**RĪGAS VALSTS TEHNIKUMS**

DATORIKAS NODAĻA

Izglītības programma: Programmēšana

**Paskaidrojošais raksts**

**Darba meklēšanas sistēma**

Autors:

DP3-2 grupas audzēkis

Mareks Pārums

Rīga, 2024

SATURS

[IEVADS 3](#_Toc169540268)

[1.UZDEVUMA NOSTĀDNE 4](#_Toc169540269)

[1.1. Sistēmas lietotāji un to mijiedarbība 4](#_Toc169540270)

[1.2. Uzdevuma risināšanas līdzekļu izvēle 5](#_Toc169540271)

[2. DATU MODELĒŠNA 6](#_Toc169540272)

[2.1 Datu bāzes fiziskā struktūra 6](#_Toc169540273)

[2.2 Entītiju relāciju datu modelis 7](#_Toc169540274)

[3. DATU BĀZES IZSTRĀDE 9](#_Toc169540275)

[3.1. Datu bāzes fiziskā struktūra 9](#_Toc169540276)

[3.2. Datu bāzes optimizēšana 10](#_Toc169540277)

[4. SISTĒMAS FUNKCIONĀLO PRASĪBU REALIZĒŠANA 11](#_Toc169540278)

[4.1 Datu pievienošana ,labošana, dzēšana 11](#_Toc169540279)

[4.2 Datu validācija 15](#_Toc169540280)

[4.2.1. Datu izgūšana 16](#_Toc169540281)

[4.2.2 Datu apstrāde 17](#_Toc169540282)

[5. Nobeigums 18](#_Toc169540283)

# IEVADS

Tālāk aprakstīta mācību projekta mērķis ir izstrādāt darba / prakses meklēšanas sistēmu. Tā ļauj efektīvi meklēt darbu izmantojot plašus filtrus un darba veidu variācijas.

Tradicionālās darba meklēšanas platformas bieži vien saskaras ar problēmām, piemēram, neefektīvu filtrēšanu, nekonkrētiem vakances aprakstiem un neiespēju precīzi pielāgot darba meklēšanas kritērijus individuālajām vēlmēm. Tas var novest pie laika zaudēšanas, sliktas pieredzes un nepieciešamības apskatīt lielu skaitu nepiemērotu vakancu.

Lai risinātu augstāk minētas problēmas, izlēmu veidot savu darba meklēšanas sistēmu, kura sevī ietvers labāko no konkuretniem un paplašinās iespējas, atvieglinot saziņu ar darba dēvējiem/ ņemējiem un darba meklēšanas procesu kopumā.

Šobrīd pats sastapos ar īstu izaicinājumu - prakses meklēšanu, un esmu pamanījis, ka meklēšanas sistēmām nepietiek īpašu kategoriju, kā arī nevar sazināties ar darba devēju uzreis mājaslapā, kas, manuprāt, būtu ļoti noderīgi pirmatnējai saziņai.

Viena no līdzīgām sistēmām ir VisiDarbi.lv, kura ir populāra Latvijā un arī dod pietiekamu funkcionālu darba meklēšanai. Bet šajā sistēmā nepietiek filtru un arī tādu darba veidu kā vasaras darbs, prakse un vienreizējais darbs.

No otras puses ir GetaPro - vienreizēja darba meklēšanas resurs, kuram ir daudz ērtāks un saprotamāks dizains, bet nav pastāvīga darba kategorijas. GetaPro dod plašāku iespēju meklēšanai. Sistēmā arī ir reitings, kuru var redzēt visi lietotāji un atrast labāko darba izpildītāju.

Pēc šo un citu sistēmu analīzes secināju, ka neviena neietver sevī visus darba veidus un ērtu funkcionālu un dizainu. Tas ir tāpēc, ka katrā ir ļoti šaura mērķauditorija. No lietotāja puses nav ērti meklēt dažādas vietnes dažādam nolūkam. Arī mazāks ir darbadevēju loks, jo latrs ir savā vietnē, un tos apvienojot var gūt lielāku interesi no litotāju puses, kas savukārt veicinās interesi no darbadevēju puses.

# 1.UZDEVUMA NOSTĀDNE

Projekta mērķis ir izstrādāt ērtu, funkcionālu vietni darba meklēšanai visām pieļaujama darba vecuma grupām. Ts ietvers labākos aspektus no konkurentiem. Mūsdienās darba meklēšana tiek veikta tiešsaistē, tāpēc pieaug pieprasījums darba meklēsanas resursos, kuri piedāvātu iespēju efektīvi un viegli atrast darbu.

Sistēmā bus pieejama šāda funkcionalitate:

* Darba došanas sludinājuma izvietošana
* Katra sludinājuma apskate minimizētā un izklāstā veidā
* Sludinājumu filtrēšana pēc lietotāja vajadzībām
* Profila izveidošana, rediģēšana
* Darba sludinājumu pievienošana “favorītiem”
* Paroles maiņa jebkuram lietotājam
* Profila veidošana atbilstoši privātpersonām un uzņēmumiem
* Administrātoriem iespēja moderēt lietotājus un sludinājumus

Sistēmas prasības:

* Optimizācija, lai sistēma strādātu droši jebkuros apstākļos
* Ērts un viegli saprotams dizains, lai jebkurš lietotājs viegli varētu orientēties sistēmā un izmantot to bez grūtībām

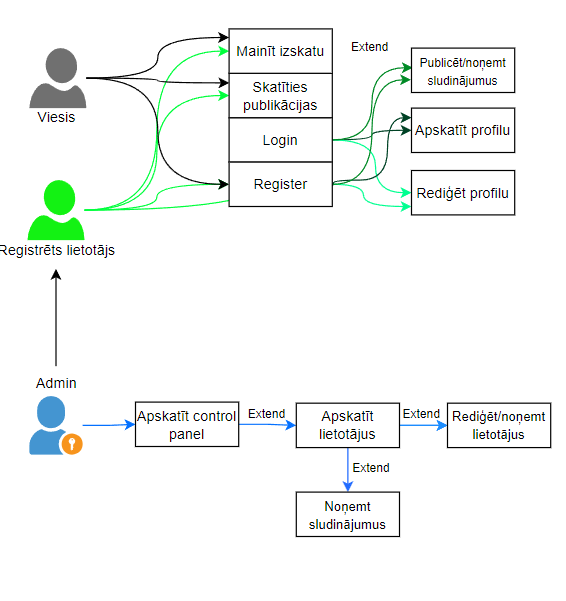
## 1.1. Sistēmas lietotāji un to mijiedarbība

Lai veiksmīgāk izveidot efektīvu un ērtu sistēmu, ir nepieciešams saprast lietotajus, un kādas tiem bus iespējas. Lai izpētīt šo jautajumu tika izveidota lietotaju un to mijiedarbības shēma (1.1 att.). Sistēmu izmantos 3 lietotāju grupas: reģistrētie lietotāji, nereģistrētie lietotāji un administrators.

Nereģistrēts lietotājs (viesis) varēs apskatīt sludinājumus, būs iespēja reģistrēties ar epastu, izveidojot paroli un ievadot vārdu.

Reģistrēti lietotaji savukārt varēs darīt to pašu ko vieslietotājs. Būs arī paplašinātas iespējas -paroles maiņa sludinājumu izvietošana un dzēšana, sludinājumu pievienošana favorītiem.

Administrātoram Būs pieieja visam reģistrēta lietotāja funkcionālam, kā arī iespēja noņemt citu sludinājumus pēc nepieciešamības, kā arī pieeja sistēmas pārvaldīšanai.



1.1 att. Lietojumu gadījumu diagramma

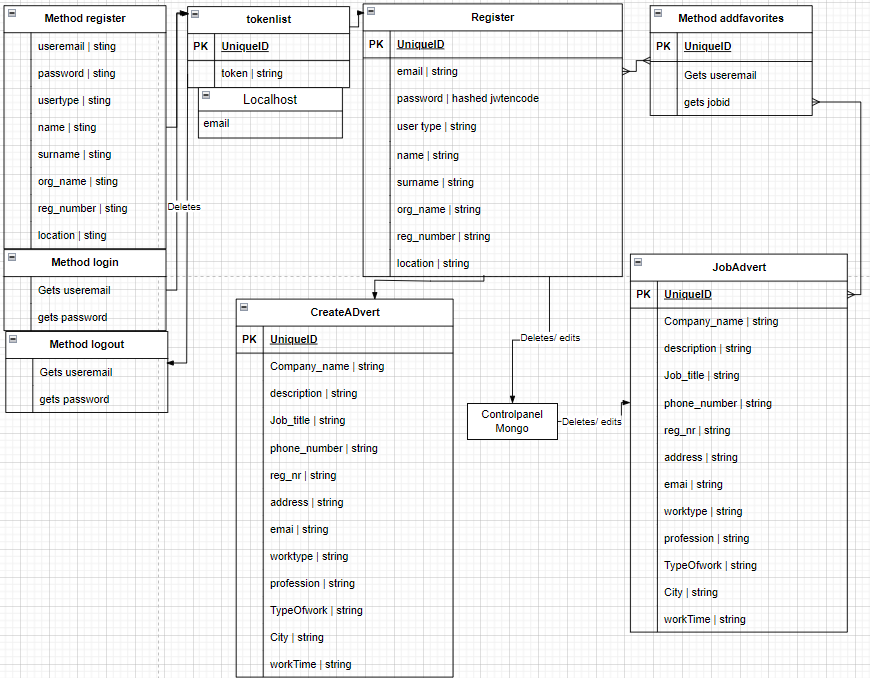
## 1.2. Uzdevuma risināšanas līdzekļu izvēle

Datu bāzes izstrādei un pārvaldei tika izmantots MongoDB, jo tas ļauj gan veidot datu bāzi, gan to pārvaldīt. Datu bāzes funkcijām tika izmntota python valoda kopā ar Flask bibliotēku, kura ļauj atvieglot funkciju rakstīšanu.

Mājaslapas veidošnai izmantoju vue.js, jo tas ļauj viegli integrēt rūteri un atvieglo mājaslapas funkcionāla izveidi. Kopā ar vuejs tika pielietots arī Tailwind, kas atvieglina un paplašina mājaslapas dizaina un atsucīguma izveidi

# 2. DATU MODELĒŠNA

## 2.1 Datu bāzes fiziskā struktūra



2.1 att. UML diagramma

* “Method login”, “Method register”, Method logout –metodes, kas kontrolē auentifikāciju, tā sazinās ar DB, pārbaudot lietotāju un tā informāciju.
* “ControlPanel” – klase, kas nodrošina funkcijas administratoriem. Tā sazinās ar
* datubāzi.
* “Register” – entitijs, kurš glabā informāciju par reģistrēto ietotāju.
* “Tokenlist” – klase, kas nodrošina drošu un ērtu saziņu ar datubāzi un glabā lietotāju sesijas - cookies.
* “Method regiter“ - metode, kas izveido jauno lietotāju, pieņemot epastu, paroli, vārdu, uzvārdu, organizācijas nosaukumu un reģ.Nr, un atrašanas vietu, kā arī pati metode izveido lietotāja tipu- organizācija vai privātpersona, atbilstoši lietotāja izvēlei.
* “Method logout” – funkcija, kas izdzēs cookies un epastu kas glabājas localhost, un atsēdz lietotāju no mājaslapas.
* “Method login” – metode, kas pieņem epastu un paroli, un pieslēdz lietotāju izveidotam profilam.
* “Method addFavorites”- metode, kas ļauj lietotājam pievienot sludinājumus favorītiem. Pieņem lietotāja epastu, un izvēlētā sludinājuma ID.
* “Lokalhost”- glabā lietotāja epastu, ir nepieciešams funkciju darbībai
* “JobAdvert“ - entitijs, kas glabā sevī informāciju par darba sludinājumiem

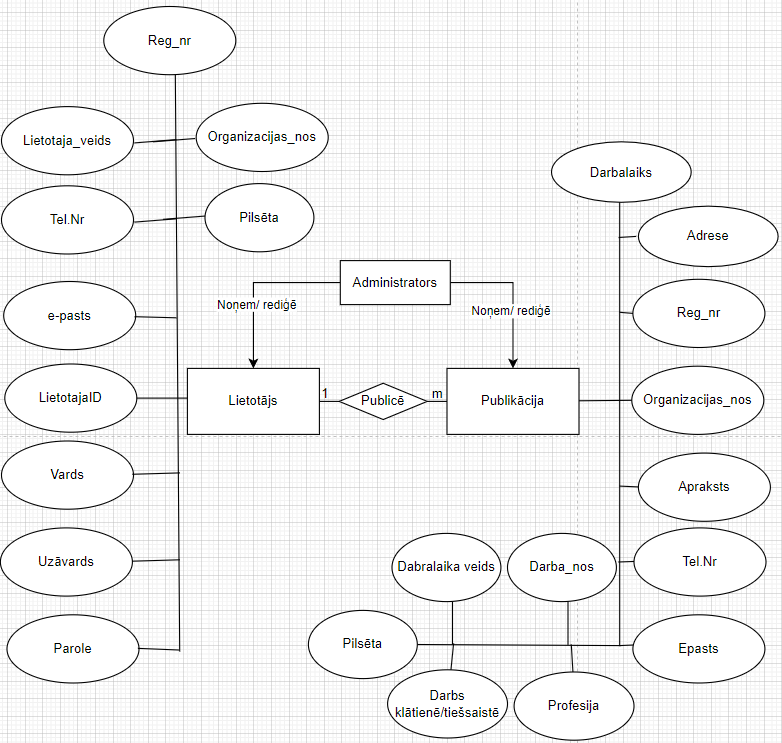
## 2.2 Entītiju relāciju datu modelis

Projektēšanas laikā tiks izstrādāts ER modelis (2.2.1. att.). Ar tā palīdzību ir vieglāk saprast datubāzes struktūru. Tādā veidā ir viegli izvairīties no kļūdām.

ER modelis sastāv no 4. entītijām.

Lietotājs - apraksta sistēmas lietotāju. Piemītošie atribūti: lietotāja tips, epasts, parole, vārds, uzvārds, organizācijas nosaukums un reģ.Nr un atrašanas vieta.

Publikācija - apraksta publikācijas, kas ir izvietotas vietnē. Piemītošie atribūti: Nosaukums, apraksts, Orgaņizācijas nosaukums, atrašanas vieta, tālruņa Nr., e-pasts, darbalaika veids, professija, darba veids, grafika veids.



Att. 2.2.1 ER modelis

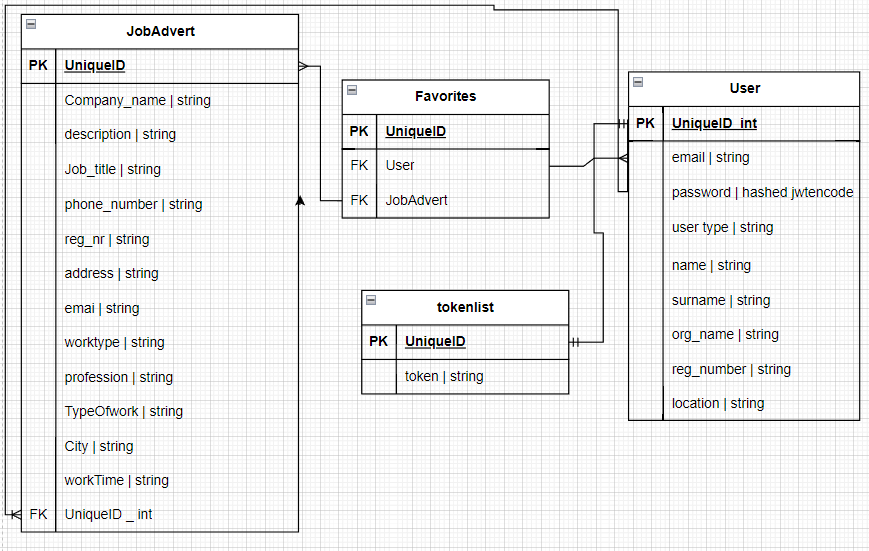
Datu bāzes tabulu savstarpējas attiecības:

* Starp tabulām "Lietotāji" un "Publikācijas" saite ir 1 pret daudziem, jo vienam lietotājam var būt vairākas publikācijas, savukārt vienai publikācijai var būt tikai viens publicētājs (lietotājs).

# 3. DATU BĀZES IZSTRĀDE

## 3.1. Datu bāzes fiziskā struktūra

Datu bāzes izstrādei ir nepieciešama Datu bāzes fiziskā struktūra (att. 3.1.). Tā tika izveidota un normalizēta pēc 3NF, un pārveidojot entitijas par relācijām. Datu bāze sastāv no 43tabulām, kas satur informāciju par lietotājiem, publikācijām, sessiju.

Att. 3.1. Datu bāzes fiziskā struktūra

Tabulas “user” struktūra ir redzama 3.1. tabulā. Tā glabā informāciju par sistēmas lietotājiem: identifikatora kodu, e-pastu, paroli, vārdu, uzvārdu, lietotāja veidu, organizācijas vārdu un reģistrācijas numuru, adresi.

Tabulas “User” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr. | Lauka saturs | Lauka Nosaukums | Tips | Piezīme |
| 1. | Identifikācijas kods | UserID | int | Primāra atslēga |
| 2. | Vārds | Name | string |  |
| 3. | Uzvārds | Lastname | string |  |
| 4. | Parole | Password | string | Hashed |
| 5. | Organizācijas nosaukums | Org\_name | string | Tikai organizācijām |
| 6. | Lietotāja veids | User\_type | string | Privātpersona vai organizācija |
| 7. | Epasts | Email | string |  |
| 8. | Reģistrācijas NR. | Reg\_number | string | Tikai organizācijām |
| 9. | Adrese | location | string | Tikai organizācijām |

Tabulas “JobAdvert” struktūra ir redzama zemāk, tajā glabājas informācija par lietotāja publikācijām, un ir saistīta ar tabulu “users”.

Tabulas “JobAdvert” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr | Lauka saturs | Lauka Nosaukums | Tips | Piezīme |
| 1. | Identifikācijas kods | AdvertID | int | Primāra atslēga |
| 2. | Publikācijas nosauk. | Job\_title | string |  |
| 3. | Aparaksts | description | string |  |
| 4. | Organizācijas nosaukums | Company\_name | string |  |
| 5. | Telefona Nr. | Phone\_number | string |  |
| 6. | Reģistrācijas NR. | Reg\_number | string |  |
| 7. | Adrese | location | string |  |
| 8. | Epasts | email | String |  |
| 9. | Darba laika veids | worktype | string |  |
| 10. | Profesija | Profession | String |  |
| 11. | Darba veids | typeOfwork | String |  |
| 12. | Pisēta | City | string |  |
| 13. | Darbalaiks | Worktime | String |  |

Tabulas “Favorites” struktūra ir redzama zemāk, tajā glabājas informācija par lietotāju un publikācijām, kuras ir atzīmētas kā svarīgās. Tā ir saistīta gan ar tabulu “user”, gan ar   
“Jobcard”. Bet ir starptabula, kura palīdz atrast atzīmētas tabulas.

Tabulas “Favorites” struktūra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr. | Lauka saturs | Lauka Nosaukums | Tips | Piezīme |
| 1. | Identifikāciajs kods | \_ID | string | Primāra atslēga |
| 2. | Lietotāja epasts | user\_email | string |  |
| 3. | JobAdvert id | Job\_id | string |  |

## 3.2. Datu bāzes optimizēšana

Neskatoties uz to, ka datubāze nav liela un nepieprasa lielus resursus, tās optimizēšana ir svarīgs process, lai minimizēt resursu izmantošanu un izmaksas. Beigās datubāze tika pielāgota 3NF, lai to optimizētu.   
  
Tabula favorites, tika izveidota, lai neglabāt nepieciešamos datus tabulās “user” un Jobadvert, bet glabāt atsevišķi un patstāvīgi.

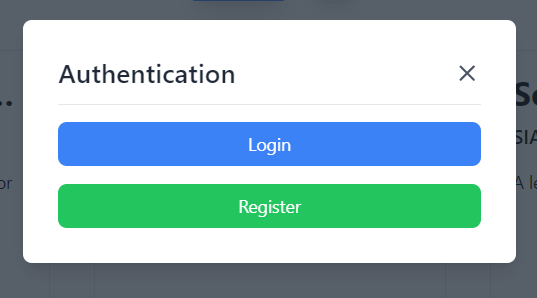
Visi atkarīgi dati tika piesavināti pie primāras atslēgas, novēršot dublešanos un atstājot brīvo atmiņas vietu citiem procesiem.

# 4. SISTĒMAS FUNKCIONĀLO PRASĪBU REALIZĒŠANA

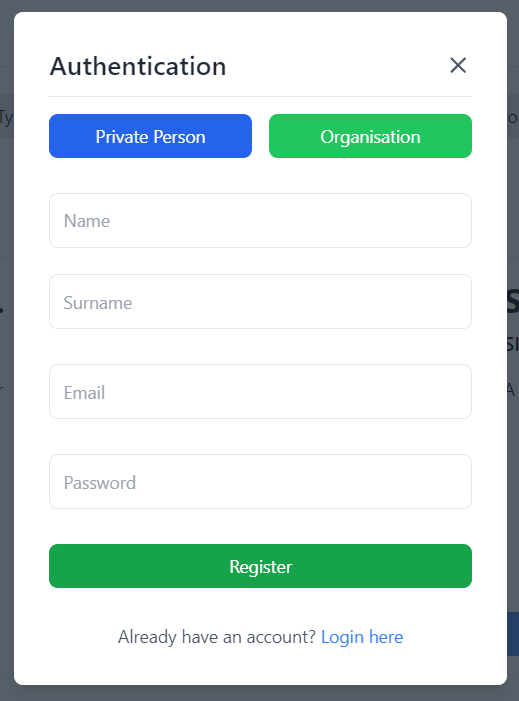
## 4.1 Datu pievienošana ,labošana, dzēšana

**Datu pievienošana:**

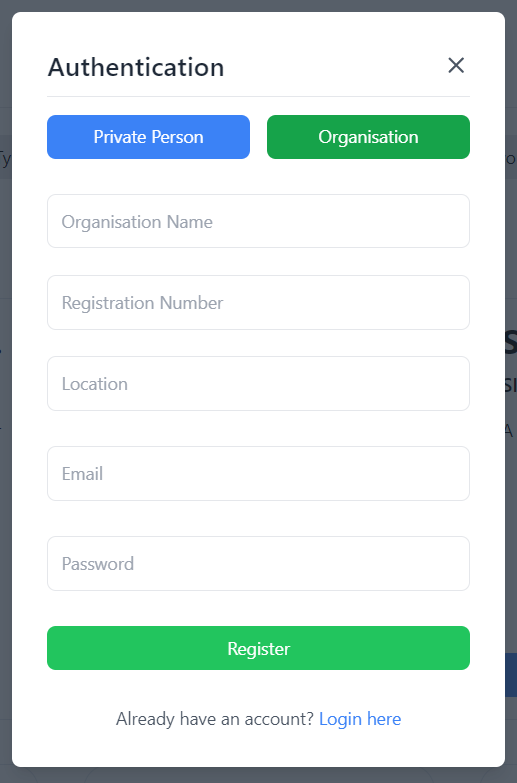
Jauna lietotāja pievienošanai var izmantot Register/Login logu. Tajā jāievada vārdu, uzvārdu, e-pastu, paroli ja tiek reģistrēta privātpersona, un epastu, paroli, organizācijas reģ.Nr un nosaukumu, un adresi ja tiek reģistrēta organizācija.



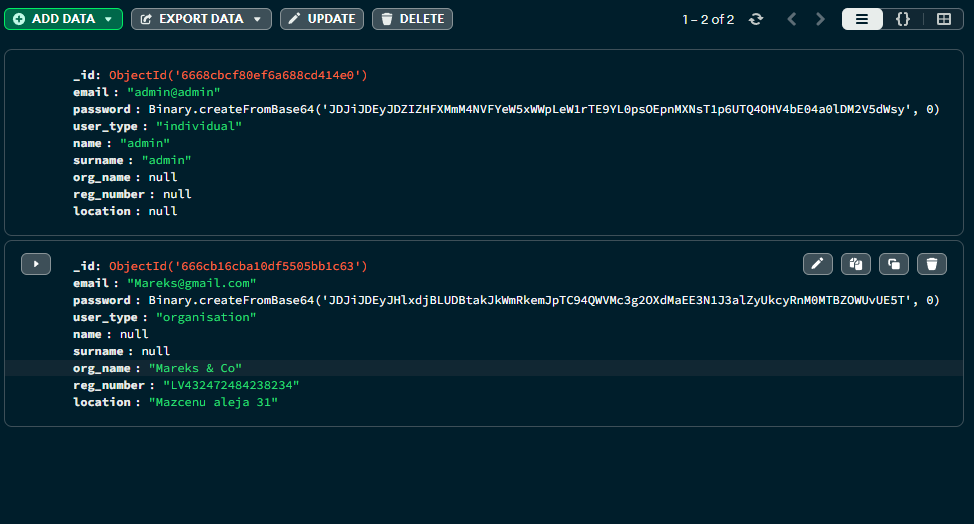
Att. 4.1. Reģistrācijas/Pieslēgšanas izvēles logs



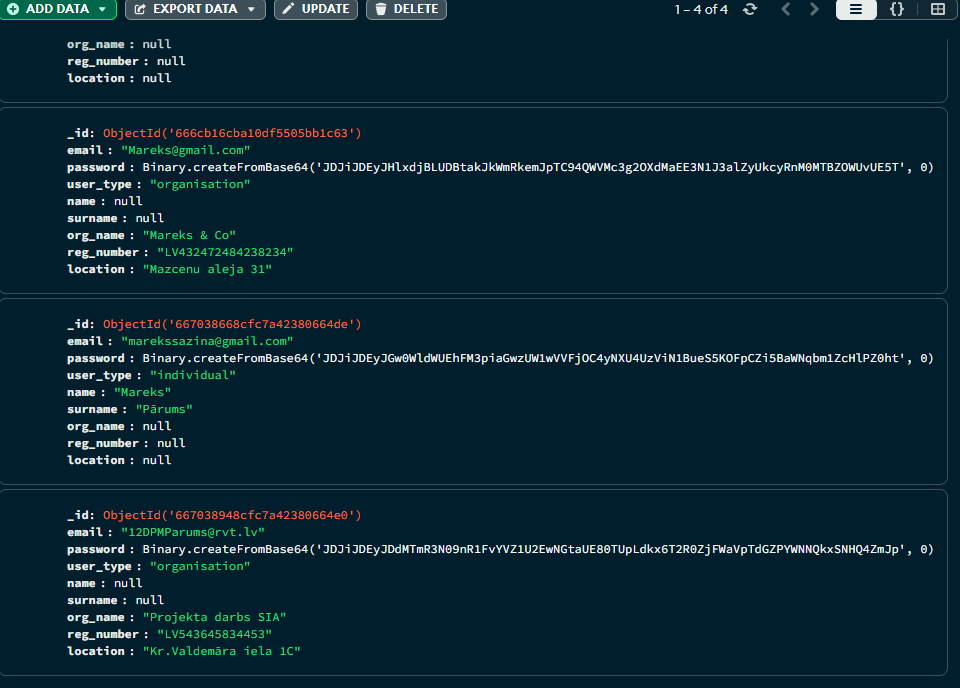
Att. 4.2. Reģistrācijas logs privātpersonām



Att. 4.3. Reģistrācijas logs organizācijām



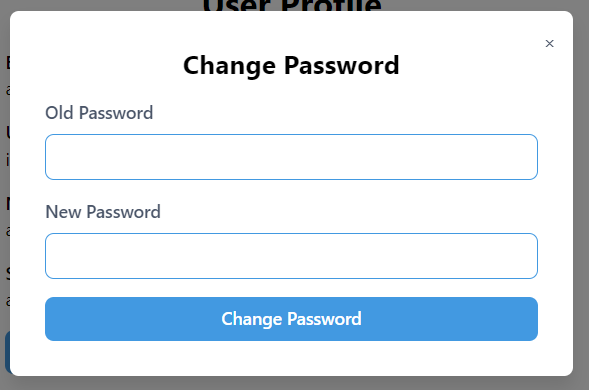
Att. 4.4. Datubāze pirms jaunā lietotāja pievienošanas



Att. 4.4. Datubāze pēc jaunu lietotāju pievienošanas (privātpersona Mareks un uzņēmums Projekta darbs SIA)

**Datu labošana:**

Lietotājs var mainīt paroli, izvēloties šadu funkciju Profila lapā:



Att. 4.5. Paroles maiņas logs

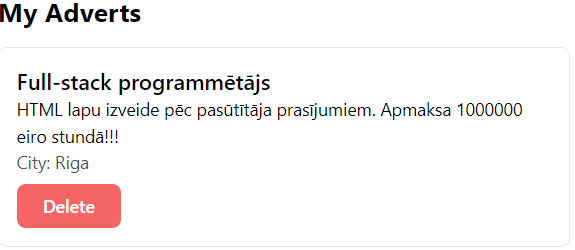


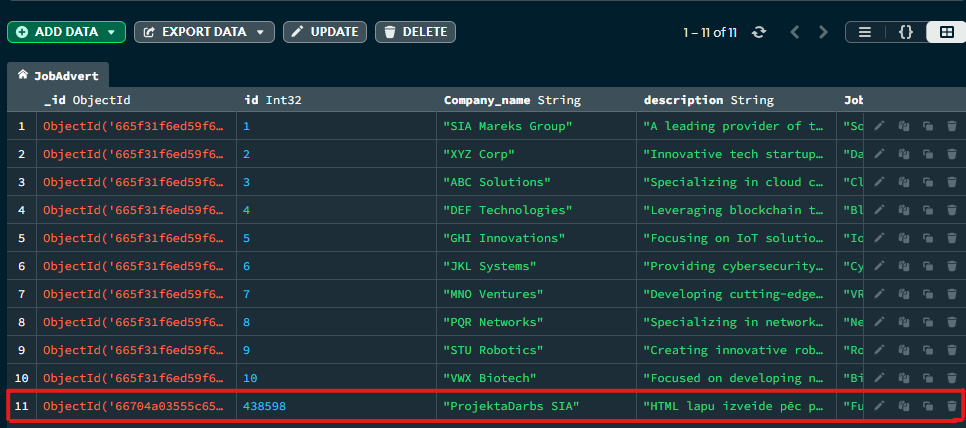
Att. 4.5. Heshēta parole pirms maiņas  
  


Att. 4.5. Heshēta parole pēc maiņas

**Datu dzēšana:**

Lietotājs, kas ir organizācija, var pievienot un dzēst darba sludinājumus, tam ir paredzēta speciāla poga dzēšanai.

  
Att. 4.5. Sludinājuma izskats ar pogu “delete” Profila sadaļā



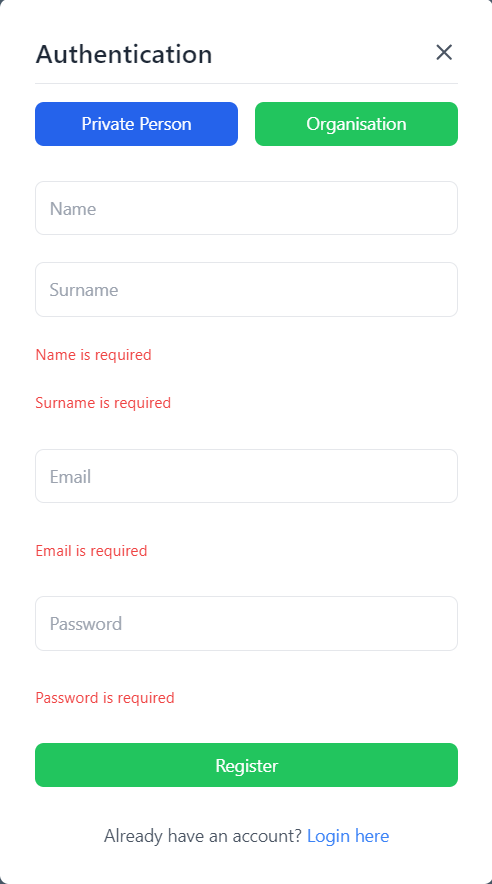
Att. 4.6. Datubāzes ieraksts un skaits pirms dzēšanas



Att. 4.7. Datubāzes ieraksts (nav) un skaits pēc dzēšanas

## 4.2 Datu validācija

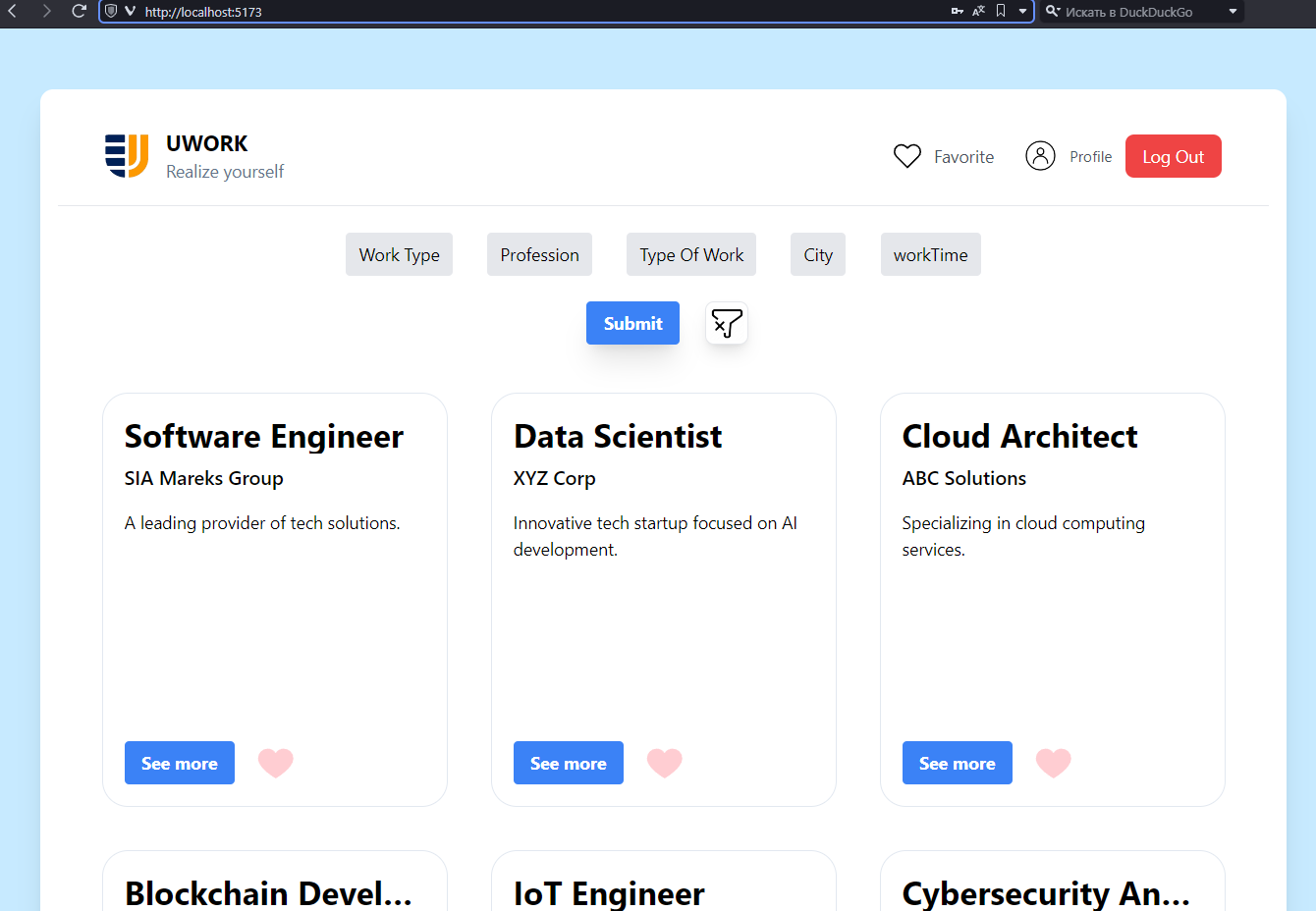
Lietotājam reģistrējoties, sistēma pārbauda, vai visi lauki ir ievadīti:



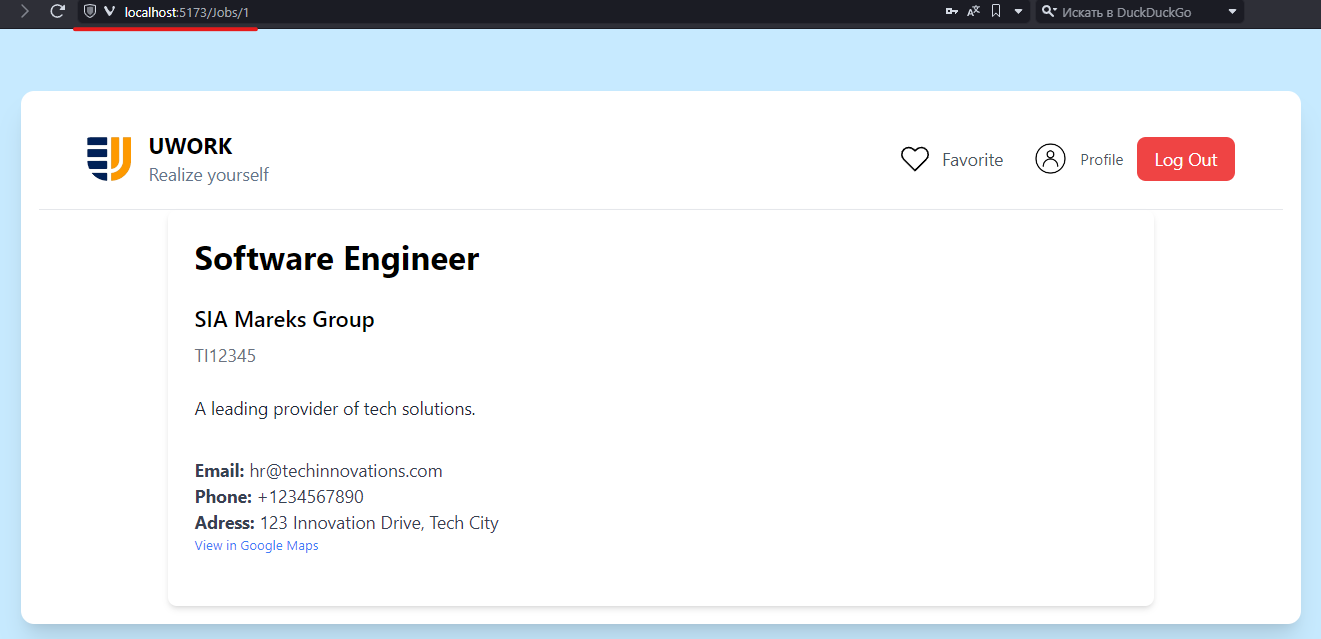
Att. 4.7. Datu pārbaude uz esamību

## 4.2.1. Datu izgūšana

Datu izgūšanas piemērs sistēmā ir izplesta darba sludinājuma apskate. Kad lietotājs izvēlas sludinājumu un nospied “see more”, lietotājs tiek novests uz citu lappusi, ar nepieciešamā sludinājuma id (piemēram: /Jobs/1), kurā ir redzams pilns sludinājums.



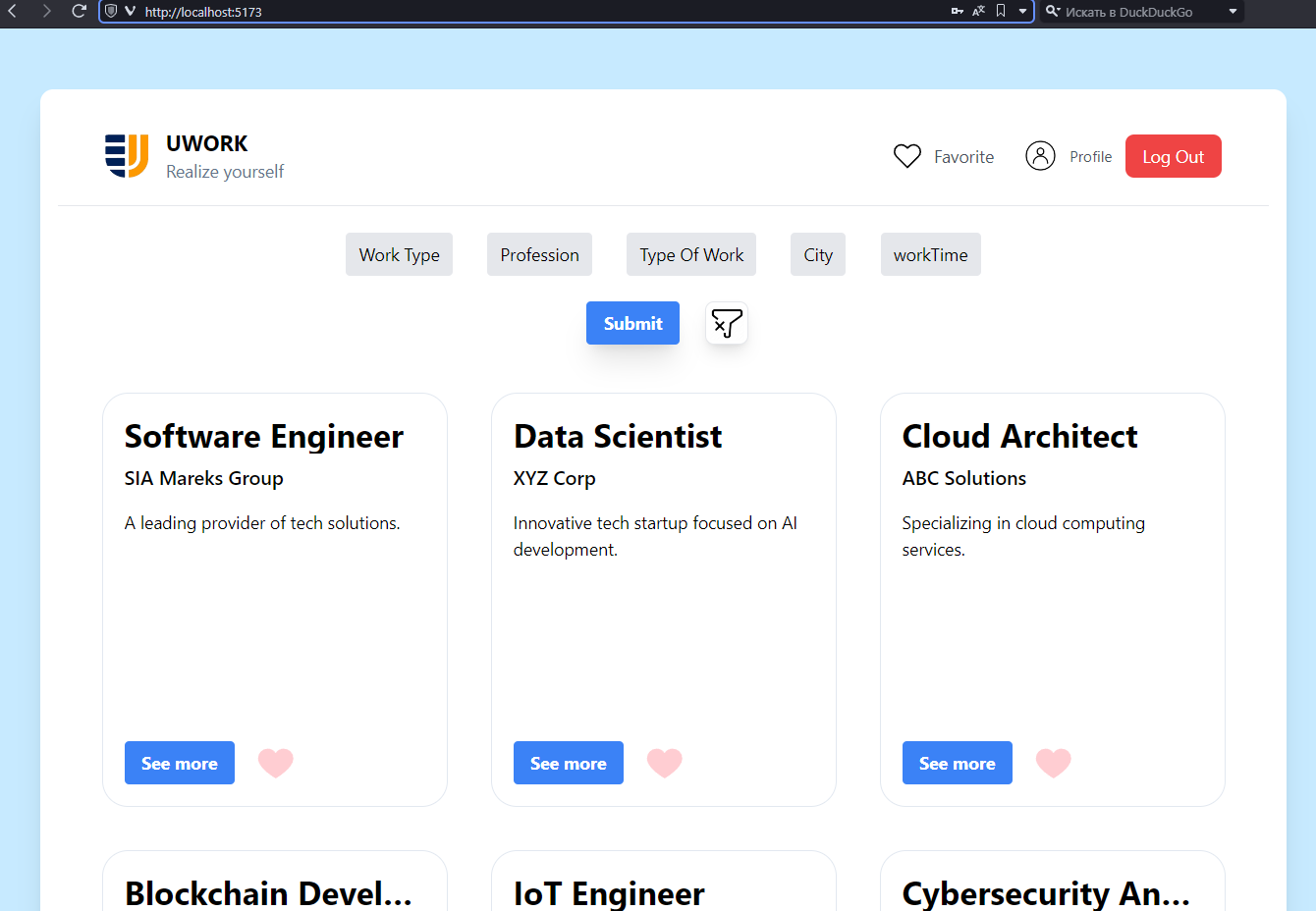
Att. 4.8. Mājaslapa ar visiem sludinājumiem



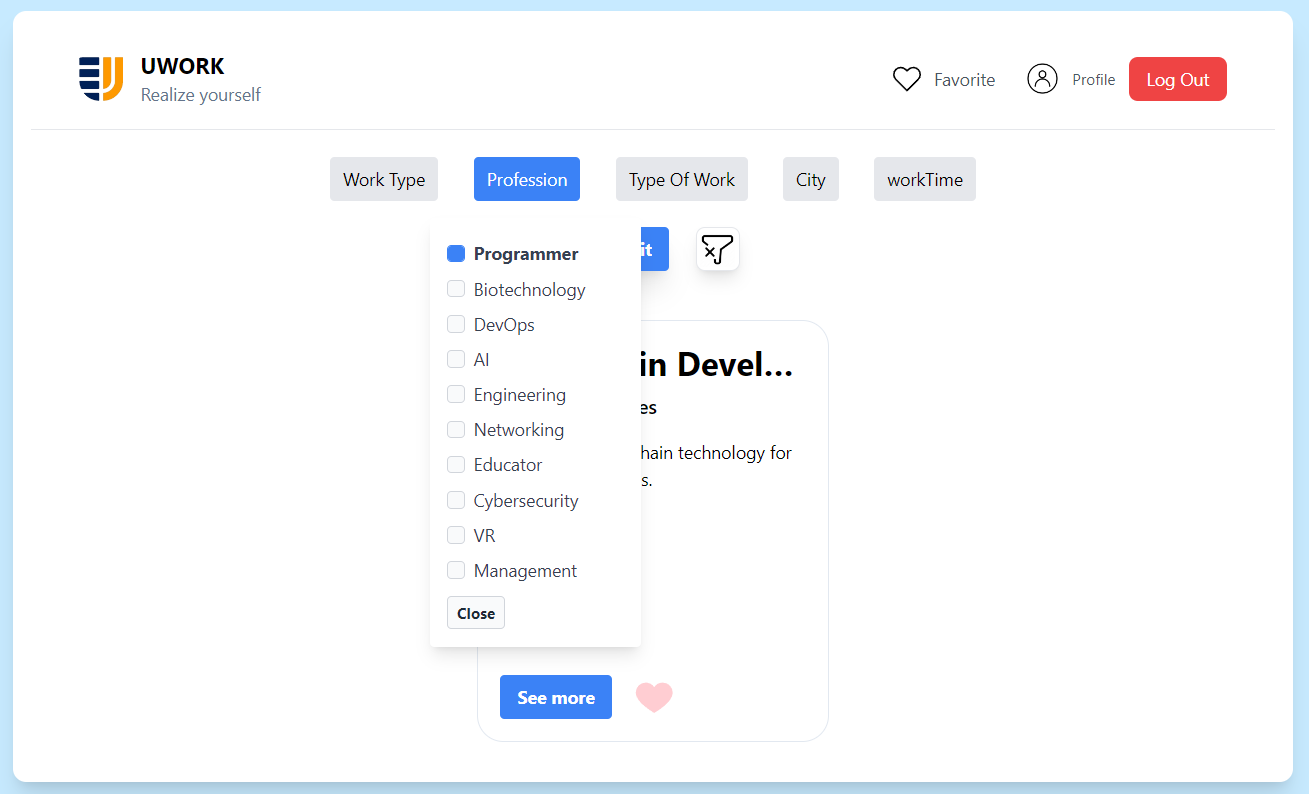
Att. 4.9. Lapa ar noteikto sludinājuma ID.

## 4.2.2 Datu apstrāde

Datu apstrādes piemērs ir meklēšana pēc filtriem mājaslapā. Lietotājs var izvēlēties filtrus, kurus vēlās pielietot vai noņemt.



Att. 4.10. Mājaslapa ar visiem sludinājumiem



Att. 4.10. Mājaslapa ar sludinājumiem, kas atrasti ar filtru “professija-programmētājs”

# 5. Nobeigums

Šis projekts ir tikai iesākums un uzmetums pilnam darba meklēšanas projektam. Darba laikā tika izveidota pati mājaslapa ar datu apstrādes funkcijām un datubāze, kas atbilst visām prasībām.

Darba laikā tika izpētīta lietotāju mijiedarbība ar sistēmu, lai veidot to ērtāku un optimizētāku, kā arī datubāzes izveidei nepieciešamās tabulas un to saistības. Ir aprasktīta sistēmas specifikācija, nepieciešamais funkcionāls.  
  
Tika izveidota dokumentācija, kas apraksta visus izveides posmus, kā arī apraksta datu bāzes struktūru, funkcijas, dati, kurus tās pielieto. Tika aprakstīti sistēmas izveides rīki un tehnoloģijas un to izvēles pamatojums. Dokumentācijā ir parādītas visas funkcijas – datu pievienošana, apstrāde, dzēšana, labošana validācija izgūšana, kuras ir realizētas kopā ar attēliem, kas labāk parāda kā funkcijasir realizētas.