorgrafiku. Izgaistošais tonis no labās augšējās puses tika panākts izmantojot CSS. Logotips tika izmantots navigācijā, 404. koda kļūdas ziņouma skatā un sugu hierarhijas rādītājā iekš ēdienu meklēšanas skata. Logotipam bija viegli nomainīt krāsu iekš SVG atribūtiem tiešā veidā caur CSS.