

Liepājas Valsts tehnikums

**Internetveikals “Sparkly Dream”**

Kvalifikācijas eksāmena praktiskās daļas tehniskā dokumentācija

Izglītības programma **33484011 Programmēšana**

Personālā kvalifikācija **Programmēšanas tehniķis**

Darba autors(-e): ………………………………………………

/vārds, uzvārds, paraksts/

Eksāmena datums 202 \_\_.gada \_\_.jūnijs

Liepāja 202\_\_

# Saturs

[Ievads 5](#_Toc187598991)

[1. Uzdevuma formulējums 6](#_Toc187598992)

[2. Programmatūras prasību specifikācija 8](#_Toc187598993)

[2.1. Produkta perspektīva 8](#_Toc187598994)

[2.2. Sistēmas funkcionālās prasības 8](#_Toc187598995)

[2.3. Sistēmas nefunkcionālās prasības 19](#_Toc187599024)

[2.4. Gala lietotāja raksturiezīmes 19](#_Toc187599025)

[3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums 20](#_Toc187599026)

[3.1. Izvēlēto risinājuma līdzekļu un valodu apraksts 20](#_Toc187599027)

[3.2. Iespējamo (alternatīvo) risinājuma līdzekļu un valodu apraksts 2](#_Toc187599028)2

[4. Sistēmas modelēšana un projektēšana 2](#_Toc187599029)4

[4.1. Sistēmas struktūras modelis 2](#_Toc187599030)4

[4.1.1. Izvietojuma (Deployment) diagramma / komponentu (Component) diagramma 2](#_Toc187599031)4

[4.1.2. Klašu (Class) diagramma / ER (Entity Relationship) diagramma 2](#_Toc187599032)4

[4.2. Funkcionālais un dinamiskais sistēmas modelis 2](#_Toc187599033)4

[4.2.1. Aktivitāšu (Activity) diagramma 2](#_Toc187599034)4

[4.2.2. Lietojumgadījumu (Use Case) diagramma 2](#_Toc187599035)4

[4.2.3. Stāvokļu (State) diagramma 2](#_Toc187599036)4

[4.3. Datu struktūru apraksts 2](#_Toc187599037)4

[5. Lietotāju ceļvedis 2](#_Toc187599038)5

[6. Testēšanas dokumentācija 2](#_Toc187599039)6

[6.1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums 2](#_Toc187599040)6

[6.2. Alternatīvās testēšanas metodes un rīki 2](#_Toc187599041)6

[6.3. Testpiemēru kopa 2](#_Toc187599042)6

[6.4. Testēšanas žurnāls 2](#_Toc187599043)6

[Secinājumi 2](#_Toc187599044)7

[7. Lietoto saīsinājumu un terminu skaidrojums 2](#_Toc187599045)8

[Literatūras un informācijas avotu saraksts 2](#_Toc187599046)9

[Pielikumi 30](#_Toc187599047)

# Ievads

Šis projekts ir paredzēts unikāla internetveikala "Sparkly Dream" izstrādei, kura ideja balstīta darba autora ģimenes uzņēmuma pieredzē. Uzņēmums ir izveidojies kā neliels, bet radošs darbnīcas projekts, kas specializējas unikālu Ziemassvētku eglīšu rotājumu izstrādē. Roku darbs ir uzņēmuma pamatvērtība, piešķirot rotājumiem īpašu izskatu un augstu kvalitāti. Līdz šim uzņēmuma produkcija tika piedāvāta tikai lokāli, taču pieaugošā interese par personalizētiem un augstvērtīgiem rotājumiem ir veicinājusi nepieciešamību paplašināt darbību.

Dokumentā ir aprakstītas šādas sadaļas: uzdevuma formulējums, programmatūras prasību specifikācija, izstrādes līdzekļu un rīku apraksts ar izvēles pamatojumu, sistēmas modelēšana un projektēšana, lietotāju ceļvedis, testēšanas dokumentācija, secinājumi, lietoto saīsinājumu un terminu skaidrojums, kā arī literatūras un informācijas avotu saraksts.

“Sparkly Dream” ir vairāk nekā tikai veikals – tā ir vieta, kur ikviens var radīt kaut ko īpašu un unikālu saviem svētkiem. Viena no svarīgākajām platformas funkcijām ir iespēja klientiem pielāgot savus rotājumus pēc savas gaumes un vēlmēm. Klienti var izvēlēties pamatkrāsu, kas veidos rotājuma galveno toni, kā arī pielāgot rotājuma malas figūru – ārējo formu vai dekoratīvu akcentu. Šīs figūras var būt, piemēram, eņģeļi, putni vai citi simboli, kas piešķir rotājumam individuālu raksturu. Ka arī klienti var izvēlēties arī rotājuma formu, kas noteiks, kādā stilā vai veidā rotājums izskatīsies – apaļa, medaljona, lāsteka vai pat lietussargs, lai atbilstu konkrētajam svētku tematam. Šī pielāgošanas iespēja ļauj radīt personalizētus un neaizmirstamus dekorus, kas atspoguļo klienta gaumi un svētku noskaņu. "Sparkly Dream" ir radīts, lai katra ideja pārvērstos par mirdzošu realitāti! Tā ir aktuāla ikvienam, kurš meklē ne tikai rotājumu, bet arī radošu pieredzi un iespēju radīt kaut ko īpašu.

Projekta mērķis ir attīstīt ģimenes uzņēmumu, izveidojot ērtu un lietotājam draudzīgu internetveikalu "Sparkly Dream". Mājaslapa nodrošinās klientiem vienkāršu veidu, kā apskatīt un pasūtīt ar rokām darinātus Ziemassvētku eglīšu rotājumus. Tiks piedāvāta arī iespēja personalizēt rotājumus, tādējādi apmierinot individuālās klientu vēlmes. Šis projekts palīdzēs sasniegt plašāku auditoriju, palielinās uzņēmuma atpazīstamību un veicinās pārdošanas apjomus, radot modernu un pieejamu platformu unikālu produktu iegādei.

Līdztekus biznesa attīstībai projekts palīdzēs autoram paplašināt savas zināšanas programmēšanā, iegūstot praktisku pieredzi un pielietojot mūsdienīgas tehnoloģijas programmatūras izstrādē. Tādējādi tas ne tikai stiprinās uzņēmuma konkurētspēju, bet arī sniegs būtisku profesionālo izaugsmi projektā iesaistītajām personām, radot vērtību gan pašam autoram, gan lietotājiem, gan visai uzņēmējdarbības videi.

# Uzdevuma formulējums

Projekta ietvaros tiks izstrādāts internetveikals "Sparkly Dream", kura mērķis ir nodrošināt klientiem ērtu piekļuvi Ziemassvētku eglīšu rotājumu iegādei un personalizācijai, kā arī administrācijas rīkus uzņēmuma pārvaldībai.

Mājaslapas dizains tiks veidots mūsdienīgā stilā, kas atbilst aktuālajām tīmekļa dizaina tendencēm, radot lietotnē harmonisku vidi. Tiks izvēlēta krāsu gamma un grafiskie elementi, kas uzsver svētku noskaņu un izceļ piedāvāto produkciju.

Klientu daļa būs paredzēta lietotāju ērtībai un tajā tiks iekļautas šādas funkcionalitātes:

* Jaunumi: sadaļa, kurā lietotāji varēs apskatīt aktuālo informāciju par uzņēmuma piedāvājumiem, jauniem produktiem un citiem paziņojumiem.
* Gatava produkcija: sadaļa, kurā tiks parādīti jau gatavie eglīšu rotājumi ar detalizētu aprakstu, cenām un attēliem.
* Pasūtīšana: iespēja izvēlēties un iegādāties produktus, izmantojot ērti lietojamu pasūtījuma veidlapu.
* Izejmateriālu apskate: sadaļa, kurā klienti varēs iepazīties ar materiāliem, no kuriem tiek izgatavoti rotājumi, lai veidotu personalizētus produktus.
* Savas rotaļas izveidošana: funkcionalitāte, kas ļaus klientiem personalizēt rotājumus, izvēloties krāsas, formas un dekoratīvos elementus, lai radītu unikālu dizainu.
* Saziņa ar uzņēmumu: kontaktforma, kas nodrošina ātru komunikāciju ar uzņēmuma pārstāvjiem jautājumu un ierosinājumu gadījumā.
* **Klientu atsauksmes**: sadaļa, kurā lietotāji varēs lasīt citu klientu atsauksmes par produktiem, kā arī iesniegt savas atsauksmes pēc pirkuma veikšanas. Šī funkcionalitāte palīdzēs veidot uzticību potenciālajiem klientiem un sniegs uzņēmumam iespēju uzlabot piedāvājumu, ņemot vērā klientu ierosinājumus un viedokļus.

Administrācijas daļa tiks veidota, lai nodrošinātu ērtu uzņēmuma vadību, un tajā būs šādas iespējas:

* Pasūtījumu pārvaldība: administratoriem būs iespēja apskatīt saņemtos pasūtījumus, tos rediģēt vai dzēst, ja nepieciešams, kā arī mainīt pasūtījuma statusu, piemēram, "izpildīts" vai "gaida apstiprinājumu".
* Produkcijas pārvaldība: funkcionalitāte, kas ļaus administratoriem pievienot jaunus produktus gatavo rotājumu katalogam, rediģēt esošo produkciju vai to dzēst.
* Izejmateriālu pārvaldība: iespēja pievienot jaunus materiālus, rediģēt informāciju par esošajiem vai dzēst tos no sistēmas, lai nodrošinātu aktuālu informāciju klientiem.

Visas sistēmas funkcijas būs balstītas uz datu bāzi, kas nodrošinās drošu un efektīvu datu uzglabāšanu un apstrādi. Datu bāzē tiks glabāta informācija par produktiem, materiāliem, pasūtījumiem un lietotājiem, ļaujot sistēmai ātri piekļūt nepieciešamajiem datiem. Klientu un administratoru saskarnes būs savstarpēji saistītas, un izmaiņas administrācijas daļā automātiski atspoguļosies klientu daļā.

Šāds sistēmas izstrādes plāns nodrošinās ne tikai funkcionalitāti un lietotāju ērtības, bet arī efektīvu uzņēmuma darbības pārvaldību un ilgtspējīgu attīstību.

# Programmatūras prasību specifikācija

Šī sadaļa apraksta produkta perspektīvu, funkcionālās un nefunkcionālās prasības, kā arī gala lietotāja raksturiezīmes. **Produkta perspektīvā** ir vērsta uz personalizācijas iespēju paplašināšanu, videi draudzīgu materiālu izmantošanu un ilgtspējīgas attīstības principu ievērošanu, stiprinot zīmola tēlu un klientu lojalitāti. **Funkcionālās prasības** ietver vienkāršu produktu apskati, pasūtīšanu un personalizāciju, savukārt **nefunkcionālās prasības** pieprasa mājaslapas ātru lietojamību un modernu dizainu. **Gala lietotāji** ir klienti, kas novērtē unikālus un pielāgojamus produktus, un administratori, kas nodrošina sistēmas efektīvu darbību un kvalitatīvu atbalstu.

## 2.1. Produkta perspektīva

Internetveikala "Sparkly Dream" nākotnes perspektīvas ir vērstas uz produkta attīstību, klientu pieredzes pilnveidošanu un uzņēmuma ilgtspējību. Platforma var paplašināt savu piedāvājumu, ieviešot jaunus roku darba produktus, kas piemēroti dažādiem svētkiem, piemēram, Lieldienām, dzimšanas dienām vai kāzām. Tas ļaus saglabāt klientu interesi un nodrošinās pieprasījumu visa gada garumā. Turklāt mūsdienu tehnoloģiju integrācija, piemēram, papildinātās realitātes rīki, varētu piedāvāt klientiem iespēju vizualizēt personalizētos rotājumus pirms pasūtījuma veikšanas, kas uzlabos lietotāja pieredzi un veicinās uzticību.

## 2.2. Sistēmas funkcionālās prasības

### 2.2.1. Klientu reģistrācija

Mērķis: Nodrošināt iespēju klientiem izveidot kontus mājaslapā.

Ievaddati:

1.tabula

Klientu reģistrācija

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| E-pasts | Jā | Unikāls |
| Vārds | Jā |  |
| Uzvārds | Jā |  |
| Tālrunis | Jā |  |
| Parole | Jā | Tiek saglabāts šifrētā formā |

Apstrāde: Kad reģistrācijas poga tiek nospiesta, sistēma pārbauda vai visi ievadlauki ir aizpildīti. Tālāk tiek pārbaudīti dati, vai datu bāzē neeksistē ievadītais e-pasts un lietotājvārds. Ja visi ievadlauki ir aizpildīti korekti, tad klients tiek reģistrēts, un viņa dati tiek saglabāti datu bāzē ar šifrētu paroli.

Izvaddati:

1. Kļūdas paziņojums, ja gadījumā datu bāze eksistē ievadītais e-pasts.
2. Kļūdas paziņojums, ja visi ievadlauki nav aizpildīti.
3. Paziņojums par veiksmīgu reģistrāciju.

### 2.2.2. Klientu autorizēšana sistēma

Mērķis: Funkcija nodrošina iespēju klientiem autorizēties savā kontā.

Ievaddati:

2.tabula

Klientu autorizēšana sistēmā

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| E-pasts | Jā | Unikāls |
| Parole | Jā | Tiek saglabāts šifrētā formā |

Apstrāde: Kad autorizācijas poga tiek nospiesta, sistēma pārbauda vai visi ievadlauki ir aizpildīti. Tālāk tiek pārbaudīti dati, lai tie sakristu ar tiem, kas atrodas datu bāzē. Ja visi ievadlauki ir aizpildīti korekti, tad klients ir autorizēts savā kontā.

Izvaddati:

1. Kļūdas paziņojums, ja ievadītais e-pasts neeksistē datu bāze.
2. Kļūdas paziņojums, ja tiek ievadīta neparieza parole.
3. Kļūdas paziņojums, ja visi ievadlauki nav aizpildīti.

### 2.2.3. Klienta izlogošanas

Mērķis: Nodrošināt autorizētajam klientam izlogoties no sava konta

Ievaddati: Noklikšķināt uz izlogošanas pogu

Apstrāde: Sistēma pārtrauc sessiju, un atgriež klientu uz mājaslapas sākumlapu.

Izvaddati: Lietotāja atgriešana uz mājaslapas sākumlapu.

### 2.2.4. Klientu dzēšana no sistēmas

Mērķis: Nodrošināt iespēju klientam dzēst savu kontu.

Ievaddati: Klientam jābūt autorizētam, un sadaļā “Profils” jānospiež poga “Dzēst kontu”, un jāapstiprina darbība.

Apstrāde: Funkcija pārbauda vai lietotājs ir autorizēts, dzēš klientu no datu bāzes un pārtrauc sesiju, atgriežot klientu uz sākumlapu.

Izvaddati: Klients saņem paziņojumu par veiksmīgu konta dzēšanu un tiek atgriezts uz sākumlapu.

### 2.2.5. Klienta paroles maiņa

Mērķis:Funkcija ir paredzēta klientiem, lai tie varētu mainīt savu autentifikācijas paroli.

Ievaddati:

**3.tabula**

**Dati paroles maiņai**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| E-pasts | Jā |  |
| Jaunā parole | Jā |  |
| Parole atkārtoti | Jā |  |

Apstrāde: Kad tiek nospiesta poga “Saglabāt”, sistēma pārbauda vai klients ar tādu e-pastu eksistē un vai ievadītas paroles abos laukos sakrīt. Ja tā ir, tad sistēmā tiek pievienota jauna šifrēta parole.

Izvaddati:

1) Paziņojums par veiksmīgu paroles maiņu;

2) Kļūdas ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.6. Produkcijas apskate

Mērķis: Nodrošināt mājaslapas apmeklētajiem apskatīt esošo produkciju attiecīgajā mājaslapas sadaļā.

Apstrāde: Attiecīgajā mājaslapas sadaļā, tiek paradīta visa produkcija no datu bāzes.

Izvaddati: Kļūdas paziņojums, ja datu bāzē nav nevienas produkcijas.

### 2.2.7. Izejmateriālu apskate

Mērķis: Nodrošināt mājaslapas apmeklētajiem apskatīt izejmateriālus, no kuriem tiek veidotas rotaļlietas attiecīgajā mājaslapas sadaļā.

Apstrāde: Attiecīgajā mājaslapas sadaļā, tiek paradīti visi izejmateriāli no datu bāzes.

Izvaddati: Kļūdas paziņojums, ja datu bāzē nav neviena ieraksta.

### 2.2.8. Atsauksmju apskate

Mērķis: Nodrošināt mājaslapas apmeklētajiem apskatīt atsauksmes attiecīgajā mājaslapas sadaļā.

Apstrāde: Attiecīgajā mājaslapas sadaļā, tiek paradīta visas atsauksmes no datu bāzes.

Izvaddati: Kļūdas paziņojums, ja datu bāzē nav neviena ieraksta.

### 2.2.9. Jaunumu apskate

Mērķis: Nodrošināt mājaslapas apmeklētajiem apskatīt jaunumus attiecīgajā mājaslapas sadaļā.

Apstrāde: Attiecīgajā mājaslapas sadaļā, tiek paradīti visi jaunumi no datu bāzes.

Izvaddati: Kļūdas paziņojums, ja datubāzē nav neviena ieraksta.

### 2.2.10. Produktu pievienošana iepirkuma grozā

Mērķis: Nodrošināt klientiem pievienot izvēlēto produktu savā iepirkuma grozā.

Ievaddati: Pie izvēlēta produkta klikšķināt uz pogu “Pievienot grozam”.

Apstrāde: Sistēma pārbauda, vai klients ir autorizēts, ja tā ir, tad produkts tiek pievienots iepirkuma grozā, pretēja gadījuma, sistēma atver ielogošanas lapu.

Izvaddati:

Kļūdas paziņojums, ja klients nav autorizēts.

Paziņojums par veiksmīgu pievienošanu grozā.

### 2.2.11. Produktu dzēšana no iepirkuma grozā

Mērķis: Nodrošināt iespēju klientiem dzēst produktu no iepirkuma groza.

Ievaddati: Klients klikšķina uz dzēšanas pogu pie konkrēta produkta, ko vēlās dzēst no iepirkuma groza.

Apstrāde: Produkts tiek izdzēsts no klienta iepirkuma groza.

Izvaddati: Paziņojums par veiksmīgu produkta dzēšanu no iepirkuma groza.

### 2.2.12. Saziņa ar pārdevēju

Mērķis: Nodrošināt iespēju klientam sazināties ar pārdevēju caur e-pastu.

Ievaddati:

**4.tabula**

**Saziņa ar pārdevēju**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Vārds | Jā |  |
| E-pasts | Jā |  |
| Tālrunis | Jā |  |
| Ziņojuma teksts | Jā |  |

Apstrāde: Kad ir nospiesta poga “Sazināties”, sistēma pārbauda visus obligāti aizpildāmos laukus un datu tipus. Ja tā ir, ziņa tiek nosūtīta uz organizācijas e-pastu.

Izvaddati:

Kļūdas paziņojums, ja nav aizpildīti visi ievades lauki.

Paziņojums par veiksmīgu e-pasta nosūtīšanu.

### 2.2.13. Savas rotaļas izveide

Mērķis: Sadaļā “Izveido pats” nodrošināt iespēju klientam izvēlēties izejmateriālus, lai izveidotu savu rotaļu.

Ievaddati:

**5.tabula**

**Savas rotaļas izveide**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Forma | Jā | Izvēlēties no saraksta |
| Pamatkrāsa | Jā | Izvēlēties no saraksta |
| Mālu figūra | Nē | Izvēlēties no saraksta |
| Dekorējums | Jā | Izvēlēties no saraksta |

Apstrāde: Sistēma pārbauda vai klients ir autorizēts, ja tā ir, tad saglabā klienta izvēlētus parametrus datu bāzē un parāda tos iepirkuma grozā. Ja klients nav autorizēts, tad sistēma atver ielogošanas logu.

Izvaddati: Paziņojums par veiksmīgu rotaļas izveidi.

### 2.2.14. Pasūtījuma veikšana

Mērķis: Nodrošināt klientiem veikt pasutījumu.

Ievaddati:

6.tabula

Pasūtījuma veikšana

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| E-pasts | Jā |  |
| Vārds | Jā |  |
| Uzvārds | Jā |  |
| Tālrunis | Jā |  |
| Adrese | Jā |  |
| Kartes numurs | Jā |  |
| Kartes derīguma datums | Jā |  |
| Kartes CVC kods | Jā |  |

Apstrāde: Sistēma pārbauda, vai visi ievadlauki ir aizpildīti, un pievieno pasūtījumu datu bāzē ar statusu “Iesniegts”.

Izvaddati:

1. Kļūdas paziņojums, ja visi ievadlauki nav aizpildīti.
2. Paziņojums, par veiksmīgu pasūtījumu.

### 2.2.15. Administratoru un moderatoru ielogošanas sistēma

Mērķis: Funkcija nodrošina iespēju administratoriem un moderatoriem ielogoties savā kontā.

Ievaddati:

7.tabula

Administratoru un moderatoru ielogošanas sistēma

|  |  |  |
| --- | --- | --- |
| Nosaukums | Obligāts | Piezīmes |
| Lietotājvārds | Jā | Unikāls |
| Parole | Jā | Tiek saglabāts šifrētā formā |

Apstrāde: Kad autorizācijas poga tiek nospiesta, sistēma pārbauda vai visi ievadlauki ir aizpildīti. Tālāk tiek pārbaudīti dati, lai tie sakristu ar tiem, kas atrodas datu bāzē. Ja visi ievadlauki ir aizpildīti korekti, tad lietotājam atvērās atbilstoši viņa lomai sadaļas. .

Izvaddati:

1. Kļūdas paziņojums, ja ievadītais lietotājvārds vai parole nesakrīt ar datu bāzi.
2. Kļūdas paziņojums, ja visi ievadlauki nav aizpildīti.

### 2.2.16. Lietotāja pievienošana sistēmā

Mērķis: Funkcija paredzēta autorizētajam administratoram pievienot jaunu lietotāju sistēmā.

Ievaddati:

8.tabula

Lietotāja pievienošana sistēmā

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| E-pasts | Jā | Unikāls |
| Vārds | Jā |  |
| Uzvārds | Jā |  |
| Lietotājvārds | Jā |  |
| Tālrunis | Jā |  |
| Parole | Jā | Tiek saglabāts šifrētā formā |
| Loma | Jā | Izvēlēties no saraksta |

Apstrāde: Kad poga “Pievienot” tiek nospiesta, sistēma pārbauda, ka visi ievadlauki ir aizpildīti, ka arī pārbauda, vai datu bāzē neeksistē ievadītais e-pasts un lietotājvārds. Ja viss ir aizpildīts korekti, tad jaunais lietotājs tiek saglabāts datu bāzē.

Izvaddati:

1. Kļūdas paziņojums, ja visi ievadlauki nav aizpildīti.
2. Kļūdas paziņojums, ja datu bāzē jau eksistē ievadītais e-pasts vai lietotājvārds.
3. Paziņojums par veiksmīgu lietotāja pievienošanu.

### 2.2.17. Lietotāja izlogošanas

Mērķis: Nodrošināt autorizētajam lietotājam izlogoties no sava konta

Ievaddati: Noklikšķināt uz izlogošanas pogu

Apstrāde: Sistēma pārtrauc sessiju, un atgriež lietotāju uz autorizēšanas logu.

Izvaddati: Lietotāja atgriešana uz autorizēšanas logu.

### 2.2.17. Lietotāja dzēšana no sistēmas

Mērķis: Funkcija paredzēta autorizētajam administratoram dzēst lietotāju no sistēmas.

Ievaddati: Pie konkrēta lietotāja jānospiež poga “Dzēst”.

Apstrāde: Sistēma prasa apstiprinājumu, un pēc tā dzēš lietotāju.

Izvaddati: Paziņojums par veiksmīgu lietotāja dzēšanu.

### 2.2.18. Produkta pievienošana

Mērķis: Nodrošināt sistēmas administratoriem vai moderatoriem iespēju pievienot jaunus produktus.

Ievaddati:

**9.tabula**

**Produkta pievienošana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Nosaukums | Jā |  |
| Forma | Jā | Izvēlēties no saraksta |
| Pamatkrāsa | Jā | Izvēlēties no saraksta |
| Dekorējums | Jā | Izvēlēties no saraksta |
| Mālu figūra | Jā | Izvēlēties no saraksta |
| Apraksts | Jā |  |
| Attēls | Jā |  |
| Cena | Jā |  |

Apstrāde: Kad poga “Pievienot” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu produkta pievienošanu sistēmai.

2) Kļūdas paziņojums, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.19. Produkta rediģēšana

Mērķis: Nodrošināt sistēmas lietotājiem rediģēt produktu.

Ievaddati: Skat. 8. tabulu

Apstrāde: Kad poga “Saglabāt” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu produkta rediģēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.20. Produktu dzēšana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju dzēst produktus.

Ievaddati: Lietotājs klikšķina uz pogu “Dzēst” pie konkrēta produkta, ko vēlās dzēst.

Apstrāde: Kad poga “Dzēst” tiek nospiesta, sistēma dzēš visu ierakstu par noteiktu produktu.

Izvaddati:

1) Paziņojums par veiksmīgu produkta dzēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.21. Jaunumu pievienošana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju pievienot jaunumus.

Ievaddati:

**10.tabula**

**Jaunumu pievienošana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Nosaukums | Jā |  |
| Teksts | Jā |  |
| Attēls | Jā |  |

Apstrāde: Kad poga “Pievienot” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu jaunuma pievienošanu sistēmai.

2) Kļūdas paziņojums, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.23. Jaunumu rediģēšana

Mērķis: Nodrošināt sistēmas lietotājiem rediģēt jaunumus.

Ievaddati: Skat. 9. tabulu

Apstrāde: Kad poga “Saglabāt” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu jaunuma rediģēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.24. Jaunumu dzēšana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju dzēst jaunumus.

Ievaddati: Lietotājs klikšķina uz pogu “Dzēst” pie konkrēta jaunuma, ko vēlās dzēst.

Apstrāde: Kad poga “Dzēst” tiek nospiesta, sistēma dzēš visu ierakstu par noteiktu jaunumu.

Izvaddati:

1) Paziņojums par veiksmīgu jaunumu dzēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.25. Izejmateriālu pievienošana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju pievienot izejmateriālus.

Ievaddati:

11.tabula

Izejmateriālu pievienošana

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Nosaukums | Jā |  |
| Kategorija | Jā | Izvēlēties no saraksta |
| Attēls | Jā |  |

Apstrāde: Kad poga “Pievienot” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu izejmateriālu pievienošanu sistēmai.

2) Kļūdas paziņojums, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.26. Izejmateriālu rediģēšana

Mērķis: Nodrošināt sistēmas lietotājiem rediģēt izejmateriālus.

Ievaddati: Skat. 11. tabulu

Apstrāde: Kad poga “Saglabāt” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu izejmateriālu rediģēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.27. Izejmateriālu dzēšana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju dzēst izejmateriālus.

Ievaddati: Lietotājs klikšķina uz pogu “Dzēst” pie konkrēta izejmateriāla, ko vēlās dzēst.

Apstrāde: Kad poga “Dzēst” tiek nospiesta, sistēma dzēš visu ierakstu par noteiktu jaunumu.

Izvaddati:

1) Paziņojums par veiksmīgu jaunumu dzēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.28. Klientu pievienošana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju pievienot klientus.

Ievaddati:

**12.tabula**

**Klientu pievienošana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| E-pasts | Jā | Unikāls |
| Vārds | Jā |  |
| Uzvārds | Jā |  |
| Tālrunis | Jā |  |
| Parole | Jā | Tiek saglabāts šifrētā formā |

Apstrāde: Kad poga “Pievienot” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu klienta pievienošanu sistēmai.

2) Kļūdas paziņojums, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.29. Klienta rediģēšana

Mērķis: Nodrošināt sistēmas lietotājiem rediģēt klientus.

Ievaddati: Skat. 12. tabulu

Apstrāde: Kad poga “Saglabāt” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu klienta rediģēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.30. Klientu dzēšana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju dzēst klientus.

Ievaddati: Lietotājs klikšķina uz pogu “Dzēst” pie konkrēta klienta, ko vēlās dzēst.

Apstrāde: Kad poga “Dzēst” tiek nospiesta, sistēma dzēš visu ierakstu par noteiktu klientu.

Izvaddati:

1) Paziņojums par veiksmīgu klienta dzēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.31. Pasūtījuma pievienošana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju pievienot pasūtījumus.

Ievaddati:

13.tabula

Pasūtījuma pievienošana

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| E-pasts | Jā |  |
| Vārds | Jā |  |
| Uzvārds | Jā |  |
| Tālrunis | Jā |  |
| Adrese | Jā |  |
| Kartes numurs | Jā |  |
| Kartes derīguma datums | Jā |  |
| Kartes CVC kods | Jā |  |

Apstrāde: Kad poga “Pievienot” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu pasūtījuma pievienošanu sistēmai.

2) Kļūdas paziņojums, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.32. Pasūtījuma rediģēšana

Mērķis: Nodrošināt sistēmas lietotājiem rediģēt pasūtījumus.

Ievaddati: Skat. 13. tabulu

Apstrāde: Kad poga “Saglabāt” tiek nospiesta, sistēma veic pārbaudi, vai visi obligāti lauki ir aizpildīti un pārbauda visus datu tipus.

Izvaddati:

1) Paziņojums par veiksmīgu pasūtījuma rediģēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

### 2.2.33. Pasūtījuma dzēšana

Mērķis: Nodrošināt sistēmas lietotājiem iespēju dzēst klientus.

Ievaddati: Lietotājs klikšķina uz pogu “Dzēst” pie konkrēta pasūtījuma, ko vēlās dzēst.

Apstrāde: Kad poga “Dzēst” tiek nospiesta, sistēma dzēš visu ierakstu par noteiktu klientu.

Izvaddati:

1) Paziņojums par veiksmīgu klienta dzēšanu sistēmā.

2) Kļūdu ziņojumi, ja procesā radusies kļūda vai nepieciešamas papildu darbības.

## 2.3. Sistēmas nefunkcionālās prasības

1) Tīmekļa vietne ir izstrādāta latviešu valodā, lai nodrošinātu tās piemērotību vietējiem lietotājiem un viegli saprotamu informāciju dažādām sabiedrības grupām neatkarīgi no lietotāju vecuma, izglītības vai tehniskajām zināšanām.

2) Mājaslapas dizains ir estētisks, mūsdienīgs un viegli pārskatāms, lai mājaslapas apmeklētajiem būtu viegli mijiedarboties.

3) Mājaslapa ir responsīva visām ierīcēm.

4) Mājaslapa ir pilnībā funkcionāla visās populārākajās pārlūkprogrammās (Chrome, Firefox, Edge, Safari).

5) Katra mājaslapas sadaļa tiek atvērta jaunajā logā.

## 2.4. Gala lietotāja raksturiezīmes

Mājaslapas lietotājiem nav īpaši jābūt datora prasmēm, tāpēc ka mājaslapa ir uztaisīta ar vieglu un saprotamu interfeisu, gan lietotājiem, gan administratoriem.

* Klienti: Šajā mājaslapa klienta tiesības ir apskatīt visu publisko mājaslapu, veikt pasūtījumus, izstrādāt savu īpašu rotaļu un sazināties ar organizāciju.
* Administratori: Administratora tiesības ir CRUD (Create, read, update, edit) operācijas saistībā ar klientiem, lietotājiem, produkciju, izejmateriāliem, pasūtījumiem un jaunumiem.
* Moderatori: Moderatora tiesības ir CRUD (Create, read, update, edit) operācijas saistībā ar klientiem, produkciju, izejmateriāliem, pasūtījumiem un jaunumiem.

# Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums

Lai nodrošinātu "Sparkly Dream" platformas augstu veiktspēju, drošību un lietošanas ērtumu, ir rūpīgi izvēlēti piemēroti izstrādes rīki un tehnoloģijas. Šajā sadaļā tiek apskatīti izmantotie programmēšanas rīki, datubāzes risinājumi, dizaina un lietotāja interfeisa izstrādes instrumenti, kā arī izvēles pamatojums, kas balstīts uz projekta prasībām un tehniskajām iespējām.

## 3.1. Izvēlēto risinājuma līdzekļu un valodu apraksts

* **HMTL**

HTML (HyperText Markup Language) ir izmantots kā galvenais valodas pamats "Sparkly Dream" platformas lietotāja saskarnes izveidei. Tas nodrošina mājaslapas struktūru un ļauj efektīvi organizēt satura izvietojumu, kas ir būtiski ērtai navigācijai un lietotāju pieredzei.

* **CSS**

CSS (Cascading Style Sheets) ir izmantots, lai nodrošinātu "Sparkly Dream" mājaslapas vizuālo noformējumu un lietotāja pieredzi. Ar CSS palīdzību tiek definēti krāsu salikumi, fonti, izkārtojums un animācijas, kas padara mājaslapu pievilcīgu un viegli lietojamu. Responsīvais dizains nodrošina pareizu mājaslapas attēlošanu dažādās ierīcēs, tostarp datoros, planšetēs un viedtālruņos.

* **JavaScript**

JavaScript ir populāra programmēšanas valoda, kas tiek izmantota tīmekļa izstrādē, lai pievienotu interaktivitāti un dinamisku funkcionalitāti mājaslapām. Tā darbojas lietotāja pārlūkprogrammā un ļauj manipulēt ar HTML un CSS elementiem reāllaikā. JavaScript nodrošina iespēju veidot dinamiskas saskarnes, uzlabot lietotāja pieredzi un efektīvi apstrādāt datus bez nepieciešamības pārlādēt lapu.

JavaScript tiek izmantots "Sparkly Dream" mājaslapā šādiem mērķiem:

1) Animācijas un lietotāja mijiedarbība:

JavaScript tiks izmantots, lai pievienotu dažādas animācijas, piemēram, vieglus pārejas efektus, modālos logus un hover efektus, izmantojot bibliotēkas, piemēram, GSAP vai tikai CSS klasēm pievienotas un noņemtas klases ar JavaScript.

2) Asinhrona datu ielādē (AJAX, Fetch API):

Lai uzlabotu lapas ielādēs ātrumu un lietotāja pieredzi, tiks izmantots JavaScript, lai asinhroni iegūtu un atjauninātu datus no servera, nenoslogojot visu lapu.

* **PHPMyAdmin**

phpMyAdmin ir tīmekļa lietojumprogramma, kas nodrošina MySQL datubāzes pārvaldību, izmantojot grafisko saskarni. "Sparkly Dream" projektā phpMyAdmin tiek izmantots, lai:

1) Pārvaldītu datubāzi, kas satur informāciju par lietotājiem, pasūtījumiem, produktiem un citiem sistēmas datiem.

2) Izpildītu SQL vaicājumus, lai pievienotu, rediģētu vai dzēstu ierakstus datubāzē.

phpMyAdmin ir izvēlēts, jo tas ir viegli lietojams, sniedz plašas funkcionalitātes iespējas un atbalsta MySQL/MariaDB datubāzes, kas tiek izmantotas "Sparkly Dream" projekta ietvaros.

* **PHP**

PHP (Hypertext Preprocessor) ir servera puses programmēšanas valoda, kas tiek izmantota "Sparkly Dream" platformā, lai nodrošinātu dinamisku un drošu datu apstrādi. Galvenās PHP funkcijas projektā ietver:

1) Datu apstrāde un loģika:

PHP tiek izmantots, lai apstrādātu lietotāju pieprasījumus, veiktu pasūtījumu pārvaldību un īstenotu autentifikācijas sistēmu.

2) Savienojums ar MySQL datubāzi:

PHP darbojas kā starpnieks starp mājaslapas saskarni un MySQL datubāzi, ļaujot veikt CRUD (Create, Read, Update, Delete) operācijas ar klientu datiem, pasūtījumiem un produktiem.

3) Formu validācija un drošība:

PHP tiek izmantots, lai validētu lietotāju ievadītos datus, novērstu SQL injekcijas un nodrošinātu drošu lietotāju pieredzi.

* **SQL**

SQL (Structured Query Language) ir valoda, kas tiek izmantota datu bāzu pārvaldībai un manipulācijai. "Sparkly Dream" projektā SQL tiek izmantots, lai veiktu svarīgākās datu bāzes operācijas, piemēram, datu pievienošanu, lasīšanu, atjaunināšanu un dzēšanu (CRUD operācijas). Tas nodrošina efektīvu un strukturētu piekļuvi datiem, kas nepieciešami lietotāju kontu pārvaldībai, pasūtījumu izsekošanai, produktu pievienošanai un citām sistēmas funkcijām.

* **Visual Studio Code**

Visual Studio Code ir populārs un jaudīgs koda redaktors, kas tiek plaši izmantots programmēšanas un tīmekļa izstrādes projektos. Tas piedāvā plašu funkcionalitāti, piemēram, koda krāsu atzīmēšanu, automātisko pabeigšanu, kļūdu atklāšanu un atbalstu daudziem programmēšanas valodām. Ar pieejamajiem paplašinājumiem, piemēram, atkļūdošanas rīkiem, versiju kontroles atbalstu un integrāciju ar citiem izstrādes rīkiem, Visual Studio Code padara koda rakstīšanu un pārvaldību vienkāršāku un efektīvāku. Pateicoties tā vieglajam izmēram un ātrai veiktspējai, tas ir piemērots gan maziem, gan lieliem projektiem, nodrošinot labu līdzsvaru starp funkcionalitāti un lietotāja pieredzi.

* **Github**

"Sparkly Dream" projektā GitHub tiek izmantots kā versiju kontroles rīks, lai nodrošinātu projekta koda pārvaldību un saglabātu tā organizētību. Ar GitHub palīdzību tiek sekots visām izmaiņām kodā, ļaujot viegli pārvaldīt dažādas koda versijas un atgriezties pie iepriekšējiem darba posmiem, ja nepieciešams. GitHub arī ļauj izstrādātājam vienkārši veikt izmaiņas un saglabāt tās drošā veidā, nodrošinot projektu ar versiju vēsturi un iespēju apstrādāt izmaiņas bez riska tās pazaudēt.

* **Stripe API**

Stripe API ir populārs maksājumu apstrādes rīks, kas ļauj tiešsaistes platformām un lietotnēm droši apstrādāt maksājumus. Stripe piedāvā plašu funkcionalitāti, kas ietver maksājumu pieņemšanu ar kredītkartēm, debetkartēm, elektroniskajiem maciņiem (piemēram, Apple Pay, Google Pay) un pat dažādām banku pārskaitījumu metodēm. Tā ir elastīga un viegli integrējama platforma, kas nodrošina vienkāršu un drošu risinājumu uzņēmumiem, kas vēlas veikt maksājumus tiešsaistē. "Sparkly Dream" projektā Stripe API tiks izmantots, lai apstrādātu maksājumus par pasūtījumiem, kas tiek veikti platformā.

* **PHP Mailer**

PHP Mailer ir plaši izmantota bibliotēka, kas ļauj nosūtīt e-pastus izmantojot PHP, tā vienkāršo e-pasta funkcionalitātes integrēšanu. "Sparkly Dream" projektā PHP Mailer tiks izmantots, lai nodrošinātu komunikāciju starp platformu un klientiem ļaujot lietotājiem viegli nosūtīt ziņojumus vai paziņojumus organizācijai.

## 3.2. Iespējamo (alternatīvo) risinājuma līdzekļu un valodu apraksts

* **Markdown**

Markdown ir vienkāršs un viegls marķējuma valoda, kas tiek izmantota, lai rakstītu tekstu tīmekļa lapām, bieži tiek izmantots dokumentācijas un saturu rakstīšanai. Tas ļauj ātri formatēt tekstu, piemēram, pievienot virsrakstus, sarakstus un linkus.

Kāpēc netiks izmantots: Markdown nav piemērots sarežģītākas struktūras vai dinamisku tīmekļa lapu veidošanai, jo tas ir ierobežots tikai tekstu un vienkāršu formātu veidošanā, tāpēc tas nav ideāli piemērots "Sparkly Dream" projektam, kur nepieciešams plašs un dinamisks HTML atbalsts.

* **Python**

Python ir augstākā līmeņa, interpretēta programmēšanas valoda, kas ir ļoti populāra tīmekļa izstrādē, datu apstrādē, mākslīgā intelekta un automatizācijas risinājumos. Tas piedāvā plašu bibliotēku un ietvaru izvēli, piemēram, Django, Flask, un FastAPI, kas ļauj viegli veidot tīmekļa lietotnes, apstrādāt datus un izveidot REST API.

Kāpēc netiks izmantots: Lai gan Python ir ļoti elastīgs un piemērots daudziem lietojumiem, "Sparkly Dream" projektā varētu būt labāk izmantot PHP, kas ir jau iekļauts tehnoloģiju kaudzē, un tas ir īpaši piemērots tīmekļa serveru un dinamisku tīmekļa lapu izveidei. Python ir nedaudz smagāks un var radīt liekus sarežģījumus, īpaši ja Python nav galvenā valoda tīmekļa servera izstrādē jūsu izvēlētajā vidē. Python varētu būt piemērots citas funkcionalitātes pievienošanai, bet šajā gadījumā PHP ir vienkāršāks un efektīvāks risinājums.

* **Tailwind CSS**

Tailwind CSS ir utilitāšu pamatots CSS ietvars, kas ļauj ļoti ātri izveidot tīmekļa lapas dizainu, izmantojot nelielas CSS klases tieši HTML elementiem.

Kāpēc netiks izmantots: Tailwind CSS var būt grūti pielāgojams, ja nepieciešama sarežģītāka dizaina izstrāde, jo tā pieeja ir ļoti atkarīga no tiešiem stilistikas iestatījumiem HTML, kas var apgrūtināt lielāku pielāgojumu un atbalsta veidošanu.

* **Node.js**

Node.js ļauj JavaScript darboties servera pusē, piedāvājot ļoti ātru un mērogojamu risinājumu tīmekļa lietotnēm.

Kāpēc netiks izmantots: Node.js var būt ļoti piemērots ļoti mērogojamām aplikācijām, bet PHP jau labi atbilst "Sparkly Dream" vajadzībām, piedāvājot ļoti stabilu un pārbaudītu risinājumu servera puses datu apstrādei, kas ir vieglāk uzturams un vairāk atbalstīts izvēlētajā tehnoloģiju kaudzē.

* **JetBrains WebStorm**

WebStorm ir pilnīga integrētā attīstības vide (IDE), kas piedāvā plašas iespējas JavaScript un Node.js izstrādei.

Kāpēc netiks izmantots: WebStorm var būt pārāk smags un lēns salīdzinājumā ar vieglāku, ātrāku un elastīgāku Visual Studio Code, kas ļauj ātri veikt izmaiņas un ir ļoti pielāgojams.

# Sistēmas modelēšana un projektēšana

## 4.1. Sistēmas struktūras modelis

### 4.1.1. Izvietojuma (Deployment) diagramma / komponentu (Component) diagramma

### 4.1.2. Klašu (Class) diagramma / ER (Entity Relationship) diagramma

## 4.2. Funkcionālais un dinamiskais sistēmas modelis

### 4.2.1. Aktivitāšu (Activity) diagramma

### 4.2.2. Lietojumgadījumu (Use Case) diagramma

### 4.2.3. Stāvokļu (State) diagramma

## 4.3. Datu struktūru apraksts

# Lietotāju ceļvedis

# Testēšanas dokumentācija

## 6.1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums

## 6.2. Alternatīvās testēšanas metodes un rīki

## 6.3. Testpiemēru kopa

## 6.4. Testēšanas žurnāls

# Secinājumi

# Lietoto saīsinājumu un terminu skaidrojums

# Literatūras un informācijas avotu saraksts

# Pielikumi