Liepājas Valsts tehnikums

**Uzmeklēšanas tīmekļa vietne**

Kvalifikācijas eksāmena praktiskās daļas dokumentācija

Profesionālā kvalifikācija

Grupas nosaukums

Projekta izstrādātājs

/vārds, uzvārds, paraksts/

Eksāmena datums 2023. gada

Liepāja 2023

# Saturs

[Saturs 4](#_Toc137449218)

[Ievads 3](#_Toc137449219)

[1. Uzdevuma formulējums 4](#_Toc137449220)

[2. Programmatūras prasību specifikācija 5](#_Toc137449221)

[2.1. Produkta perspektīva 5](#_Toc137449222)

[2.2. Sistēmas funkcionālās prasības 5](#_Toc137449223)

[2.2.1. IP ASN uzmeklēšana 5](#_Toc137449224)

[2.2.2. Domēnu uzmeklēšana 6](#_Toc137449225)

[2.2.3. Jaucējvērtību uzmeklēšana 7](#_Toc137449226)

[2.2.4. IP drošību uzmeklēšana 8](#_Toc137449227)

[2.2.5. Iesniegšanas poga 9](#_Toc137449228)

[2.3. Sistēmas nefunkcionālās prasības 10](#_Toc137449229)

[2.4. Gala lietotāja raksturiezīmes 11](#_Toc137449230)

[3. Izstrādes lidzekļu, rīku apraksts un izvēles pamatojums 12](#_Toc137449231)

[3.1. Iespējamo risinājuma līdzekļu un valodu apraksts 12](#_Toc137449232)

[3.2. Izvēlēto risinājumu līdzekļu un valodu apraksts 13](#_Toc137449233)

[4. Sistēmas modelēšana un projektēšana 14](#_Toc137449234)

[4.1. Sistēmas struktūras modelis 14](#_Toc137449235)

[4.2. Klašu diagramma 15](#_Toc137449236)

[4.3. Funkcionālais un dinamiskais sistēmas modelis 16](#_Toc137449237)

[4.4. Aktivitāšu diagramma 17](#_Toc137449238)

[4.5. Lietojumgadījumu diagramma 18](#_Toc137449239)

[4.6. Sistēmas moduļu apraksts un algoritmu shēmas 19](#_Toc137449240)

[5. Lietotāju ceļvedis 20](#_Toc137449241)

[6. Testēšanas dokumentācija 27](#_Toc137449242)

[6.1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums 27](#_Toc137449243)

[6.2. Testpiemēru kopa 27](#_Toc137449244)

[6.3. Testēšanas žurnāls 29](#_Toc137449245)

[7. Secinājumi 30](#_Toc137449246)

[8. Lietoto saīsinājumu saraksts 31](#_Toc137449247)

[9. Literatūras un informācijas avotu saraksts 32](#_Toc137449248)

[Pielikums 33](#_Toc137449249)

# Ievads

Projekts ir izstrādāts ar mērķi, lai lietotājs varētu ātri un vienkārši kurā nav pārāk daudz informācijas, uzmeklēt no piedāvātajām uzmeklēšanas iespējām. Mājas lapa ir bāzēta uz Python programmēšanas valodas. Mājaslapu varētu izmantot jebkurš lietotājs, gan kiberdrošības speciālists, gan tīklāja drošības speciālists, gan lietotājs kuram ir interese šādā jomā.

Lietotājs var apskatīt savas ievadītās ievades rezultātu tabulā, bez liekas navigēšanas.

Pirmā nodaļa saturēs par to, kā es viekšu uzdevumu, lai sasniegtu savu mērķi projektam. Kādus rīkus izmantošu, kādas valodas un programmatūras.

Otrā nodaļa saturēs funkcijas ko mans projekts veic detalizētā aprakstā, kā arī mazs ieskats kas lietotājam ir tāds jāzina. Arī kādēļ produkts tika veidots un kurš šo produktu izmantos.

Trešajā nodaļā būs neliels apraksts par to ko es varēju izmantot savā projektā. Taču būs arī apraksts par tām valodām un rīkiem ko es izmantoju šajā projektā.

Ceturtajā nodaļā saturēs diagrammas par projektu. Katrā diagrammā būs mazs apraksts par diagrammu kurā pēc tam būs attēlota diagramma.

Piektā nodaļa saturēs par to kā lietotājs var izmantot projektu. Kā arī būs apraksts par to, ko darīt ja lietotājs ir ievadījis kļūdu ievades laukā.

Astostā nodaļa būs par projektu, kā arī kur bija viegli, kādas grūtības bija, ko es iemācījos.

# Uzdevuma formulējums

CERT man uzticēja ar uzņēmuma sasaldēto projektu, prakses vadītājs prakses laikā teica ko izmantot un ko mainīt tā, lai CERT uzņēmums būtu apmierināts ar projektu izstrādi.

Lai sasniegtu funkcionalitāti un lai lietotājs varētu uzmeklēt ko vēlas, mans projekts izmantos vairākas programmatūras. Taču, lai mans projekts realizētos man vajadzēja izmantot HTML priekš ārējas saskarnes, Python un Flask priekš aizmugursistēmas, lai varētu izvadītu rezultātu. Lai nodrošinātu tīmekļa lietojumprogrammu stabilitāti un mērogojamību, izmantos Nginx un Docker.

Galvenā funkcionalitāte ir ietvērta vienā HTML lapā. Lietotājs var ievadīt vienu vai vairākas ievades, piemēram, IP adresi, domēnu vai jaucējvērtību ievades laukā, iesniedzot, aizmugursistēma apstrādās datus. Pēc datu apstrādes, informācija tiks izvadīta tabulas formātā.

Aizmugursistēmai ir iekļauts kļūdu apstrādes pasākums, lai novērstu problēmas kas var rasties nederīgas ievades dēļ. Gadījumā lietotājs iesniedz tukšu ievadi, lietotājam parādīsies teksts kas paziņos, ka ievades lauks ir tukšs. Gadījumā informāciju par konkrēto ievadi neeksistē, tad tas tiks parādīts lietotājam tabulā.

# Programmatūras prasību specifikācija

## Produkta perspektīva

Produkts ir uzmeklēšanas tīmekļa vietne kas uzmeklē to ko lietotājs ir ievadījis, atšķirībā ko lietotājs ir ievadījis, informācija par ievadīto tiks izvadīts. Tīmekļa lietojumprogramma piedāvā ātru un nesarežģītu viedu lietotājiem, lai varētu uzmeklēt bez reģistrācijas. Produkts ir paredzēts tīkla administratoriem, drošības speciālistiem vai arī kiberdrošības speciālistiem.

## Sistēmas funkcionālās prasības

Lai izveidotu projektu, man vajadzēja pielietot Python un Flask, priekš aizmugursistēmas apstrādes. Priekš ārējās saskarnes vajadzēja pielietot HTML un Jinja. Lai varētu apkalpot vairākus lietotājus, projektam vajadzēja izmantot Nginx kā starpniekserveris. Lai projekts varētu pielietot vairākas programmatūras vienlaicīgi, vajadzēja izmantot Docker, kurā programmatūras bija Nginx un MySQL.

### IP ASN uzmeklēšana

Mērķis:

Funkcija ir ļaut lietotājiem ievadīt vienu vai vairākus IPv4 adreses sistēmā, tā lai sistēma var apskatīt un analizēt ievades datus.

Ievades dati:

Lietotājs ievada vienu vai vairākus IPv4 adreses vienā ievades laukā tīmekļa lietojumprogrammā. Ievades lauks ir apzīmēts kā “IP adrese” un apstiprinās IPv4 adreses kuras ir atdalītas ar komatu.

Apstrāde:

Aizmugursistēma veiks vairākas lietotāja ievades pārbaudes, lai pārliecinātos, ka tā ir derīga. Sākumā sistēma pārbaudīs, vai ievades laukā ir dati. Ja nav dati, tad sistēma nosūtīs ziņojumu atpakaļ uz ārējo saskarni un likt lietotājam ievadīt vismaz vienu IP adresi. Ja ievadē nav derīgu IPv4 adrešu, tad ārējā saskarnē zem ievades lauka būs sarkans paziņojums par nepareizu ievadi.

Ja ievades laukā satur vairākas derīgas IP adreses, tad sistēma atdalīs to vienu no otra, lai katrs būtu savs elements, noņemot komatu un visas esošās atstarpes. Sistēma lasīs katru elementu individuāli, lai vienkāršāks process datu meklēšanā. Process tiks atkārtots līdz ko vairs nav elementi. Visi apkopotie dati tiek nosūtīti uz ārējo saskarni, kurā notiek pēdējā secību kārtošana.

Izvades dati:

1. Ievade satur vienu vai vairāku derīgu IPv4 adresi un ir vieksmīgi apstrādāts, sistēma parādīs iegūtos datus tabulas formātā sistēmas tīmekļa lapā. Tabula būs ietvertas IP adreses, ASN nosaukums un valsts, katrs elements tiks parādīts attiecīgajās kolonnās;
2. Gadījumā ievade nesatur komatu starp divām IP adresēm, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
3. Gadījumā ievade nesatur derīgu IPv4 adresi, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
4. Ja viena vai vairākas ievades laukā ir IP adrese kam nav informācijas, tad uz atbilstošās rindas būs iekrāsots ar dzeltenu krāsu, kā arī ISP kolonnā būs ar šādu tekstu “Nav informācijas”;
5. Gadījumā lietotājs iesniedz tukšu ievades lauku, sistēma parādīs virs ievades lauka ar sarkanu tekstu “Error: tukšs lauks, ievadiet IPv4 adresi;
6. Ja IP adrešu apstrādes laikā rodas iekšēja servera kļūda, sistēma parādīs kļūdas ziņojumu tīmekļa lietojumprogrammas lapā, norādot ka izgūstot informāciju radās problēma un lietotājam vēlāk jāmēģina vēlreiz.

### Domēnu uzmeklēšana

Mērķis:

Funkcija ir ļaut lietotājiem ievadīt vienu vai vairākus domēnu adreses sistēmā, tā lai sistēma var apskatīt un analizēt ievades datus.

Ievades dati:

Lietotājs ievada vienu vai vairākus domēnu adreses vienā ievades laukā tīmekļa lietojumprogrammā. Ievades lauks ir apzīmēts kā “domēnu adrese” un apstiprinās domēnu adreses kuras ir atdalītas ar komatu.

Apstrāde:

Aizmugursistēma veiks vairākas lietotāja ievades pārbaudes, lai pārliecinātos, ka tā ir derīga. Sākumā sistēma pārbaudīs, vai ievades laukā ir dati. Ja nav dati, tad sistēma nosūtīs ziņojumu atpakaļ uz ārējo saskarni un likt lietotājam ievadīt vismaz vienu domēnu adresi. Ja ievadē nav derīgu domēņu adrešu, tad ārējā saskarnē zem ievades lauka būs sarkans paziņojums par nepareizu ievadi.

Ja ievades laukā satur vairākas derīgas domēnu adreses, tad sistēma atdalīs to vienu no otra, lai katrs būtu savs elements, noņemot komatu un visas esošās atstarpes. Sistēma lasīs katru elementu individuāli, lai vienkāršāks process datu meklēšanā. Process tiks atkārtots līdz ko vairs nav elementi. Visi apkopotie dati tiek nosūtīti uz ārējo saskarni, kurā notiek pēdējā secību kārtošana.

Izvades dati:

1. Ievade satur vienu vai vairāku derīgu domēnu adresi un ir vieksmīgi apstrādāts, sistēma parādīs iegūtos datus tabulas formātā sistēmas tīmekļa lapā. Tabula būs ietvertas domēnu adreses, IP adrese un ‘VirusTotal’ saite, katrs elements tiks parādīts attiecīgajās kolonnās;
2. Gadījumā ievade nesatur komatu starp divām domēnu adresēm, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
3. Gadījumā ievade nesatur derīgu domēnu adresi, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
4. Ja viena vai vairākas ievades laukā ir domēnu adrese kam nav informācijas, tad uz atbilstošās rindas būs iekrāsots ar dzeltenu krāsu, kā arī IP adrešu kolonnā būs ar šādu tekstu “Nav informācijas”;
5. Gadījumā lietotājs iesniedz tukšu ievades lauku, sistēma parādīs virs ievades lauka ar sarkanu tekstu “Error: tukšs lauks, ievadiet domēnu adresi;
6. Ja domēņu adrešu apstrādes laikā rodas iekšēja servera kļūda, sistēma parādīs kļūdas ziņojumu tīmekļa lietojumprogrammas lapā, norādot ka izgūstot informāciju radās problēma un lietotājam vēlāk jāmēģina vēlreiz.

### Jaucējvērtību uzmeklēšana

Mērķis:

Funkcija ir ļaut lietotājiem ievadīt vienu vai vairākus jaucējvērtības sistēmā, tā lai sistēma var apskatīt un analizēt ievades datus.

Ievades dati:

Lietotājs ievada vienu vai vairākus jaucējvērtības vienā ievades laukā tīmekļa lietojumprogrammā. Ievades lauks ir apzīmēts kā “Jaucējvērtība” un apstiprinās trīs no jaucējvērtībām (MD5, SHA-1, SHA-256) kuras ir atdalītas ar komatu.

Apstrāde:

Aizmugursistēma veiks vairākas lietotāja ievades pārbaudes, lai pārliecinātos, ka tā ir derīga. Sākumā sistēma pārbaudīs, vai ievades laukā ir dati. Ja nav dati, tad sistēma nosūtīs ziņojumu atpakaļ uz ārējo saskarni un likt lietotājam ievadīt vismaz vienu jaucējvērtību. Ja ievadē nav derīgu jaucējvērtību, tad ārējā saskarnē zem ievades lauka būs sarkans paziņojums par nepareizu ievadi.

Ja ievades laukā satur vairākas derīgas jaucējvērtības, tad sistēma atdalīs to vienu no otra, lai katrs būtu savs elements, noņemot komatu un visas esošās atstarpes. Sistēma lasīs katru elementu individuāli, lai vienkāršāks process datu meklēšanā. Process tiks atkārtots līdz ko vairs nav elementi. Visi apkopotie dati tiek nosūtīti uz ārējo saskarni, kurā notiek pēdējā secību kārtošana.

Izvades dati:

1. Ievade satur vienu vai vairāku jaucējvērtību un ir vieksmīgi apstrādāts, sistēma parādīs iegūtos datus tabulas formātā sistēmas tīmekļa lapā. Tabula būs ietvertas jaucējvērtības, jaucējvērtību drošība un ‘VirusTotal’ saite, katrs elements tiks parādīts attiecīgajās kolonnās;
2. Gadījumā ievade nesatur komatu starp divām jaucējvērtībām, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
3. Gadījumā ievade nesatur derīgu jaucējvērtību, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
4. Ja vienā vai vairākas ievades laukā ir jaucējvērtības kam nav informācijas, tad uz atbilstošās rindas būs iekrāsots ar dzeltenu krāsu, kā arī jaucējvērtību drošības kolonnā būs ar šādu tekstu “Nav informācijas”;
5. Gadījumā lietotājs iesniedz tukšu ievades lauku, sistēma parādīs virs ievades lauka ar sarkanu tekstu “Error: tukšs lauks, ievadiet jaucējvērtību;
6. Ja jaucējvērtības apstrādes laikā rodas iekšēja servera kļūda, sistēma parādīs kļūdas ziņojumu tīmekļa lietojumprogrammas lapā, norādot ka izgūstot informāciju radās problēma un lietotājam vēlāk jāmēģina vēlreiz.

### IP drošību uzmeklēšana

Mērķis:

Funkcija ir ļaut lietotājiem ievadīt vienu vai vairākas IP adreses sistēmā, tā lai sistēma var apskatīt un analizēt ievades datus.

Ievades dati:

Lietotājs ievada vienu vai vairākas IPv4 adreses vienā ievades laukā tīmekļa lietojumprogrammā. Ievades lauks ir apzīmēts kā “IP adrese” un apstiprinās IPv4 adreses kuras ir atdalītas ar komatu.

Apstrāde:

Aizmugursistēma veiks vairākas lietotāja ievades pārbaudes, lai pārliecinātos, ka tā ir derīga. Sākumā sistēma pārbaudīs, vai ievades laukā ir dati. Ja nav dati, tad sistēma nosūtīs ziņojumu atpakaļ uz ārējo saskarni un likt lietotājam ievadīt vismaz vienu IP adresi. Ja ievadē nav derīgu IPv4 adrešu, tad ārējā saskarnē zem ievades lauka būs sarkans paziņojums par nepareizu ievadi.

Ja ievades laukā satur vairākas derīgas IP adreses, tad sistēma atdalīs to vienu no otra, lai katrs būtu savs elements, noņemot komatu un visas esošās atstarpes. Sistēma lasīs katru elementu individuāli, lai vienkāršāks process datu meklēšanā. Process tiks atkārtots līdz ko vairs nav elementi. Visi apkopotie dati tiek nosūtīti uz ārējo saskarni, kurā notiek pēdējā secību kārtošana.

Izvades dati:

1. Ievade satur vienu vai vairāku derīgu IPv4 adresi un ir vieksmīgi apstrādāts, sistēma parādīs iegūtos datus tabulas formātā sistēmas tīmekļa lapā. Tabula būs ietvertas IP adreses, IP adrešu drošība un ‘Virustotal’ saite, katrs elements tiks parādīts attiecīgajās kolonnās;
2. Gadījumā ievade nesatur komatu starp divām IP adresēm, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
3. Gadījumā ievade nesatur derīgu IPv4 adresi, zem ievades lauka būs sarkans teksts kas paziņo lietotājam par kādu ievades kļūdu;
4. Ja viena vai vairākas ievades laukā ir IP adrese kam nav informācijas, tad uz atbilstošās rindas būs iekrāsots ar dzeltenu krāsu, kā arī IP adrešu drošības kolonnā būs ar šādu tekstu “Nav informācijas”;
5. Gadījumā lietotājs iesniedz tukšu ievades lauku, sistēma parādīs virs ievades lauka ar sarkanu tekstu “Error: tukšs lauks, ievadiet IPv4 adresi;
6. Ja IP adrešu apstrādes laikā rodas iekšēja servera kļūda, sistēma parādīs kļūdas ziņojumu tīmekļa lietojumprogrammas lapā, norādot ka izgūstot informāciju, radās problēma, un lietotājam vēlāk jāmēģina vēlreiz.

### Iesniegšanas poga

Mērķis:

Iesniegt lietotāja ievadītos datus

Ievaddati:

Nav

Apstrāde

Iesniedz lietotāja ievadītos datus uz aizmugursistēmu, pēc datu iesniegšanas, informācija par ievadīto tiks izvadīts ārā.

Izvades dati:

Izvadīs apstrādātos datus ārējā saskarnē.

## Sistēmas nefunkcionālās prasības

1. Tīmekļa vietnē ir spēja uzmeklēt vairākas ievades vienlaicīgi, bet par katru atbilstošu ievadi ir jābūt atdalītam ar komatu.
2. Tīmekļa vietnei ir spēja pielāgoties vairāku tīklāju pieprasījumam nezaudējot funkcijas ātrumu.
3. Aizsardzībai tiek izmantots Nginx, kā starpniekserveris, tā ir kā barjera priekš tīmekļa vietnes, lai pasargātu tīmekļa vietni no tiešās saskarsmes ar internetu.
4. Tīmekļa vietne ir vienkārša, bez sarežģītas navigācijas, viegli saprotama izvades rezultāts.
5. Tīmekļa vietnē var apskatīt dažādās pārlūkprogrammās.
6. Mājas lapai ir jābūt izstrādātai latviešu valodā.

## Gala lietotāja raksturiezīmes

Projekts ir domāts priekš tīklāju administratoriem, kiberdrošības speciālistiem, izmeklētājiem vai cilvēkiem kuriem ir hobijs ar tiklāja infrastruktūru. Gala lietotājam ir vajadzīgs pieslēgums internetam, kā arī zināt terminus par:

* IPv4 adrese;
* Jaucējvērtības:
  + MD5;
  + SHA-1;
  + SHA-256.
* Domēns.

Lietotājam varētu būt vairāki mērķi. Lietotājam ir iespēja apskatīties vai IP adrese ir reģstrēta, kā arī lietotājam varētu būt mērķis pārbaudīt IP adrešu drošību. Lietotājs varētu arī uzmeklēt domēnu adreses IP reģistru un to drošību izmantojot ‘Virustotal’ servisu. Lietotājam ir iespēja uzmeklēt jaucējvērtību drošības statusu uz ātro, taču lietotājam ir arī iespēja apskatīties detalizētāk par jaucējvērtību izmantojot ‘Virustotal’ servisu.

# Izstrādes lidzekļu, rīku apraksts un izvēles pamatojums

## Iespējamo risinājuma līdzekļu un valodu apraksts

Šajā nodaļā būs aprakstīts par alternatīvajiem rīkiem ko varēja pievienot projektā. Taču šie rīki netika izmantoti, dēļ prasības no prakses vadītāja.

* Django – Ir augsta līmeņa mājaslapu izstrāde Python programmēšanas valodas, taču projektā es nevarēju izmantot, dēļ tā, ka projektam vajadzēja izmantot Flask, pēc prakses vadītāja ieteikuma.
* Buildah – ir komandrindas rīks, kas ir paredzēts konteineru attēlu veidošanai un manipulēšanai ar tiem. Taču prakses laikā bija jāizmanto Docker, jo ar to varēja veidot konteinerus, pārvaldību un orķestrēt vairākas programmatūras vienlaikus.
* PostgreSQL – ir jaudīga atvērtā pirmkoda relāciju datubāzes pārvaldes sistēma, taču projektā nevarēja izmantot divu galveno iemeslu dēļ. Pirmkārt, projektas izstrādes laikā, labāk sapratu MySQL. Otrkārt, MySQL bija tuvu ieviešanai manā projektā, tāpēc bija prakstiskāk un efektīvāk turpināt ar MySQL.

## Izvēlēto risinājumu līdzekļu un valodu apraksts

Projektā izmantoju:

* Produkts izmanto Docker, lai varētu palaist vairākas programmatūras kopā ar citām programmatūrām, kā viena aplikācija vai serviss. Docker projektā tiek izmantots, lai uzbūvētu pašu uztaisītu projektu un kopā tiek palaists ar MySQL un Nginx.
* Produkts tiek izmantots programmēšanas valoda Python, ar kuru palīdzību var uzmeklēt un apstrādāt lietotāja ievadītos datus. Lai ir vienkāršāka mājaslapu izveidošana, Flask ir pievienota projektu izstrādei kopā ar Python.
* Produkts izmanto MySQL, lai dati tiek ievietoti tabulā pēc datu apstrādes. MySQL projektā strādā pēc CRUD principa (Create, Read, Update and Delete). Tabulas tiek izveidotas, jauni dati tiek izlasīti un atjaunoti pēc katras ievades un duplikāti tiek dzēsti automātiski.
* Produkts izmanto HTML un Jinja priekš ārējās saskarnes. HTML tiek izmantots, lai varētu izveidot ievades lauku, kā arī tabulas priekš izvades, kopā ar HTML arī tiek izmantots bootstrap, lai būtu vieglāk skārenjūtīgiem mobiliem tārluņiem ar pārlūkprogrammu viegli navigēt tīmekļa vietnē
* Jinja ir ieteicams izmantot kopā, ja programmētājs vēlās izveidot mājaslapu ar Python programmēšanas valodu, tamdēļ ka ar Jinja var vieglāk sūtīt un saņemt datus no ārējās saskarnes uz aizmugursistēmu.

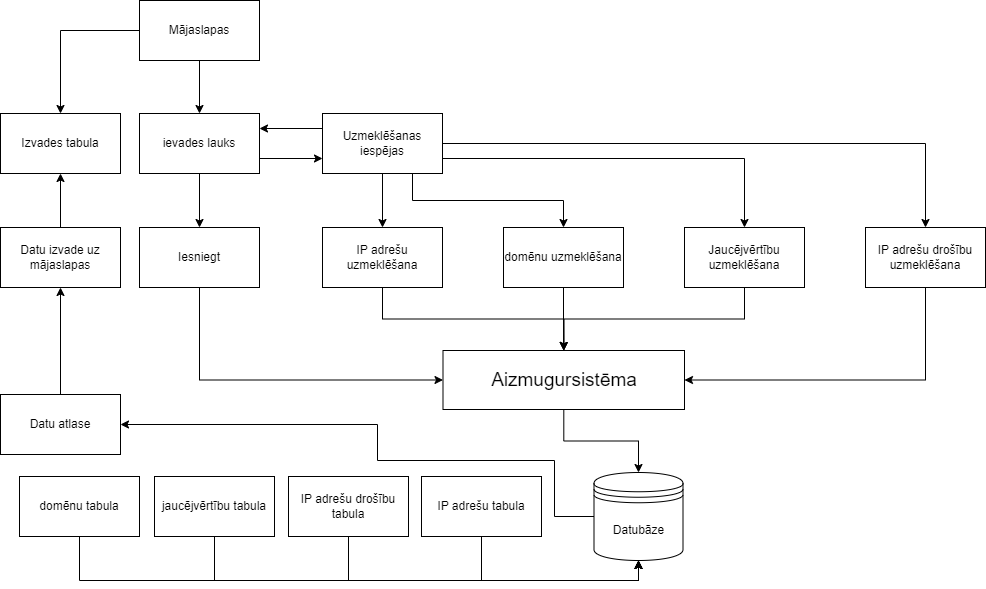
# Sistēmas modelēšana un projektēšana

## Sistēmas struktūras modelis

Šajā modelī ir parādīts kā projekts strādā un ar ko ir saistīts (Skatīt 1. attēlu).

* + - 1. **Attēls**

**Sistēmas struktūras modelis**

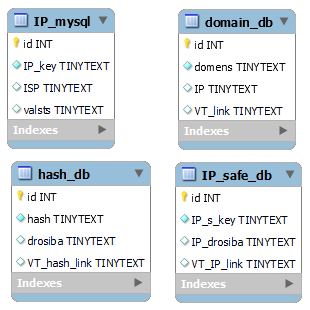


## Klašu diagramma

Datubāze satur četras tabulas kurā katra tiek aizpildīta ar lietotāja ievadītiem, bet apstrādātiem datiem. Katrā ir ‘TINYTEXT’ kā datu tips, tamdēļ lai projektā nebūtu MySQL kļūdas par tukšu vai nederīgu ievadi. Tabulas nav savienotas vienas ar otru (Skatīt 2. attēlu).

* + - 1. **Attēls**

**Klašu diagramma**

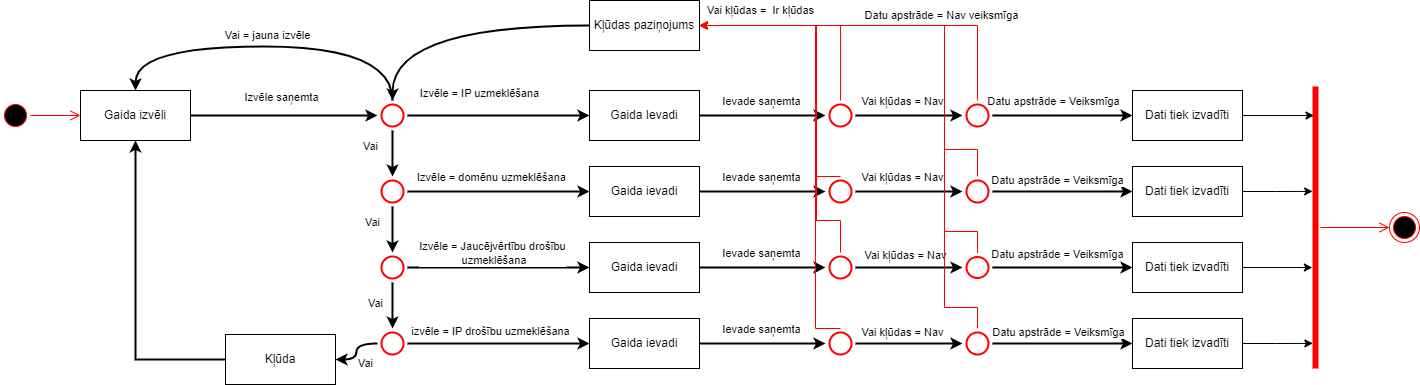


## Funkcionālais un dinamiskais sistēmas modelis

Diagramma ir stāvokļu diagramma. Pēc katra stāvokļa ir kāda darbība, piemēram lietotājs izvēlās vienu četrām opcijām, kurā pēc tam sistēma gaida ievadi. Ir veidots tā, lai parādītu lietotājam kā sistēma darbojas pēc lietotāja ievades (Skatīt 3. attēlu).

* + - 1. **Attēls**

**Stāvokļa diagramma**

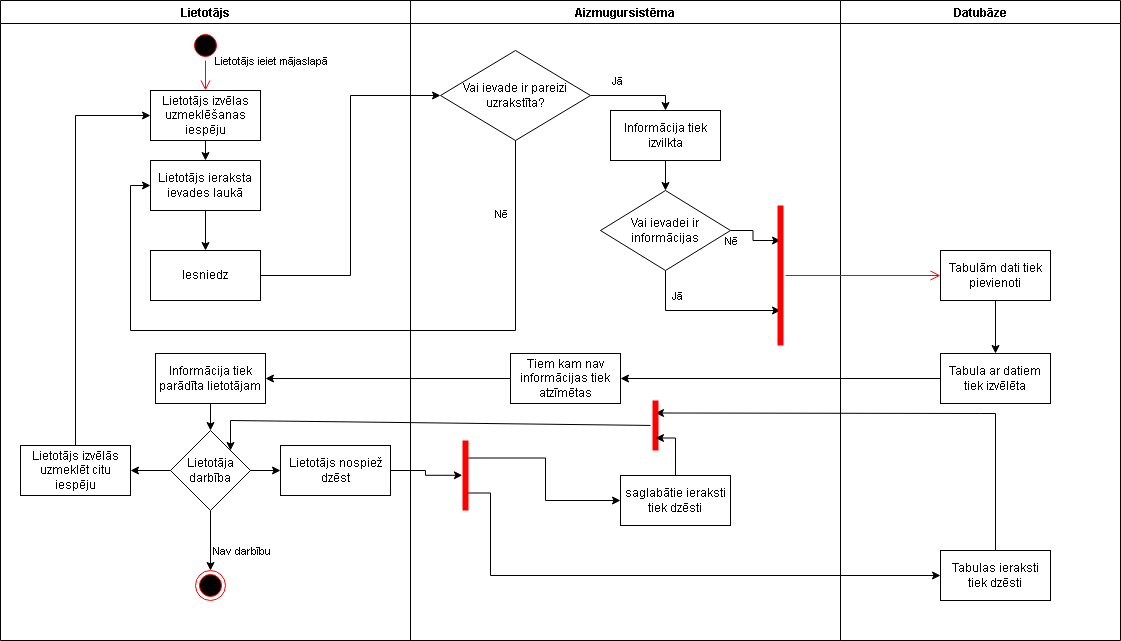


## Aktivitāšu diagramma

Šī aktivitāšu diagramma parāda, ko lietotājs var darīt ar sistēmu. Parāda to ko aizmugursistēma apstrādā, kā arī datubāzes apstrādi (Skatīt 4. attēlu).

**4. Attēls**

**Aktivitāšu diagramma**

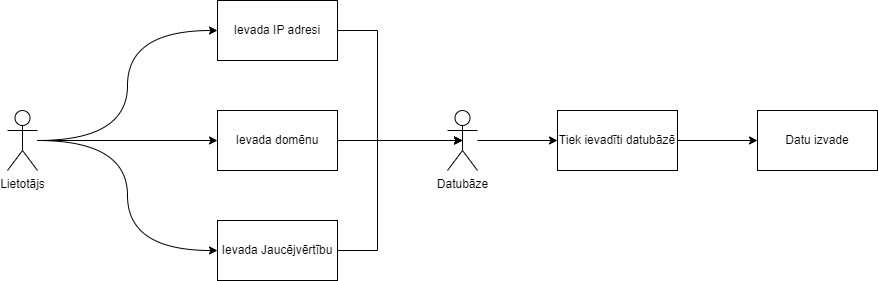


## Lietojumgadījumu diagramma

Lietotājs ievada vienu no trijām ievadēm, kas ir IP adrese, domēns un jaucējvērtība, datubāze apstrādā un tad izvada rezultātu (Skatīt 5. attēls).

**5. Attēls**

**Lietojumgadījumu diagramma**



## Sistēmas moduļu apraksts un algoritmu shēmas

Ir četri moduļi kas uzmeklē:

* IP adrešu uzmeklēšana
* Jaucējvērtību drošību uzmeklēšana
* Domēnu uzmeklēšana
* IP drošību uzmeklēšana

Lai sasniegtu IP adrešu uzmeklēšanu, modulis izmanto ‘Subprocess’ moduli, lai varētu izmantot Linux termināļa komandu ‘whois’. Termināla komanda ‘whois’ izmanto, lai izvilktu informāciju par IP adresi, kā reģistrāciju un kuram pieder šī IP adrese. Taču tas izmanto vēl ‘-h’ opciju, lai specificētu kādu ‘whois’ serveri mēs izmantosim. Šijā gadījumā projekts izmantos ‘whois.cymru.com’ serveri.

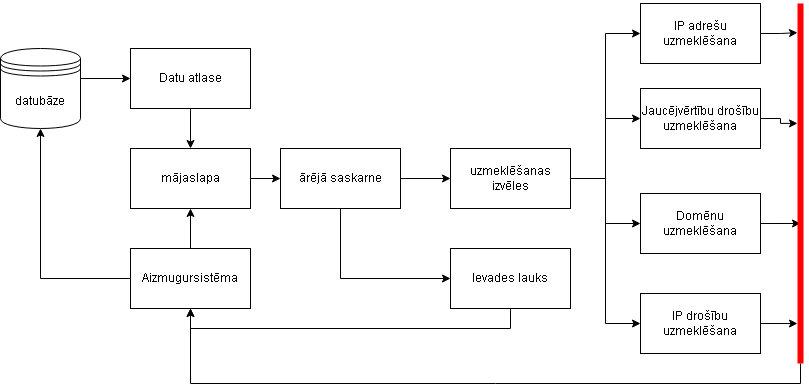
Lai sasniegtu jaucējvērtību drošību uzmeklēšanu, modulis izmanto lietojumprogrammas saskarne (API) ar Virustotal mājaslapu. Sākumā aizmugursistēma savienojas ar Virustotal mājaslapu caur API. Pēc to savienošanās, aizmugursistēma aizsūt jaucējvērtības datus un saņem šos datus kā ‘Json’ formātu. Pēc saņemšanas sistēma uzreiz pārbauda vai saņemtajos datos nav kļūdas paziņojumi. Ja jaucējvērtībai nav kļūdas paziņojums, tad aizmugursistēma pārbauda json formātu ar tekstu ‘ļaunprātīgs’. Ja ir ļaunprātīgs, tiks izvadīts ārā, ka jaucējvērtība nav droša, savādāk izprintēs ar tekstu ka ir drošs.

Lai sasniegtu domēnu uzmeklēšanu, modulis izmanto ‘Subprocess’ moduli, lai varētu izmantot Linux termināļa komandu ‘dig’. Komanda ‘dig’ Linux terminālī izmanto, lai savāktu DNS (domēnu nosaukuma serveris) informāciju, komandai nāk papildus līnija ‘+short’. Papildus komandu izmanto, lai izvilktu domēnas reģistrēto IP adresi.

Lai sasniegtu IP drošību uzmeklēšanu, modulis izmanto API ar Virustotal mājaslapu. Uzmeklēšanas sākumā, sistēma aizsūta API pieprasījumu, līdz ko ir savienots, sistēma pieprasa datus par IP adresi, kurā tiek formatēts ‘Json’ formātā. Pēc tam notiek pārbaude par kļūdas ziņojumu, tad notiek pārbaude vai IP adrese ir iespējami ļaunprātīga. Ja IP adrese ir ļaunprātīga, tad tiks izvadīts lietotājam, ka IP adrese iespējami nav droša, savādāk izvadīs to ka ir droša (Skatīt 6. attēlu).

1. **attēls**

**Sistēmas moduļu un apraksts**



# Lietotāju ceļvedis

Pirms lietošanas, lietotājam ir jāsaprot šie termini:

* IPv4
* ISP
* Domēns
* Jaucējvērtība (MD5, SHA-1, SHA-256)

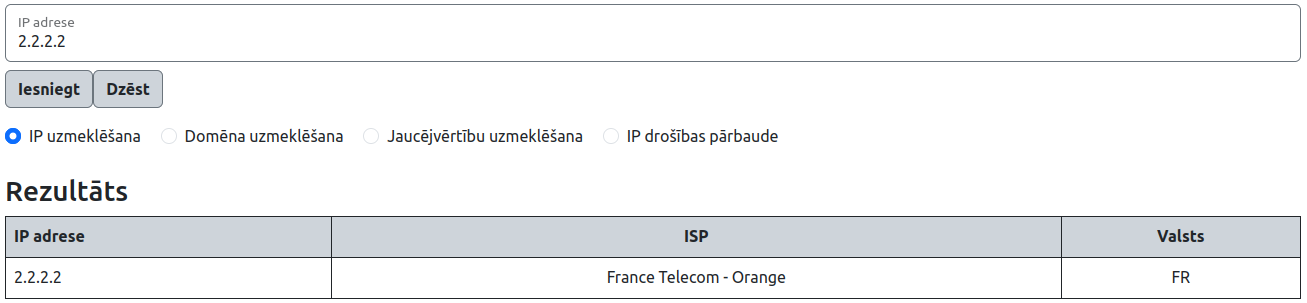
Iekļūt tīmekļa vietnē:

Sākumā atverat pārlūkprogrammu ar ko jums ir ērti lietot, tad ievadat URL laukā ‘localhost:1234’ un nospiežat uz tastatūra ‘Enter’, lai iekļūtu mājaslapas aplikācijā.

Mājaslapu aplikācijas izmantošana:

* IP adrešu uzmeklēšana:
  1. Noklikšķiniet uz ‘IP uzmeklēšana’ opcijas;

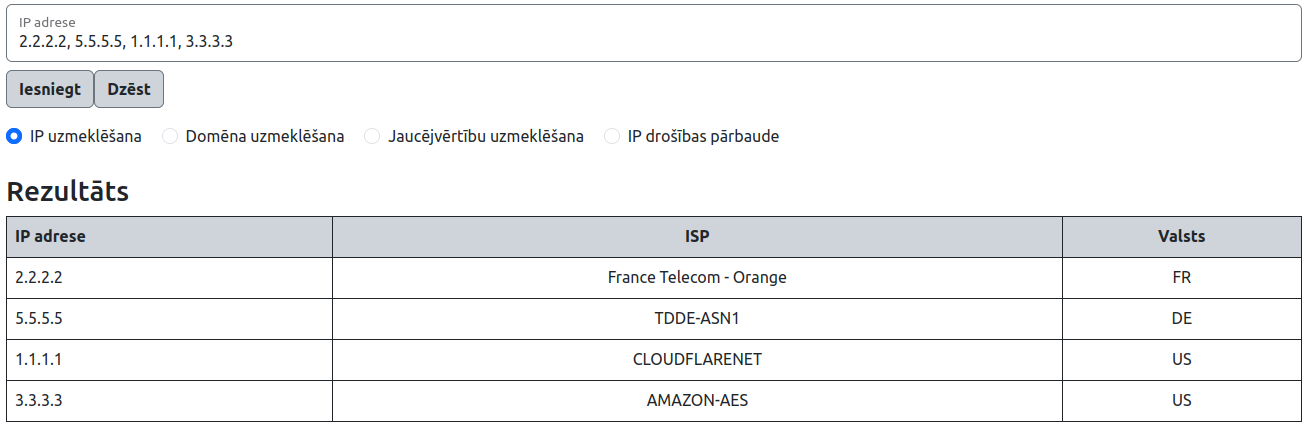
1. Ievadat derīgu IPv4 adresi ievades laukā;
2. Nospiediet “Iesniegt” pogu;
3. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur IP adresi, AS nosaukumu un valsti (Skatīt 7. attēlu.).
4. **attēls.**

**IP adrešu ievade**

Ja lietotājs vēlas uzmeklēt vairākas IP adreses, tad veic sekojošo:

1. Ievades laukā lietotājs ievada derīgu IPv4 adresi;
2. Pēc derīgas IPv4 adreses ievadīšanas, atdala ar komatu;
3. Lietotājs ievada nākošo IPv4 adresi;
   1. Ja lietotājs vēlas vairāk par divām IPv4 adresēm, atkārto soļus no 1 līdz 3.
4. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur vairākas IP adreses, AS nosaukumus un valstis (Skatīt 8. attēlu).
5. **attēls.**

**Vairākas ievades**



Atrisinājumi kopīgām problēmām:

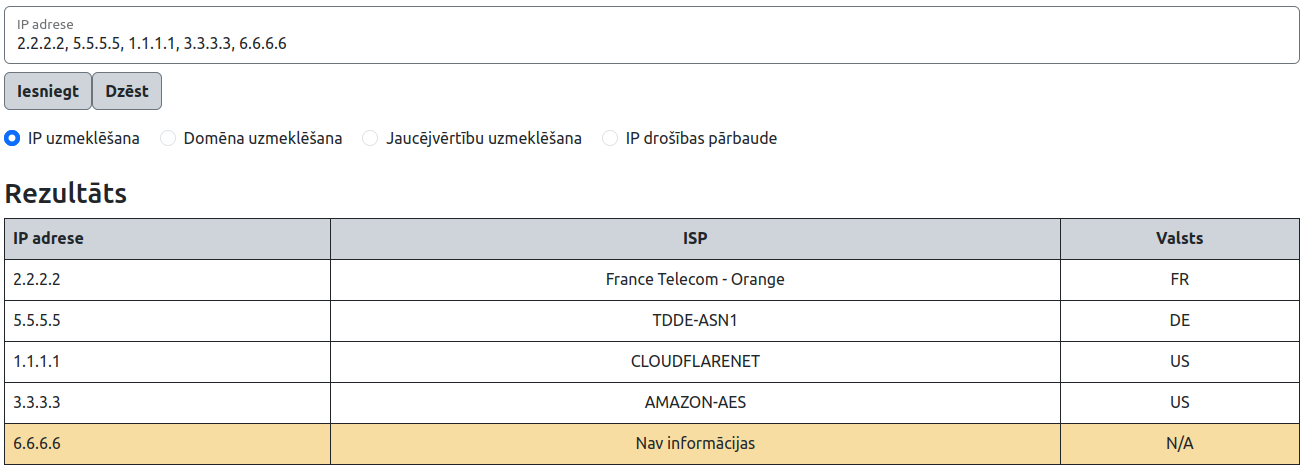
Ja lietotājam zem ievades lauka rādās ar sarkanu tekstu “Problēma: Kļūda ievadītajā indikatorā”, tad tas nozīmē, ka ievades laukā ir kāda nepareiza ievade. Lietotājam ir japārbauda:

* Ievadītais satur derīgu IPv4 adresi;
* Vai ir komats starp divām IPv4 adresēm.

Gadījumā rezultāta tabulā rādās ar dzeltenu rindas krāsu, tad tas nozīmē, ka ievadītā IPv4 adresei nav informācijas (Skatīt 9. attēlu).

1. **Attēls**

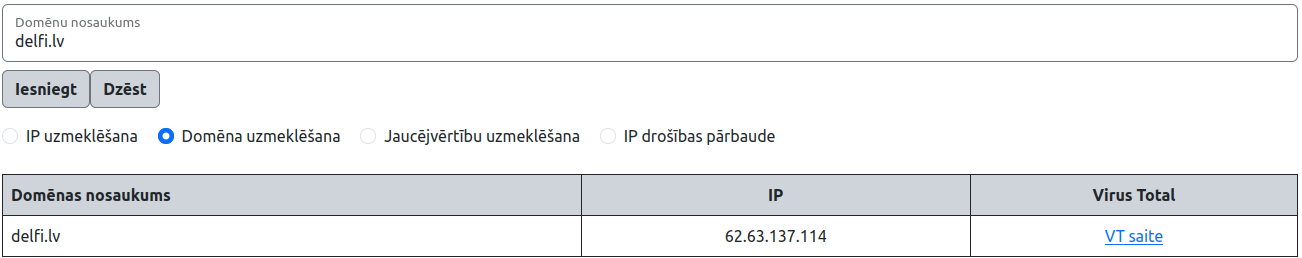
**Informācijas nav**



* Domēna uzmeklēšana
  1. Noklikšķiniet uz ‘Domēna uzmeklēšana’ opcijas;
  2. Ievadat derīgu domēnu adresi ievades laukā;
  3. Nospiediet “Iesniegt” pogu;
  4. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur Domēnas nosaukums, IP un Virus Total (Skatīt 10. attēlu.).

1. **Attēls**

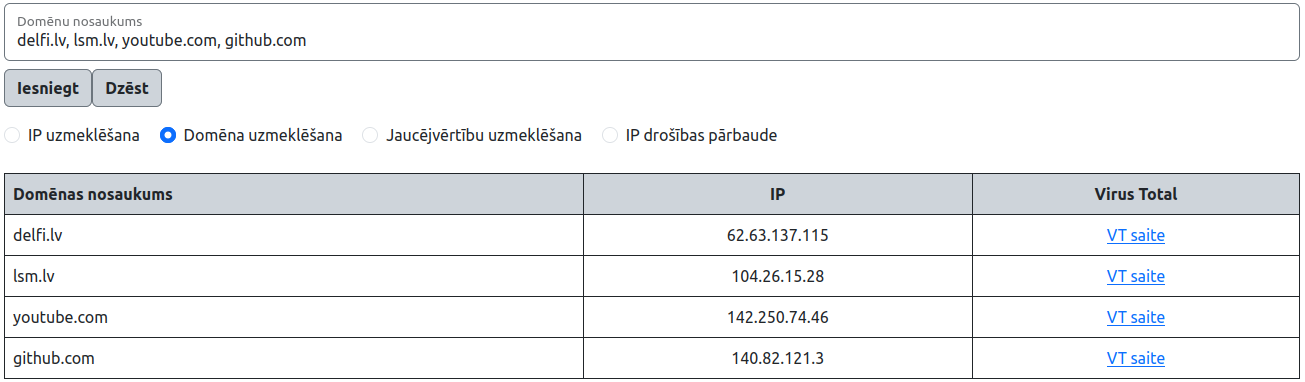
**Domēnu ievade**

****

Ja lietotājs vēlas uzmeklēt vairākas domēnas, tad veic sekojošo:

1. Ievades laukā lietotājs ievada derīgu domēnu adresi;
2. Pēc derīgas domēnu adreses ievadīšanas, atdala ar komatu;
3. Lietotājs ievada nākošo domēnu adresi;
   1. Ja lietotājs vēlas vairāk par divām domēnu adresēm, atkārto soļus no 1 līdz 3.
4. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur Domēnas nosaukums, IP un Virus Total (Skatīt 11. attēlu.).
5. **Attēls**

**Vairākas domēnu ievades**



Atrisinājumi kopīgām problēmām:

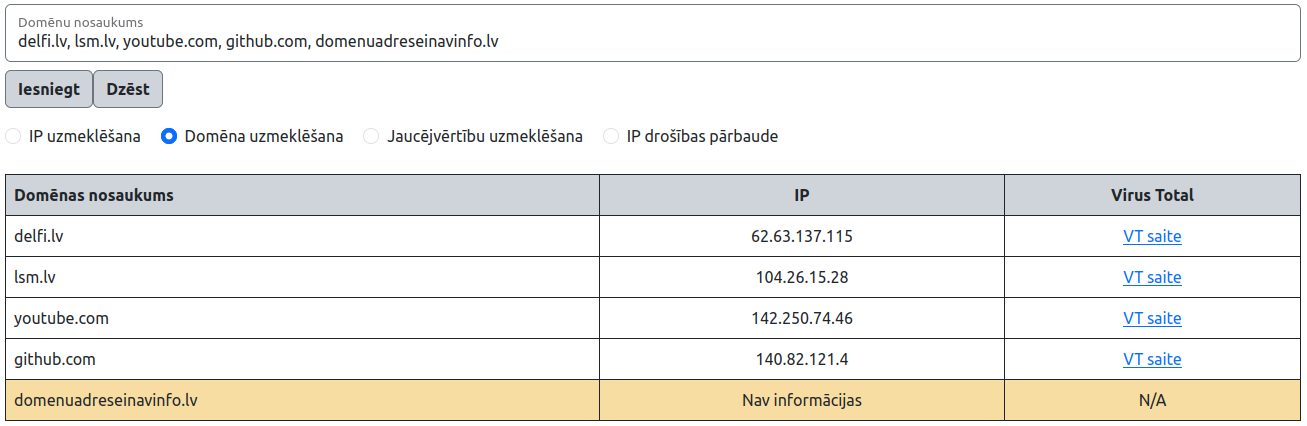
Ja lietotājam zem ievades lauka rādās ar sarkanu tekstu “Problēma: Kļūda ievadītajā indikatorā”, tad tas nozīmē, ka ievades laukā ir kāda nepareiza ievade. Lietotājam ir japārbauda:

* Ievadītais satur derīga domēnu adrese;
* Vai ir komats starp divām adresēm adresēm.

Gadījumā rezultāta tabulā rādās ar dzeltenu rindas krāsu, tad tas nozīmē, ka ievadītā domēnu adresei nav informācijas (Skatīt 12. attēlu).

1. **Attēls**

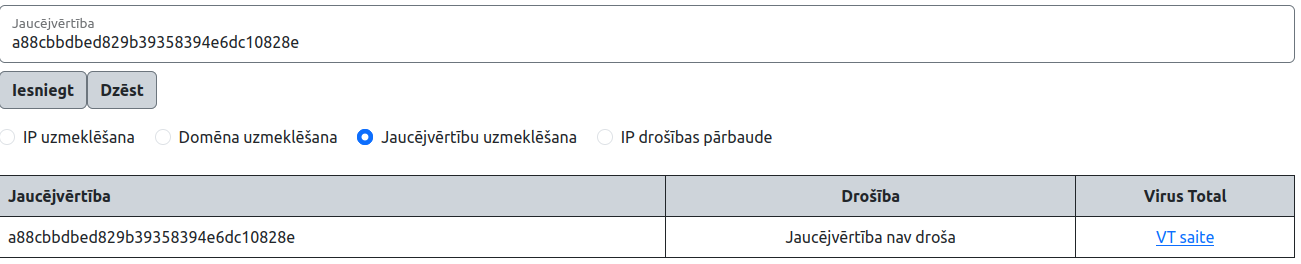
**Domēnam nav informācijas**

****

* Jaucējvērtību uzmeklēšana:
  1. Nokliķškiniet uz “Jaucējvērtību uzmeklēšana” opcijas;
  2. Ievadat derīgu jaucējvērtību ievades laukā;
  3. Nospiediet “iesniegt” pogu;
  4. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur jaucējvērtību, drošību un Virus Total (Skatīt 13. attēlu).

1. **Attēls**

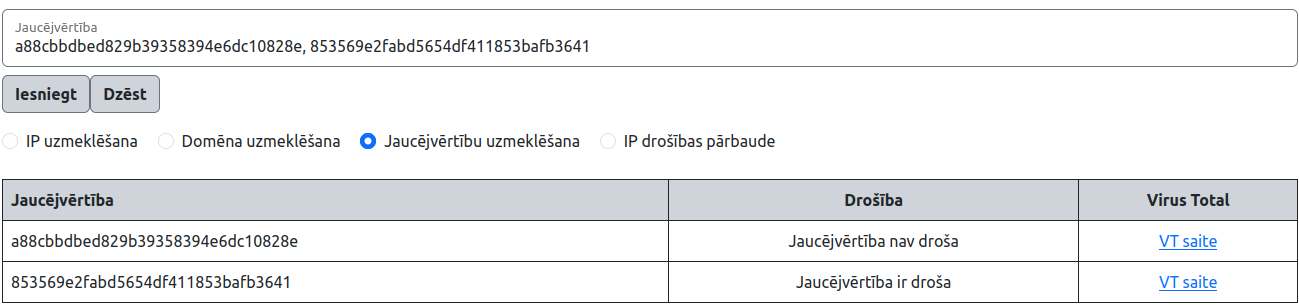
**Jaucējvērtību uzmeklēšana**



Ja lietotājs vēlas uzmeklēt vairākas jaucējvērtības, tad veic sekojošo:

1. Ievades laukā lietotājs ievada derīgu jaucējvērtību;
2. Pēc derīgas jaucējvērtības ievadīšanas, atdala ar komatu;
3. Lietotājs ievada nākošo jaucējvērtību;
   1. Ja lietotājs vēlas vairāk par divām jaucējvērtībām, atkārto soļus no 1 līdz 3.
4. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur jaucējvērtību, drošību un Virus Total (Skatīt 14. attēlu.).
5. **Attēls**

**Vairākas jaucējvērtību ievades**

****

Atrisinājumi kopīgām problēmām:

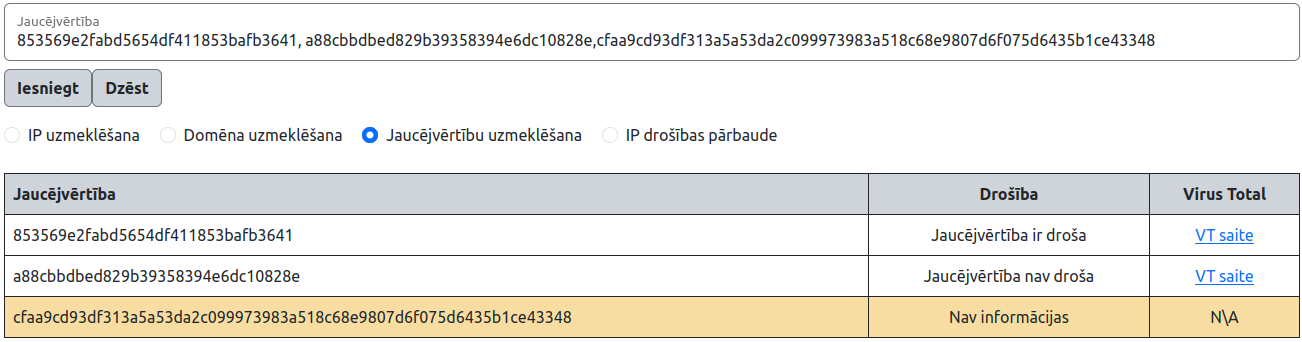
Ja lietotājam zem ievades lauka rādās ar sarkanu tekstu “Problēma: Kļūda ievadītajā indikatorā”, tad tas nozīmē, ka ievades laukā ir kāda nepareiza ievade. Lietotājam ir japārbauda:

* Ievadītais satur derīga jaucējvērtība (MD5, SHA-1, SHA-256);
* Vai ir komats starp divām jaucējvērtībām.

Gadījumā rezultāta tabulā rādās ar dzeltenu rindas krāsu, tad tas nozīmē, ka ievadītā domēnu adresei nav informācijas (Skatīt 15. attēlu).

1. **Attēls**

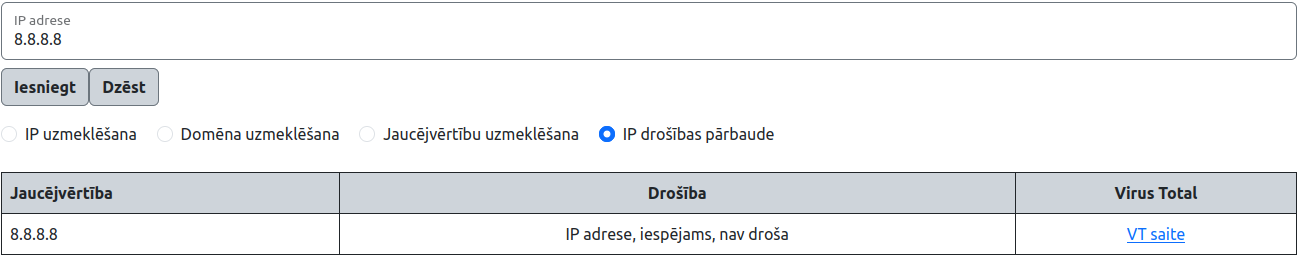
**Jaucējvērtībai nav informācijas**



* IP drošības pārbaude:
  1. Noklikšķiniet uz ‘IP uzmeklēšana’ opcijas;
  2. Ievadat derīgu IPv4 adresi ievades laukā;
  3. Nospiediet “Iesniegt” pogu;
  4. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur IP adresi, AS nosaukumu un valsti (Skatīt 16. attēlu.).

1. **Attēls**

**IP drošības ievade**

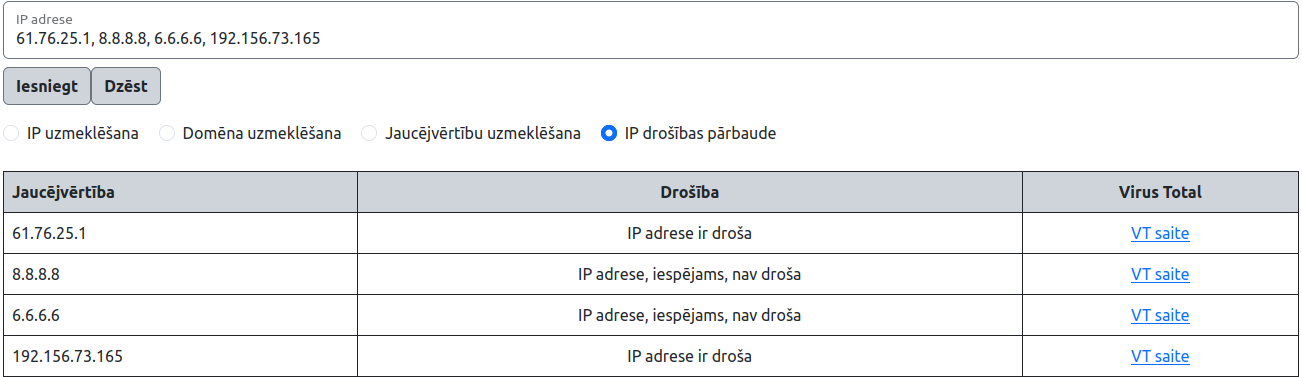


Ja lietotājs vēlas pārbaudīt vairākas IP adreses, tad veic sekojošo:

* + - 1. Ievads laukā lietotājs ievada derīgu IPv4 adresi;
      2. Pēc derīgas IPv4 adreses ievadīšanas, atdala ar komatu;
      3. Lietotājs ievada nākošo IPv4 adresi;
  1. Ja lietotājs vēlas vairāk par divām IP adresēm ievadīt, atkārto soļus no 1 līdz 3.
     + 1. Zem ievades lauka, jābūt redzamam rezultātu tabula, kas satur IP adresi, drošība un Virus Total (Skatīt 17. attēlu).

1. **Attēls**

**Vairākas IPv4 drošību ievades**



# Testēšanas dokumentācija

## Izvēlētās testēšanas metodes, rīku apraksts un pamatojums

Tīmekļa vietne ir testēta pēc melnās kastes metodes. Metode testēs galvenās funkcijas projektam un to sagaidāmo rezultātu.

## Testpiemēru kopa

**1. tabula**

**Testpiemēru kopa**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Testpiemēru ID | Testpiemēra nosaukums | Testpiemēra izpildes nosacījumu | Testpiemēra izpildes solis | Testpiemēra ievades dati | Testpiemēra sagaidāmais rezultāts | Prasības ID |
|  | Black Box | | | | | |
| TP.01 | Projektu palaišana | Projekts tiek palaists uz Linux operētājsistēmas | 1. Projekts ir mapē  2. Komandu rinda palaiž projektu | Docker-compose up | Ierakstot localhost:1234, atverās mājaslapa | PR.01 |
| TP.02 | Mājaslapu piekļuve | PR.01 | 1. Adrešu laukā ierakstot localhost:1234  2. Nospiežot Enter | Localhost:1234 | Tīmekļa vietne būs parādīta lietotājam uz pārlūkprogrammas | PR.02 |
| TP.03 | IP adrešu uzmeklēšana | PR.02 | 1. zem ievades lauka ir izvēlne  2. noklikšķiniet uz izvēlnes | Kreisais peles klikšķis | Tīmekļa vietnē, uz ievades lauka būs teksts “IP adrese” un zila radio poga | PR.03 |
| TP.04 | IP adrešu ievadīšana | PR.03 | 1. Ievades laukā jāievada IP adrese  2. noklikšķināt iesniegt pogu | IPv4 adrese | Informācijas par IP adresi | PR.04 |
| TP.05 | Domēnu uzmeklēšana | PR.02 | 1. zem ievades lauka ir izvēlne  2. noklikšķiniet uz izvēlnes | Kreisais peles klikšķis | Tīmekļa vietnē, uz ievades lauka būs teksts “domēnu nosaukums” un zila radio poga | PR.05 |
| TP.06 | Domēnu adrešu ievadīšana | PR.05 | 1. Ievades laukā jāievada Domēnu adrese  2. noklikšķināt iesniegt pogu | Domēnu nosaukums | Informācijas par Domēnu adresi | PR.06 |
| TP.07 | Jaucēvērtību uzmeklēšana | PR.02 | 1. zem ievades lauka ir izvēlne  2. noklikšķiniet uz izvēlnes | Kreisais peles klikšķis | Tīmekļa vietnē, uz ievades lauka būs teksts “Jaucējvērtība” un zila radio poga | PR.07 |
| TP.08 | Jaucējvērtību ievadīšana | PR.07 | 1. Ievades laukā jāievada Jaucējvērtība  2. noklikšķināt iesniegt pogu | Jaucējvērtību nosaukums | Informācijas par jaucējvērtību | PR.08 |
| TP.09 | IP adrešu drošību pārbaude | PR.02 | 1. zem ievades lauka ir izvēlne  2. noklikšķiniet uz izvēlnes | Kreisais peles klikšķis | Tīmekļa vietnē, uz ievades lauka būs teksts “IP adrese” un zila radio poga | PR.09 |
| TP.10 | IP adrešu drošību ievadīšana | PR.09 | 1. Ievades laukā jāievada IP adrese  2. noklikšķināt iesniegt pogu | IPv4 adrese | Informācijas par IP adresi | PR.10 |
| TP.11 | Dzēst datus | PR.02 | Uzklikšķinot uz Dzēst pogas | Kreisais peles klikšķis | Tabulas tiek dzēstas, ievades lauks tiek dzēsts | PR.11 |
| TP.12 | Izvēloties no vienas uzmeklēšanas uz otra | PR.02  PR.03/PR.05/  PR.07/PR.09 | Uzklikšķinot uz izvēlnes pogas kas nav izvēlets | Kreisais peles klikšķis | Automātiski maina uz citu uzmeklēšanu ar tabulu redzamību un ievades lauku | PR.12 |

## Testēšanas žurnāls

**2. Tabula**

**Testēšanas žurnāls**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Testēšanas ID | Testpiemēru ID | Testpiemēra nosaukums | Statuss | Kļūda | Pielikums |
| T.1 | TP.01 | Projektu palaišana | Veiksmīgs |  | Kļūdas bijušas iepriekš, tamdēļ ka nepareiza konfigurācija |
| T.2 | TP.02 | Mājaslapu piekļuve | Veiksmīgs |  |  |
| T.3 | TP.03 | IP adrešu uzmeklēšana | Veiksmīgs |  |  |
| T.4 | TP.04 | IP adrešu ievadīšana | Veiksmīgs |  | Iepriekš bijušas problēmas ar ievadi un izvadi, dēļ konfigurācijas aizmugursistēmā |
| T.5 | TP.05 | Domēnu uzmeklēšana | Veiksmīgs |  |  |
| T.6 | TP.06 | Domēnu adrešu ievadīšana | Veiksmīgs |  | Iepriekš bijušas problēmas ar ievadi un izvadi, dēļ konfigurācijas aizmugursistēmā |
| T.7 | TP.07 | Jaucēvērtību uzmeklēšana | Veiksmīgs |  | Bija problēmas ar izvēlnes saglabāšanos |
| T.8 | TP.08 | Jaucējvērtību ievadīšana | Veiksmīgs |  | Iepriekš bijušas problēmas ar ievadi un izvadi, dēļ konfigurācijas aizmugursistēmā |
| T9 | TP.09 | IP adrešu drošību pārbaude | Veiksmīgs |  |  |
| T.10 | TP.10 | IP adrešu drošību ievadīšana | Veiksmīgs |  | Iepriekš bijušas problēmas ar ievadi un izvadi, dēļ konfigurācijas aizmugursistēmā |
| T.11 | TP.11 | Dzēst datus | Nav veiksmīgs | Nav pareizi konfigurēta poga aizmugursistēmā | Ir skatīts kā labot šo problēmu, bet nav līdz galam izpētīts |
| T.12 | TP.12 | Izvēloties no vienas uzmeklēšanas uz otra | Nav veiksmīgs | Nav pareizi konfigurēta ārējā saskarnē | Ir skatīts, bet neviena metode nav izdevusies |

# Secinājumi

Projektu izstrāde nav bijusi viegla, bet ir bijušas funkcijas ko varēju pievienot bez lielām problēmām. Vieglāk bija ar jaunu funkciju pievienošanu, tam kam bija daudz informācijas par to kā pievienot funkciju projektā. Kā arī bija daudz vieglāk pievienot funkcijas pēc prakses vadītāja piemēriem un ieteikumiem.

Taču grūtāk gāja izsprast jaunas programmatūras un sintakses, kā arī pievienot MySQL un Docker savā projektā. Problēmas bija dēļ tā ka ir par maz informācijas, par maz zināšanas vai abi. Vieglāk bija pievienot, ja pirms tam esmu testējis kādā citā mapē, kurā nevarēju sabojāt savu projektu.

Lai gan uzņēmums ir apmierināts ar to kas tika paveikts, ir vēl jāpieliek pāris funkcijas, kā papildus uzmeklēšanas iespējas, vairāk rezultātu izvade.

# Lietoto saīsinājumu saraksts

**8. tabula**

**Lietoto saīsinājumu saraksts**

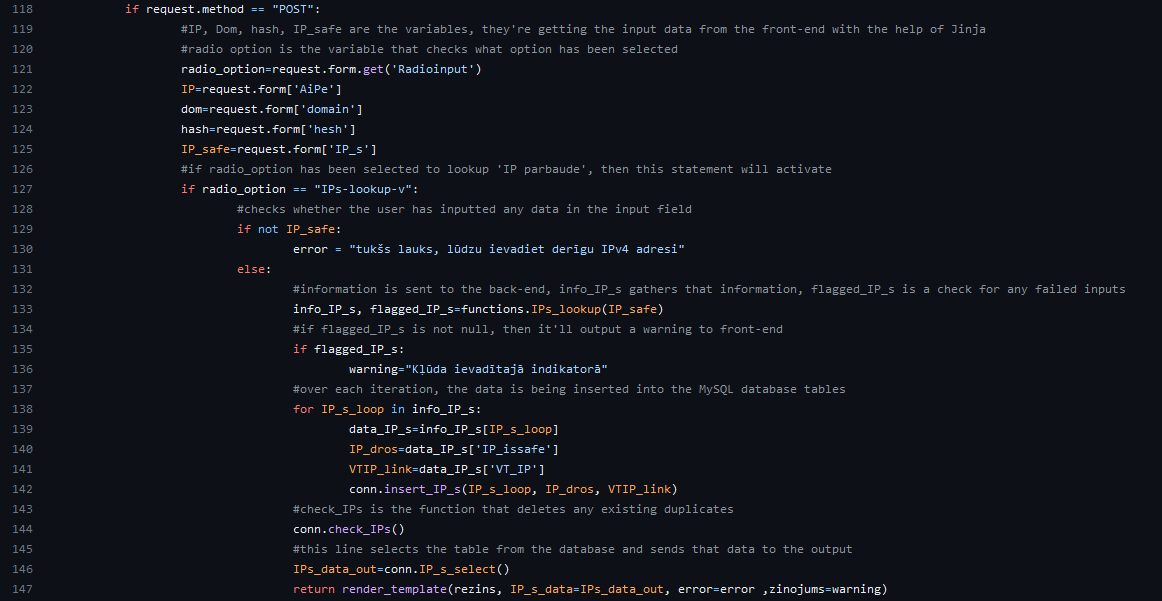
|  |  |  |
| --- | --- | --- |
| **Nr** | **Termins** | **Skaidrojums** |
| 1 | IPv4  IP adrese | Interneta Protokols versija 4 – izmantots lai identificētu un komunicētu ar ierīcēm tīklājā, piemēram, internetā. Reģistrē ierīci ar unikālu ciparu adresi, ko sauc par IP adrese, satur četras skaitļu kopas no 0 līdz 255, kas atdalītas ar punktiem (piem. ‘1.1.1.1, 10.10.10.10, 255.255.255.255’) |
| 2 | Autonomās sistēmas numurs (ASN) | Unikāls identifikators kas tiek reģistrēts uzņēmumiem un organizācijām, ļauj efektīvāku datu sūtīšanu uz citiem tīklīem. |
| 3 | Hiperteksta iezīmēšanas valoda (HTML) | Iezīmēšanas valoda tiek izmantota, lai pārlūkprogrammas vizualizē tīmekļa lapas, lai lietotāji tos varētu apskatīt. |
| 4 | Vienotais resursu vietrādis (URL – Uniform Resource Locator) | Rakstzīmju virkne ko izmanto, lai atpazītu un atrastu resursus internetā. Sākumā sastāv protokols, lai piekļūtu interneta resursam (HTTP vai HTTPS), domēna nosaukums (delfi.lv) un norādīts resurss uz tā servera (piem. “/news/zinas”) (pilnais piemērs ‘https://delfi.lv/news/zinas’) |
| 5 | Hiperteksta Transporta Protokols (HTTP) | Protokols kas tiek izmantots lai pārsūtītu datus starp mājaslapu serveri un klientu. HTTPS ir drošs saziņas protokols. |
| 6 | Virustotal - mājaslapas serviss | mājaslapa kas specializējas identificēt ļaunprogrammatūru failā, jaucējvērtībā utt. |
| 7 | Starpniekserveris | ir serveris vai rūteris kas ir starp serveri un lietotāju, ja lietotājs pieprasa kaut ko no mājaslapas, tad starpniekserveris saņem šos datus un mēģina izpildīt šos iesūtītos datus pats, taču ja starpniekserverim neizdodas, pārsūta datus uz īsto serveri, kurā pēc tam, nosūta šos gatavos datus atpakaļ lietotājam. |
| 8 | Pārlūkprogramma | Programma ko var izmantot, lai pārlūkotu internetu, bez programmu lejuplādes, piemērs kas ir lietojumprogramma: Firefox, Chrome, Edge, Opera |
| 9 | Aizmugursistēma | Sistēma kas tiek apstrādāts ko lietotājs neredz, aizmugursistēma ir tas kas apstrādā ievadītos datus vai apstrādā to ko lietotājs vēlas no programmas. |
| 10 | Jaucējvērtība | pārveido parastu tekstu par šifrētu tekstu kur rakstu zīmes būs vienmēr tādā pašā garumā |
| 10.1 | MD5 | ir šifrēšanas algoritms kas tika publicēts 1992. gadā |
| 10.2 | SHA-1 | ir šifrēšanas algoritms kas tika publicēts 1995. gadā |
| 10.3 | SHA-256 | ir šifrēšanas algoritms kas tika publicēts 2001. gadā |
| 11 | MySQL | datubāzes uzturēšanas sistēma |
| 12 | Docker | Programma ar kuru var palaist vairākas lietojumprogrammas vienlacīgi, lai vairākas programmas strādātu kā viena liela programma |
| 13 | Jinja2 | Ieteicams izmantot kopā ar flask, lai varētu pārsūtītu datus no aizmugursistēmas uz ārējo saskarne vienkāršāk |
| 14 | Flask | Python modulis kas nāk kopā, lai programmētājam būtu vienkāršāk izveidot mājaslapu |
| 15 | ISP | Interneta pakalpojuma sniedzējs, piemēram, TET, elektrons, Bite utt. |
| 16 | Python | Programmēšanas valoda |
| 17 | Nginx | Starpniekserveris |

# Literatūras un informācijas avotu saraksts

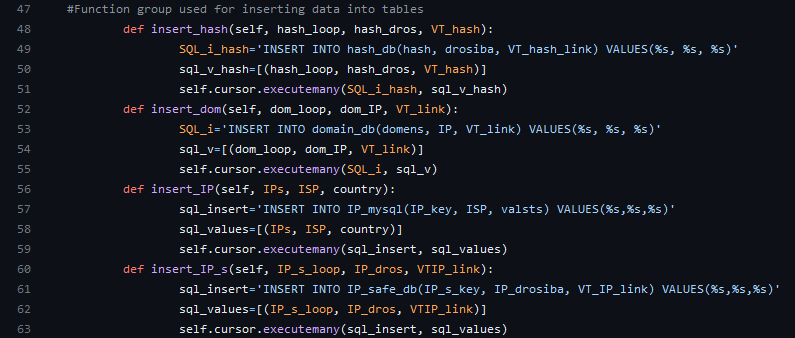
* Iegūtie termini:
  + <https://termini.gov.lv/>
* Atkļūdošanas mājaslapa:
  + <https://stackoverflow.com/>
* Dokumentācijas:
  + <https://docs.python.org/3/>
  + <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
  + Par MySQL kursoriem:
    - <https://dev.mysql.com/doc/connector-python/en/connector-python-api-mysqlcursor.html>
  + <https://flask.palletsprojects.com/en/2.3.x/>
* Vienkāršas pamācības:
  + <https://www.geeksforgeeks.org/>
  + <https://www.w3schools.com>

# Pielikums

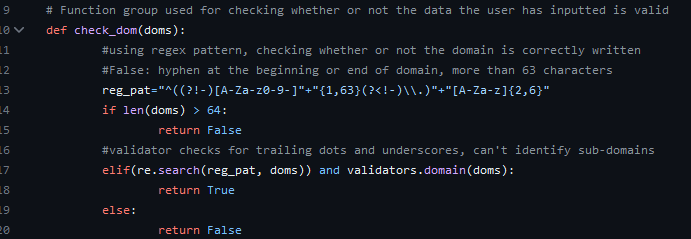
17. Attēls



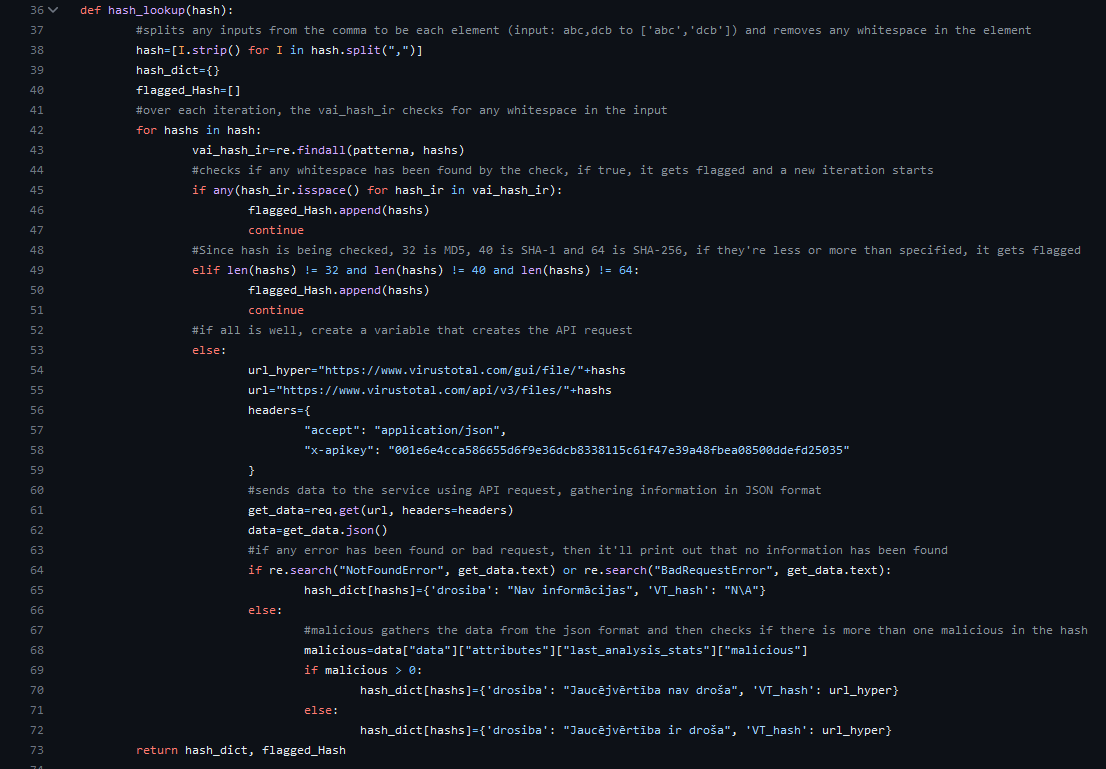
18. attēls



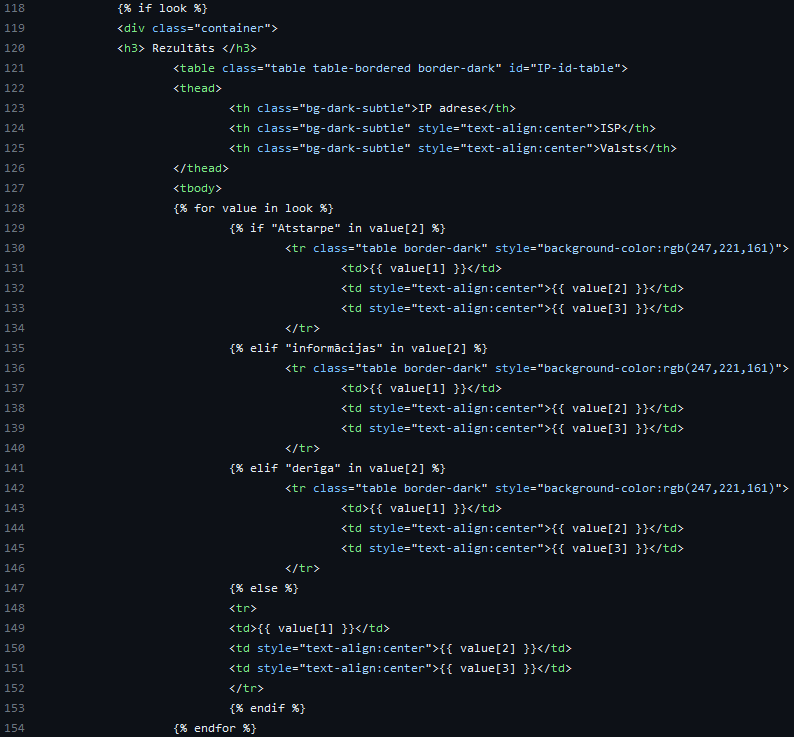
19. Attēls



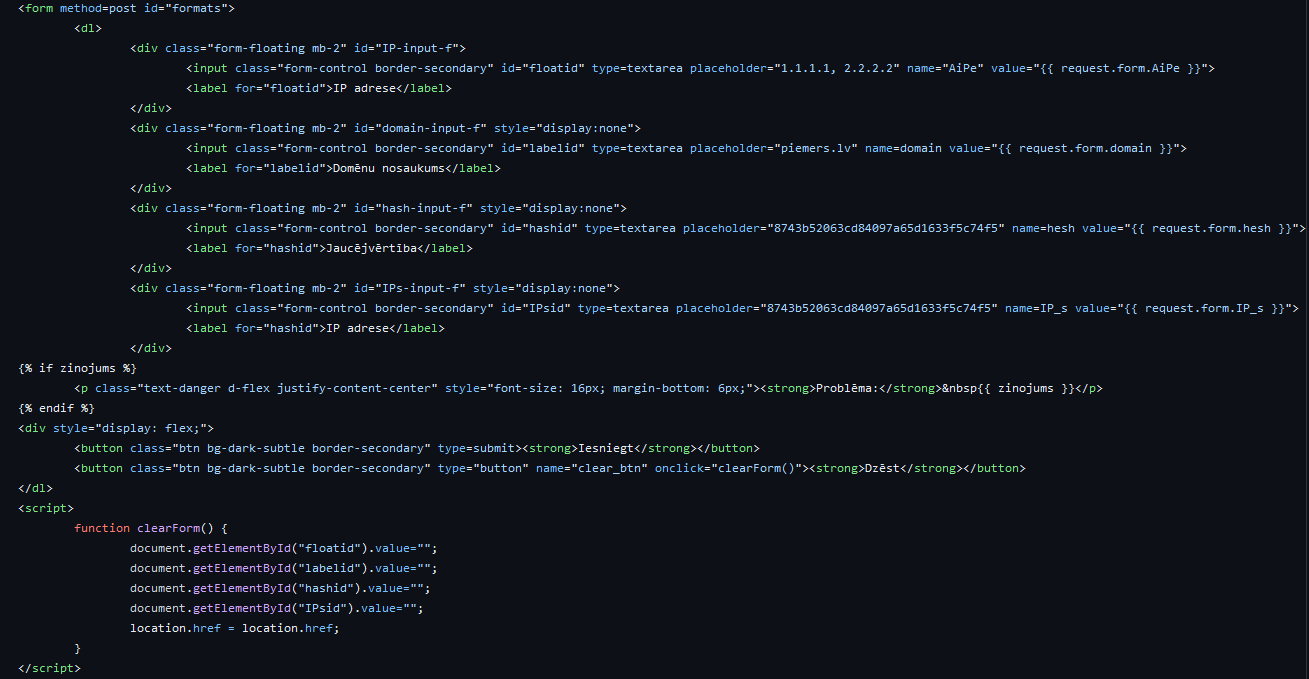
20. Attēls



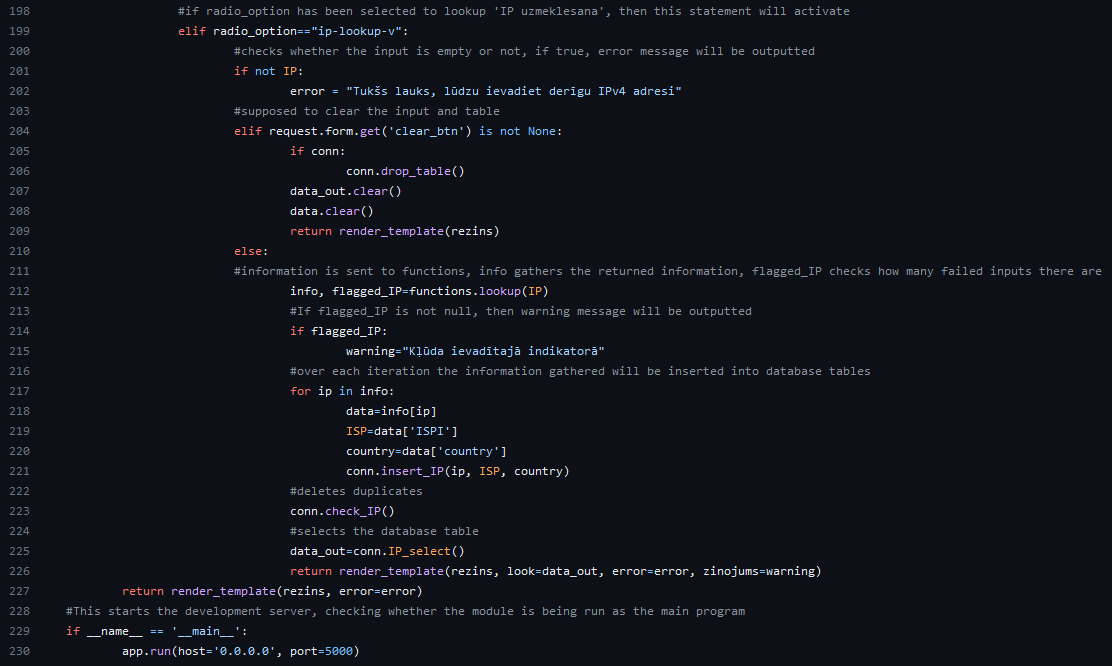
21. Attēls



22. Attēls



23. Attēls



Saite uz github projektu:

<https://github.com/Ekluzivs/Prakse_Zvagulis_4PT>