

Liepājas Valsts tehnikums

Problēmpietiekumu sistēmas izstrāde

Liepājas Valsts tehnikumā

Kvalifikācijas eksāmena dokumentācija

Profesionālā kvalifikācija Programmēšanas tehniķis

Grupas nosaukums 4PT

Projekta izstrādātājs Daniels Vidopskis

Eksāmena datums 2023. gada \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Liepāja 2023

# Satura radītājs

[Satura radītājs 3](#_heading=h.j5u6svba4029)

[Ievads 4](#_heading=h.a4ealwwmff3b)

[1. Uzdevuma formulējums 5](#_heading=h.30j0zll)

[2. Programmatūras prasību specifikācija 6](#_heading=h.1fob9te)

[2.1. Produkta perspektīva 6](#_heading=h.aywilutk98h9)

[2.2. Sistēmas funkcionālās prasības 6](#_heading=h.j3rxyzuiapqx)

[2.2.1. Apmeklē mājaslapu 6](#_heading=h.vp4pzj4b0su3)

[2.2.2. Microsoft ielogošanās 7](#_heading=h.3wghs5gpl5k7)

[2.2.3. Sākumlapas 8](#_heading=h.ajgu58ygsggw)

[2.2.4. Pievienot pieteikumu 8](#_heading=h.l8xbpb5kx3so)

[2.2.5. Pieteikuma piešķiršana 9](#_heading=h.wopr6h7rgknb)

[2.3. Sistēmas nefunkcionālās prasības 10](#_heading=h.uj9z88mx2kz)

[2.3.1. Valoda 10](#_heading=h.nga7i8kysbgy)

[2.3.2. Saderība ar pārlūkprogrammām 10](#_heading=h.ljeg7gz7hnbe)

[2.3.3. Drošība 10](#_heading=h.uxl48pylwd8e)

[2. Saderība 10](#_heading=h.b5j99q6r8n6f)

[2.1. Gala lietotāja raksturiezīmes 11](#_heading=h.jsmwme1f8v7u)

[2.2. Lietoto terminu un saīsinājumu skaidrojumi 12](#_heading=h.5brf766z4vqn)

[3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums 13](#_heading=h.k8yjs99hqdo)

[3.1. Izvēlēto risinājuma līdzekļu un valodu apraksts 13](#_heading=h.rimpjvvzt2wy)

[3.2. Iespējamo (alternatīvo) risinājuma līdzekļu un valodu apraksts 13](#_heading=h.ix6u37u7ns8z)

[4. Sistēmas modelēšana un projektēšana 14](#_heading=h.c6eaj19eebj0)

[4.1. Sistēmas struktūras modelis 14](#_heading=h.p8usgtupvmzt)

[4.2. Klašu diagramma / ER diagramma 14](#_heading=h.hu23ber2rbdw)

[4.3. Funkcionālais un dinamiskais sistēmas modelis 15](#_heading=h.5xcr7v8z9opf)

[4.4. Aktivitāšu diagramma (Activity) 16](#_heading=h.n8aq10hl1kcc)

[4.5. Lietojumgadījumu diagramma (Use Case) 17](#_heading=h.qn6d7cz7vu0i)

[4.6. Sistēmas moduļu apraksts un algoritmu shēmas 18](#_heading=h.2u3p1nlqo87i)

[5. Lietotāju ceļvedis 19](#_heading=h.q5meuc7payvc)

[6. Testēšanas dokumentācija 20](#_heading=h.c6zmw0fuht28)

[6.1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums 20](#_heading=h.tiqpr7d8w78c)

[6.2. Testpiemēru kopa 20](#_heading=h.q5qrp4lwfc28)

[6.3. Testēšanas žurnāls 20](#_heading=h.fts98hvnlyst)

[7. Individuālais ieguldījums 20](#_heading=h.dqgo9i4pg5m5)

[8. Secinājumi 20](#_heading=h.ncue3xiqwjfb)

[9. Lietoto saīsinājumu saraksts 20](#_heading=h.1oqivaolbmcy)

[10. Literatūras un informācijas avotu saraksts 20](#_heading=h.bxk5u5vhlios)

# Ievads

Prakses pienākumus es veicu Liepājas Valsts tehnikumā. Tā kā šī mācību iestāde atrodas divās atsevišķās ēkās - Ventspils ielā un Vānes ielā-, tad bieži mēdz būt problēmas ar informācijas apmaiņu starp skolas ēkām. Īpaši tad, kad jāpiesaka dažādas tenniskas problēmas un to risinājumi. Šo problēmu atrisināt varētu izstrādāta problēmpieteikumu sistēma, kurā vienkārši un operatīvi būtu iespējams pieteikt problēmu, kā arī sekot līdzi problēmas risinājuma statusam. Tāpēc tika strādāts pie šādas sistēmas izveides..

Tehniskā dokumnetācija ir viens no svarīgākajiem dokumentiem kāda projekta izstrādes procesā. Šie dokumenti ietver sevī arī informāciju par pašu “LVT Ticket Support” sistēmu. Problēmpieteikumu atbalsta sistēma ir vietne, kas nodrošina atbalstu skolas pieteikuma problēmu risināšanā LVT skolotājiem. Vietnes mērķis ir palīdzēt skolotājiem un darbiniekiem atrisināt jebkādas problēmas, kas saistītas ar problēmu pieteikumiem, piemēram, neizdevušo savienojumu ar internetu vai salauztu tehniku. Šī iemesla dēļ būtu svarīgi, lai ir izvēle, kurai nodaļai pieteikums paredzēts - IT nodaļai vai saimnieciskajam dienestam. Vēl būtu nozīmīgi, lai problēmas pieteicējs varētu reizē ar problēmas pieteikumu sniegt arī savu komentāru, tas atvieglotu problēmas risinājumu.

Sistēma izstrādāta tā, lai problēmas risinātājs redz problēmas

# 

# Uzdevuma formulējums

“LVT Ticket Support” ir mājas lapa, kura atrodas uz apache2 server, tās galvenā funkcija ir izveidot pieteikumu, kurā var ievadīt skolas atrašanās vietu (ielu), telpu, problēmu, piezīmi, komnetārus, kā arī izvēli, kurai nodaļai adresēts pieteikums - IT/Saimniecības nodaļu. Pēc tam šī informācija tiek rādīta noteiktai nodaļai, kurā varēs atzīmēt, vai konkrētais pieteikums ir izpildīts vai vēl ir procesā. Mājas lapas administrācija varēs redzēt visus pieteikumums, kā arī to, cik daudz pieteikumu ir izpildīti.

Mājaslapa parāda skolas darbiniekiem, vai viņu pieteikums tiek izpildīts, kā arī tajā var pievienot savu pieteikumu. Mājaslapā pieteikumus var pieteikt gan IT, gan samniecības nodaļai, kurā var norādīt problēmu, ielu, telpu un piezīmi par risināmo problēmu. Arī steidzamos gadījumos šeit var veikt atzīmi par to, kas atgadījies un nepieciešams steidzams risinājums.

Skolotājiem ir savs ‘skats’, tas nav saistīts ar IT un samniecības nodaļas darbiniekiem, un viņi nevar redzēt citu skatu, kas viņiem nav piesaistīts.

Uzdevums - izmantojot skolas krāsas (zils, balts, sarkani oranžs) jāizveido mājas lapa, kura ir viegli pārskatāma un saprotama.

# Programmatūras prasību specifikācija

## Produkta perspektīva

Produkta perspektīva ir dot iespēju skolas dabiniekiem viegli un ātri pieteikt savu problēmu. Kad sistēma ir veiksmīgi uzstādīta, tad skolas administrācija, kas pārskata visu sistēmu var deleģēt, kuriem ir piekļuve administrācijas panelim, kur var pārvaldīt un pārskatīt visu sistēmu. Skolas skolotāji var izveidot ātru un vieglu pieteikumu par notikušo problēmu un redzēt, kad problēma ir atrisināta.

## Sistēmas funkcionālās prasības

Sistēma jeb mājaslapa tiek iedalīta trīs apmeklētāju tipos - skolotājs, administrators, darbinieks. Skolotājs ir skolas skolotājs, kurš var izveidot savu pieteikumu par informāciju, kur un kas notika, bet darbinieks ir samniecības nodaļas darbinieks, kurš izpildīs skolotāja izveidoto pieteikumu, kuru varēs atzīmēt izpildītu vai procesā, bet administrators ir sistēmas pārvaldnieks, kurš tiek klāt pie visām sistēmas funkcijām un visu pārskata.

### Apmeklē mājaslapu

Mērķis:

* Mājaslapas adreses ievadīšana jebkurā interneta pārlūkprogrammā.

Apstrāde:

* Tiek pārbaudīta, vai ir pareizi uzstādīta sistēma uz serveriem.

Izvades dati:

1. Tiek veiksmīgi ielādēta mājaslapas ielogošanās lapa, kurā tiek prasīts skolas e -pasts un parole.
2. Kļūdas paziņojums, ka vietne nav sasniedzama.

## Microsoft ielogošanās

1. tabula

**Microsoft prasības**

| **Nosaukums** | **Obligāts** | **Piezīme** |
| --- | --- | --- |
| Skolas e-pasts | Jā | Ieeja tikai ar skolotāju e-pastiem, kuriem nav “sk” pēc @ simbola |
| Parole | Jā |  |

Mērķis:

* Neļaut ieiet mājaslapā skolēniem un citu skolu pedagogiem.

Ievades dati:

1. Ierakstot skolas e-pastu “email” laukumā un paroli “password” laukumā.

Apstrāde:

* Pēc pogas “login” uzspiešanas dati tiek aizsūtīti uz “Microsoft”, kur tie tiek

pārbaudīti un verificēti, vai patiešām ir pareizi ievadīti.

Izvades dati:

1. Pēc veiksmīgas datu aizsūtīšanas un verifikācijas aizved lietotāju uz paredzēto sākumlapu.
2. Pēc neveiksmīgas datu aizsūtīšanas vai verifikācijas parāda lietotājam, vai parole vai e-pasts ir pareizi ievadīts.

## Sākumlapas

Mērķis:

* Pēc ieiešanas mājaslapā redzama sākumlapa, kurā redzams skolotāja izveidotais pieteikums/ pieteikumi, kuru pie vajadzības var arī atcelt.

Ievades dati:

-

Apstrāde:

* Saņem pieteikuma datus no datubāzes, kuri ir veidoti no tā lietotāja datiem, kurš ir iegājis mājaslapā, un tie tiek sakārto no jaunākā uz vecāko pieteikumu.

Izvades dati:

1. Tiek veiksmīgi saņemti dati no datubāzes, parāda pieteikumus no jaunākā uz vecāko.
2. Tabula ir tukša, ja nav izveidots neviens pieteikums.

## Pievienot pieteikumu

Mērķis:

* Pievienot skolotāja pieteikumu datubāzē, kurus varēs redzēt gan dabinieki,

gan administrācija, gan skolotājs. Būs redzam informāja, kurā skolas ēkā, kurā,telpā problēma risināma, kā arī as, vai pieteikums ir adresēts IT vai saimniecības nodaļai.

Ievades dati:

* Izvēlēties Vānes ielas vai Ventspils ielas ēku, norādīt telpu, izvēlēties, vai IT, vai saimniecības nodaļu, ierakstīt problēmu un ierakstīt piezīmi (tā nav obligāta).

Apstrāde:

* Saņem pieteikuma datus no datubāzes, kuri ir veidoti no tā lietotāja, kurš iegājis mājaslapā, un sakārto no jaunākā uz vecāko pieteikumu.

Izvades dati:

1. Aizved uz sākumlapu, kurā būs redzami vecie izveidotie problēmpieteikumi un jaunākie, kuri tikko tika izveidoti.
2. Parāda kļūdu, kurā ir teikts, ka viens vai vairāki obligātie lauki nav izpildīti.

## Pieteikuma piešķiršana

Mērķis:

Sistēmai automātiski ir jāpiešķir iesniegtās biļetes atbilstošam personālam vai nodaļām, lai zināt kurām tas pieder

Ievades dati:

* Izvēlēties Vānes ielas vai Ventspils ielas ēku, norādīt telpu, izvēlēties, vai IT, vai saimniecības nodaļu, ierakstīt problēmu un ierakstīt piezīmi (tā nav obligāta).

Apstrāde:

* Pieteikumam pievieno unikālu atslēgu pie lietotāja ēpasta

Izvades dati:

* Nav izvades dati, viss notiek majaslapas aizmugurē, jeb backend

# Sistēmas nefunkcionālās prasības

## Valoda

Mājaslapa tika izveidota latviešu valodas lietotājiem.

## Saderība ar pārlūkprogrammām

Mājaslapa un administrācijas pārvaldības paneļa sistēmas funkcionalitātei jālieto populārākās pārlūkprogrammas, kā piemēram, “Mozila Firefox”, “Opera”, “Opera GX”, “Google Chrome”, “Microsoft Edge”.

## Drošība

Sistēmai jānodrošina lietotāju datu drošība un konfidencialitāte, tostarp lietotāju pieteikšanās datus un personīgo informāciju. Tā jāizmanto atbilstošas drošības pasākumi, piemēram, datu šifrēšana, drošas savienojuma protokols (HTTPS) un piekļuves kontroles, lai pasargātu pret neautorizētu piekļuvi un datu noplūdēm.

## Saderība

Sistēmai jābūt saderīgai ar dažādiem interneta pārlūkiem, operētājsistēmām un ierīcēm, kas parasti tiek izmantotas skolu personāla vidē. Tā jābūt atsaucīgai un nodrošināt vienotu lietotāju pieredzi dažādos platformos.

## Gala lietotāja raksturiezīmes

Skolā ir dažādi vecuma skolotāji, kuri lietos mājaslapu, izveidojot savus problēmpieteikumus.

## Lietoto terminu un saīsinājumu skaidrojumi

2. tabula

**Terminu skaidrojums**

| Termins un saīsinājums | Skaidrojums |
| --- | --- |
| DB | Datubāze jeb DB ir informācijas kopums ar noteiktu struktūru. Par datubāzi var saukt jebkuru strukturētas informācijas kopumu. |
| **Serveris** | Serveris ir datorsistēma klienta servera arhitektūrā, kas nodrošina klientu pieprasījumu apstrādi un nosūta tiem atbildes. Terminu "serveris" attiecina gan uz datoru, gan uz programmatūru, kas veic šādas funkcijas. |
| Mājaslapa | "Mājaslapa" latviešu valodā ir interneta vietne vai tīmekļa lapa, kas pieejama internetā un satur informāciju, resursus, pakalpojumus vai citu saturu, ko var aplūkot, lasīt vai izmantot interneta lietotāji. |
| Frontend | Frontend, jeb arī priekšgals, ir tā daļa no informācijas tehnoloģijām, kas attiecas uz lietotāja saskarnes izstrādi un darbību vietnē vai aplikācijā. Priekšgals ir atbildīgs par lietotāja pieredzes (User Experience, UX) veidošanu un ieviešanu, tostarp dizainu, interaktīvo elementu izstrādi, lietotāja ievades apstrādi un vizuālo prezentāciju. |
| BackEnd | Backend, jeb arī aizgals, ir tā daļa no informācijas tehnoloģijām, kas atrodas aiz priekšgala, un nodrošina servera pusi, kas atbild par datu apstrādi, datu bāzu savienošanu, biznesa loģikas implementāciju un citas sistēmas darbības, kas notiek aiz aizkulisēm. Aizgals nodrošina informācijas apstrādi un datu pārvaldību, kas ir nepieciešama, lai aplikācija vai vietne darbotos pareizi un efektīvi. |

# Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums

## Izvēlēto risinājuma līdzekļu un valodu apraksts

HTML:

Ir dažas alternatīvas valodas HTML, piemēram, Markdown, YAML un XML.

Markdown ir viegla un lasāma marķēšanas valoda, kas paredzēta teksta formatēšanai tīmekļa vietnēs un dokumentos. Tā piedāvā vienkāršas sintakses konstrukcijas, kas ļauj veidot formatētu tekstu bez sarežģītām HTML etiķetēm. Markdown ir populārs starp blogeriem, programmētājiem un citiem, kas vēlas viegli formatēt tekstu, bet nevēlas lietot HTML.

PHP:

Es izvēlējos izmantot PHP vietā citiem alternatīviem, jo tā bija vispiemērotākā valoda šim konkrētajam projektam.

Pieredze: man pašam ir nedaudz pieredze ar PHP valodu. Plaša atbalsta kopiena: PHP ir viena no populārākajām serverpusēm izpildāmajām valodām, un tā ir plaši izmantota visā pasaulē. Tas nozīmē, ka ir pieejami lieliski resursi, kas atvieglo izstrādi un problēmu risināšanu.

Integrācija ar datu bāzēm: PHP piedāvā lielisku integrāciju ar dažādām datu bāzēm, piemēram, MySQL un PostgreSQL. PHP ir platformas neatkarīga valoda, kas nozīmē, ka es varu to izmantot dažādos operētājsistēmās un serveru vidēs, piemēram, Linux, Windows vai macOS, bez ierobežojumiem.

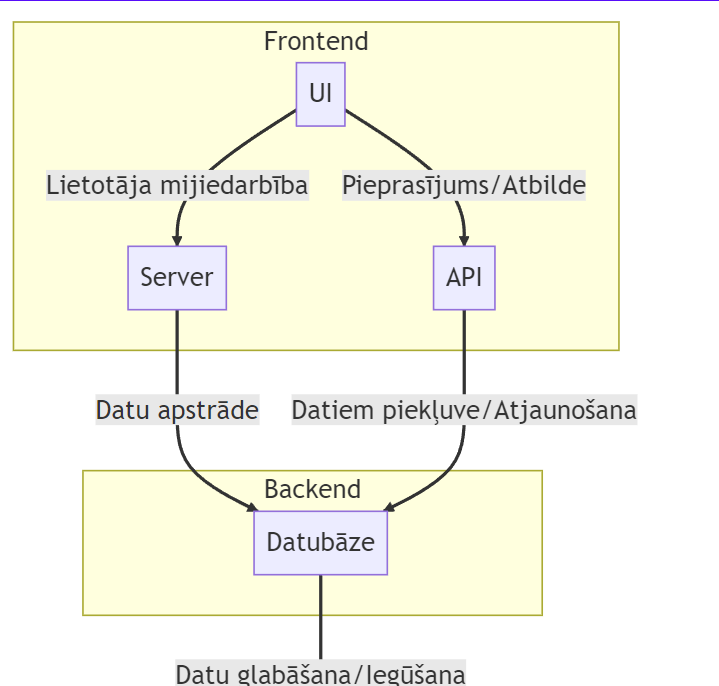
CSS (piemēram, Sass vai Less):

CSS preprocesori ir valodas, kas izmanto priekšprocesoru pielietošanai CSS stilu lapām. Tie ļauj rakstīt CSS kodu, kas ir atjaunināms, atkārtoti izmantojams un strukturizēts veids. Preprocesori piedāvā vairākas priekšrocības

## Iespējamo (alternatīvo) risinājuma līdzekļu un valodu apraksts

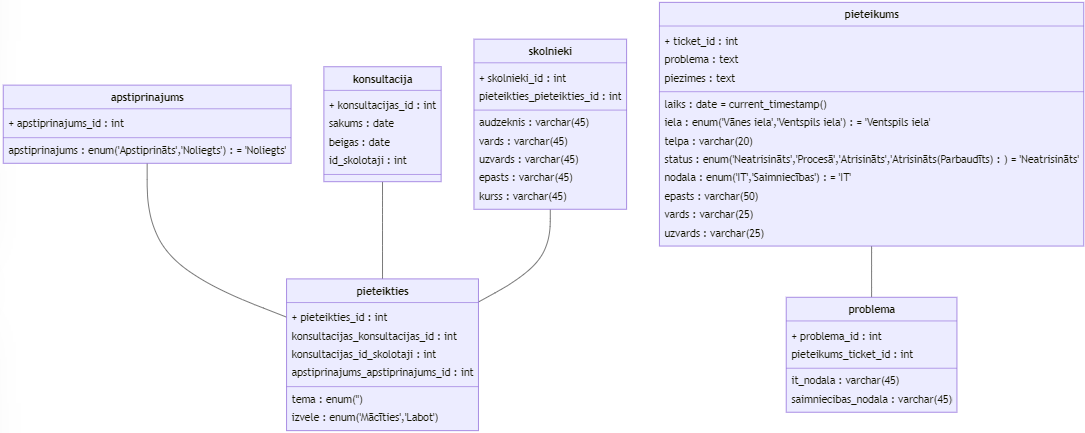
# Sistēmas modelēšana un projektēšana

## Sistēmas struktūras modelis



**1.attēls. struktūras attēls**

## Klašu diagramma / ER diagramma



**2.attēls. ER attēls**

## Funkcionālais un dinamiskais sistēmas modelis

## 

**3.attēls. dinamiskais sistēmas modeļa attēls**

## Aktivitāšu diagramma (Activity)

## 

**4.attēls. Aktivitāšu diagramma attēls**

## 

## Lietojumgadījumu diagramma (Use Case)

## 

**4.attēls. Lietojumgadījumu diagramma attēls**

## Sistēmas moduļu apraksts un algoritmu shēmas

Sistēmas moduļa apraksts:

Sistēmas modulis ir daļa no lielākas skolas tīmekļa lapas lietotnes, kas ļauj skolēniem un skolotājiem mijiedarboties un pārvaldīt dažādus skolas aspektus, piemēram, iesniegt problēmu ziņojumus IT vai administratīvām problēmām, un pārvaldīt pieteikuma informāciju. Sistēmas modulis ir atbildīgs par šo funkcionalitāšu aizmugurējās daļas loģiku un datu pārvaldību.

Algoritma shēma:

1. Inicializē sistēmas moduli, pieslēdzoties skolas tīmekļa lapas datu bāzei un iestatot nepieciešamos SQL režīmus un laika joslu.
2. Definē datu struktūras datu bāzes tabulām, piemēram, 'apstiprinajums', 'pieteikums', 'problema' un 'skolnieks'. Šīs datu struktūras ietver kolonnas, datu tipus un ierobežojumus katrai tabulai.
3. Īsteno funkcijas, kas apstrādā CRUD (izveidot, lasīt, atjaunināt, dzēst) darbības katrai tabulai, piemēram, ievietot jaunus ierakstus, izgūt ierakstus, atjaunināt ierakstus un dzēst ierakstus, izmantojot SQL vaicājumus.
4. Īsteno funkcijas, kas apstrādā konkrētas funkcionalitātes, piemēram, apstrādā studentu konsultāciju pieprasījumus, apstrādā studentu iesniegtos problēmu ziņojumus, pārvalda apstiprinājumus konsultāciju pieprasījumiem un atjaunina problēmu ziņojumu statusus.
5. Īsteno funkcijas, kas nodrošina datu validāciju un kļūdu apstrādi, piemēram, pārbauda derīgus ievades vērtības, apstrādā datu bāzes kļūdas un nodrošina atbilstošus kļūdu ziņojumus lietotājiem.
6. Īsteno funkcijas, kas nodrošina biznesa loģiku, piemēram, pārbauda konfliktus konsultāciju pieprasījumos, aprēķina datumus konsultācijām un pār

# Lietotāju ceļvedis

# 

# Testēšanas dokumentācija

## Izvēlētās testēšanas metodes, rīku apraksts un pamatojums

## Testpiemēru kopa

## Testēšanas žurnāls

# Individuālais ieguldījums

# Secinājumi

# Lietoto saīsinājumu saraksts

# Literatūras un informācijas avotu saraksts

## 