Liepājas Valsts tehnikums

**Mākslas pārdošanas platforma “Artify”**

Kvalifikācijas eksāmena praktiskās daļas dokumentācija

Profesionālā kvalifikācija ........................................................................

Grupas nosaukums .........................................................................

Projekta izstrādātājs ...............................................................

/vārds, uzvārds, paraksts/

Eksāmena datums 2024. gada\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Liepāja 2024

**Satura rādītājs**

[Ievads 3](#_Toc159706193)

[1. Uzdevuma formulējums 4](#_Toc159706194)

[2. Programmatūras prasību specifikācija 5](#_Toc159706195)

[2.1. Produkta perspektīva 5](#_Toc159706196)

[2.2. Sistēmas funkcionālās prasības 5](#_Toc159706197)

[2.3. Sistēmas nefunkcionālās prasības 17](#_Toc159706198)

[2.4. Gala lietotāja raksturiezīmes 18](#_Toc159706199)

[3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums 20](#_Toc159706200)

[3.1. Izvēlēto risinājuma līdzēkļu un valodu apraksts 20](#_Toc159706201)

[3.1.1. Front-End tehnoloģijas 20](#_Toc159706202)

[3.1.2. Back-End tehnoloģijas 20](#_Toc159706203)

[3.1.3. Papildus back-end rīki 21](#_Toc159706204)

[3.1.4. Pārlūkprogramma 21](#_Toc159706205)

[3.1.5. Koda redaktors un versionēšana 21](#_Toc159706206)

[3.1.6. Izstrādes un kvalitātes rīki 21](#_Toc159706207)

[3.2. Iespējamo (alternatīvo) risinājuma līdzekļu un valodu apraksts 22](#_Toc159706208)

[4. Sistēmas modelēšana un projektēšana 24](#_Toc159706209)

[4.1. Sistēmas struktūras modelis 24](#_Toc159706210)

[4.2. Klašu diagramma / ER diagramma 24](#_Toc159706211)

[4.3. Funkcionālais un dinamiskais sistēmas modelis 24](#_Toc159706212)

[4.4. Aktivitāšu diagramma (Activity) 24](#_Toc159706213)

[4.5. Lietojumgadījumu diagramma (Use Case) 24](#_Toc159706214)

[4.6. Sistēmas moduļu apraksts un algoritmu shēmas 24](#_Toc159706215)

[5. Lietotāju ceļvedis 25](#_Toc159706216)

[6. Testēšanas dokumentācija 26](#_Toc159706217)

[6.1. Izvēlētās testēšanas metodes, rīku apraksts un pamatojums 26](#_Toc159706218)

[6.2. Testpiemēru kopa 26](#_Toc159706219)

[6.3. Testēšanas žurnāls 26](#_Toc159706220)

[7. Secinājumi 27](#_Toc159706221)

[8. Lietoto terminu un saīsinājumu skaidrojumi 28](#_Toc159706222)

[9. Literatūras un informācijas avotu saraksts 29](#_Toc159706223)

[PIELIKUMS 30](#_Toc159706224)

# Ievads

Šis projekts, "Artify," ir mans mēģinājums radīt un ieviest digitālo risinājumu, kas uzlabo mākslas darbu reģistrācijas un tirdzniecības procesus. Ar mērķi veicināt mākslas darbu pieejamību un veidot dinamisku mākslas pārdošanas platformu, "Artify" nes sevī vīziju par ērtu telpu, kur mākslinieki var izpausties un piedāvāt savus darbus, bet arī kur mākslas cienītāji var atrast un iegādāties daudzveidīgu un unikālu mākslu.

Šajā dokumentā iekļauta informācija par:

* Programmatūras prasību specifikāciju, kas nosaka funkcionālās prasības.
* Izstrādes līdzekļu, rīku aprakstu un izvēles pamatojumu, kas skaidro tehniskos lēmumus un risinājuma izvēles loģiku.
* Sistēmas modelēšanu un projektēšanu, sniedzot pārskatu par mākslas platformas struktūru un funkcionalitāti.
* Lietotāju ceļvedi, kas palīdzēs izprast, kā lietotāji izmantos un mijiedarbosies ar platformu.
* Testēšanas dokumentāciju, kas nodrošina platformas uzticamību un drošību.
* Secinājumus, kas atspoguļo darba rezultātus un projektu kopumā.

Iekļaujot mākslas darbu pārdošanas aspektu, "Artify" piedāvā gan māksliniekiem, gan pircējiem iespēju piedalīties un dalīties mākslā, veidojot savstarpēji izdevīgu vide.

Projekta pamatuzdevumi ietver mākslas darbu reģistrēšanas un pārvaldības procesu automatizāciju, nodrošinot māksliniekiem ērtu rīku klāstu darbu augšupielādei, aprakstiem un cenu noteikšanai. Tāpat tiek uzsvērta drošība un datu precizitāte, lai mākslinieki un pircēji varētu justies pārliecināti par katru darbu un tā vērtību.

Platforma ir izstrādāta, ņemot vērā modernās tehnoloģijas un lietotāju pieredzi. Lietotājiem tiek nodrošinātas intuitīvas funkcijas, kas atvieglo mākslas darbu pārlūkošanu, sarakstu veidošanu un pirkumu veikšanu. "Artify" iekļauj arī sociālo elementu, kurā lietotāji var izveidot savus profilus, sekojot mīļākajiem māksliniekiem.

Projekta centrā ir vīzija par globālu mākslas tirgu demokratizāciju, kuras pamatā ir ideja, ka ikviens mākslinieks no jebkuras pasaules malas var piedalīties un piedāvāt savu unikālo mākslu, bet katrs pircējs var iegādāties un iekļaut mākslu savā ikdienas dzīvē. "Artify" ir ne tikai mana personīgā misija radīt digitālu mākslas telpu, bet arī solis uz priekšu, kurā es cenšos veidot uzņēmējdarbības un mākslas integrāciju digitālajā pasaulē.

# 1. Uzdevuma formulējums

Projekta "Artify" centrālais mērķis ir radīt un ieviest digitālo risinājumu, kas pilnveido un apvieno visas būtiskākās funkcionalitātes saistībā ar mākslas darbu reģistrāciju, pārvaldību un tirdzniecību. Šī platforma izceļas ar plašu un ārkārtīgi daudzpusīgu funkcionalitāti, sniedzot gan māksliniekiem, gan mākslas entuziastiem ērtu un efektīvu rīku komplektu.

Pirmkārt, māksliniekiem piedāvājas iespēja augšupielādēt savus darbus platformā, turklāt sniedzot detalizētu aprakstu, kas papildina mākslas darba kontekstu un īpašības. Turpinot, māksliniekiem tiek nodrošināta iespēja veidot un pielāgot savus personīgos profilus, iekļaujot biogrāfiskus datus, aprakstu un saites uz citiem sociālajiem tīkliem, piedāvājot lietotājiem dziļāku ieskatu mākslinieku radīšanas procesā un personīgajā pieredzē.

Platforma piedāvā plašu mākslas darbu pārlūkošanas galeriju, kas ļauj lietotājiem izmantot dažādus kategorijas, lai atrastu tieši to, kas viņus interesē. Turklāt, sistēma nodrošina drošu un ērtu pirkšanas un pārdošanas procesu.

Visu šo funkciju ietvaros, platforma tiek izstrādāta ar lietotājiem draudzīgu saskarni, nodrošinot vienkāršu un intuitīvu platformas izmantošanu gan pieredzējušiem, gan jauniem lietotājiem. "Artify" ne tikai piedāvā digitālo risinājumu mākslas darbu tirgum, bet arī padara mākslu pieejamu plašai auditorijai.

# 2. Programmatūras prasību specifikācija

Projekta "Artify" programmatūras prasību specifikācija (PPS) apkopo būtiskos datus, nepieciešamos izstrādes procesā. Tā izklāsta produkta konkrētās funkcionālās un nefunkcionālās prasības. Šī sadaļa kalpo kā ceļvedis, nodrošinot kopēju redzējumu starp iesaistītajiem un veicinot sistēmatisku pieeju, lai radītu lietotājam draudzīgu, drošu un mērogojamu tirgus vietu mākslas izpētei un darījumiem.

## 2.1. Produkta perspektīva

"Artify" platforma atšķirās no esošajiem piedāvājumiem digitālajā mākslas tirgū, piedāvājot papildu funkcionalitāti un īpašumus, kas atvieglos mākslas darbu pārdošanu un pirkšanu.

Atšķirība no esošajām platformām un produkta priekšrocības:

**2.1.1. Vienkāršība**

* Apraksts: "Artify" fokusējas uz vienkāršību, piedāvājot izmantojamas darbības un skaidru lietojumprogrammu.
* Priekšrocība: Nodrošina lietotājiem ērtu pieredzi bez sarežģījumiem.

**2.1.2. Dinamisks un atvērts mākslas tirgus**

* Apraksts: "Artify" ir vienkāršs mākslas tirgus, kur mākslinieki var piedāvāt savus darbus, un pircēji var tos iegādāties, nodrošinot vienkāršību un efektivitāti.
* Priekšrocība: Ļauj dalībniekiem viegli un ērti tirgoties ar mākslas darbiem bez liekiem sarežģījumiem.

**2.1.3. Ātra darbu publicēšana, pārbaudīšana un meklēšana**

* Apraksts: "Artify" ļauj māksliniekiem nekavējoties publicēt savus darbus pēc moderatora pārbaudes un nodrošina, ka tie ir viegli atrodami tirgū.
* Priekšrocība: Ātrums un pieejamība veicina aktīvu mākslas darbu piedāvāšanu un pirkumu veikšanu.

## 2.2. Sistēmas funkcionālās prasības

**2.2.1. Lietotāja reģistrācija**

Ievaddati:

**1. tabula**

**Reģistrācijas dati**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Vārds | Jā |  |
| Uzvārds | Jā |  |
| Lietotājvārds | Jā | Jābūt unikālam |
| E-pasta adrese | Jā | Jābūt unikālam |
| Parole | Jā | Parolei jāsatur vismaz vienu skaitli, vienu lielu burtu un vienu speciālo simbolu. Minimālais paroles garums ir 8 simboli. Parolei jābūt šifrētai. |

Apstrāde:

* Pārbaude, vai e-pasta adrese ir derīga un unikāla sistēmā.
* Pārbaude, vai lietotājvārds ir unikāls sistēmā.
* Paroles prasību pārbaude (minimālais simbolu skaits, lietojot gan burtus, gan ciparus, gan speciālo simbolu).
* Paroles šifrēšana.
* Jaunā lietotāja ievadīto datu saglabāšana datu bāzē.

Izvaddati:

* Apstiprinājuma e-pasta nosūtīšana lietotājam.
* Informācija par veiksmīgu reģistrāciju vai kļūdām, ja tādas rodas.

**2.2.2. Ielogošanās sistēmā**

Ievaddati:

**2. tabula**

**Reģistrācijas dati**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Lietotājvārds | Jā |  |
| E-pasta adrese | Jā |  |
| Parole | Jā |  |

Apstrāde:

* Pārbaude, vai lietotājs ar ievadīto e-pasta adresi vai lietotājvārdu eksistē sistēmā.
* Salīdzināšana ievadītās paroles ar sistēmā glabāto paroli.
* JSON Web Token (JWT) ģenerēšana un saglabāšana autentificētiem lietotājiem.

Izvaddati:

* Paziņojums par veiksmīgu pieslēgšanos vai kļūdas paziņojums.

**2.2.3. Izlogošanās no sistēmas**

Apstrāde:

* Lietotāja JWT dzēšana no localStorage. (Lietotājs ir izlogojies)
* Lietotāja novirzīšana uz ielogošanas lapu.

**2.2.4. Mākslas darbu pievienošana**

Ievaddati:

**3. tabula**

**Mākslas darbu pievienošanas dati**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Attēli | Jā | Maksimāli 5 attēli. |
| Kategorija | Jā | Var izvēlēties no saraksta. |
| Cena | Jā | Cenu diapazons ir 20-100000 eiro. |
| Nosaukums | Jā |  |
| Apraksts |  |  |

Apstrāde:

* Augšupielādētā attēla pārbaude uz formāta un izmēra atbilstību.
* Mākslas darba informācijas saglabāšana datu bāzē.

Izvaddati:

* Paziņojums par veiksmīgu augšupielādi vai kļūdas paziņojums.

**2.2.5. Lietotāja datu attēlošana profilā**

Ievaddati:

**4. tabula**

**Lietotāja datu attēlošana profilā**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| JWT token | Jā | Tiek paņemts no localStorage, lai autentificētu lietotāju. |

Apstrāde:

* Lietotāja autentifikācija, pamatojoties uz sniegto JWT tokenu.
* Informācijas paņemšana no datubāzes (Lietotājvārds, vārds, uzvārds, apraksts, lietotāja mājaslapa un sociālie tīkli, reģistrēšanas datums).

Izvaddati:

* Informācijas attēlošana profilā.

**2.2.6. Profila informācijas rediģēšana**

Ievaddati:

**5. tabula**

**Profila informācijas rediģēšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Vārds | Jā |  |
| Uzvārds | Jā |  |
| Mājaslapa |  |  |
| X (Twitter) |  |  |
| Instagram |  |  |
| Facebook |  |  |
| Apraksts |  |  |
| Attēls |  |  |

Apstrāde:

* Datu validācija.
* Informācijas atjaunošana datubāzē.

Izvaddati:

* Apstiprinājums par veiksmīgu profila rediģēšanu vai kļūdas paziņojums.

**2.2.7. Paroles maiņa**

Ievaddati:

**6. tabula**

**Paroles maiņa**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Veca parole | Jā |  |
| Jauna parole | Jā |  |
| Jauna parole atkārtoti | Jā |  |

Apstrāde:

* Datu validācija.
* Paroles šifrēšana.
* Informācijas atjaunošana datubāzē.

Izvaddati:

* Apstiprinājums par veiksmīgu paroles maiņu vai kļūdas paziņojums.

**2.2.7. Profila dzēšana**

Ievaddati:

**6. tabula**

**Profila dzēšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Lietotāja ID | Jā |  |

Apstrāde:

* Visa lietotāja informācija un profils tiek izdzēstās no datubāzes.
* Lietotājam tiek attēlots logs ar apstiprinājumu.

Izvaddati:

* Paziņojums par veiksmīgu profila dzēšanu vai kļūdas paziņojums (ja lietotāju ir izdzēsis administrators var moderators).

**2.2.8. Mākslas darbu attēlošana profilā**

Apstrāde:

* Lietotāja izveidotos mākslas darbus paņemšana no datubāzes.

Izvaddati:

* Lietotāja izveidotos mākslas darbus attēlošana profilā.

**2.2.9. Mākslas darba apskatīšana**

Ievaddati:

**7. tabula**

**Mākslas darbu apskatīšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Mākslas darba ID | Jā |  |

Apstrāde:

* Informācijas iegūšana par konkrētu mākslas darbu no datubāzes.

Izvaddati:

* Informācijas attēlošana par izvēlēto mākslas darbu (Visi attēli, cena, autors, izveidošanas laiks, cena, apraksts, kategorija).

**2.2.10. Mākslas darba dzēšana**

Ievaddati:

* Mākslas darba identifikācijas numurs

**8. tabula**

**Mākslas darba dzēšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Mākslas darba ID | Jā |  |

Apstrāde:

* Mākslas darbs tiek izdzēsts no datu bāzes.

Izvaddati:

* Paziņojums par veiksmīgu dzēšanu vai kļūdas paziņojums.

**2.2.11. Mākslas darbu rediģēšana**

Ievaddati:

**9. tabula**

**Mākslas darbu rediģēšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Māsklas darba ID | Jā |  |
| Attēli | Jā | Maksimāli 5 attēli. |
| Kategorija | Jā | Var izvēlēties no saraksta. |
| Cena | Jā |  |
| Nosaukums | Jā |  |
| Apraksts |  |  |

Apstrāde:

* Datu validācija.
* Dati par konkrētu mākslas darbu tiek atjaunoti datubāzē.

Izvaddati:

* Paziņojums par veiksmīgu rediģēšanau vai kļūdas paziņojums.

**2.2.12. Mākslas darbu kategorijas atlase**

Ievaddati:

* Lietotāja izvēlētā produkta kategorija.

Apstrāde:

* No datubāzes tiek paņemti darbi ar lietotāja izvēlēto kategoriju.

Izvaddati:

* Mākslas darbi tiek attēloti un lietotāji var viegli identificēt un pārlūkot produktus izvēlētajā kategorijā.

**2.2.14. Mākslas darbu kārtošana dilstoši pēc cenas**

Ievaddati:

* Mākslas darbu saraksts saistītajām cenām.

Apstrāde:

* Sistēma nodrošinās opciju sakārtot produktu sarakstu dilstošā secībā, pamatojoties uz cenām.
* Kad lietotājs izvēlas dilstošo kārtošanas opciju, sistēma sakārto produktus no augstākās līdz zemākai cenai.
* Kārtošanas process jāpiemēro dinamiski, nepārlādējot lapu.

Izvaddati:

* Lietotāja interfeiss (UI) jāatspoguļo atjauninātais produktu kārtojums, ļaujot lietotājiem viegli identificēt preces.

**2.2.15. Mākslas darbu kārtošana augoši pēc cenas**

Ievaddati:

* Mākslas darbu saraksts saistītajām cenām.

Apstrāde:

* Sistēma nodrošinās opciju sakārtot produktu sarakstu augošā secībā, pamatojoties uz cenām.
* Kad lietotājs izvēlas augošo kārtošanas opciju, sistēma sakārto produktus no zemākās līdz augstākai cenai.
* Kārtošanas process jāpiemēro dinamiski, nepārlādējot lapu.

Izvaddati:

* Lietotāja interfeiss (UI) jāatspoguļo atjauninātais produktu kārtojums, ļaujot lietotājiem viegli identificēt preces.

**2.2.16. Mākslas darba pievienošana**

Ievaddati:

* Lietotāja informācija par mākslas darbu(Attēls, cena, apraksts, nosaukums).

**10. tabula**

**Mākslas darbu pievienošana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Attēli | Jā | Maksimāli 5 attēli. |
| Kategorija | Jā | Var izvēlēties no saraksta. |
| Cena | Jā |  |
| Nosaukums | Jā |  |
| Apraksts | Jā |  |

Apstrāde:

* Statusa piešķiršana “Uploaded”
* Darbs tiek attēlots moderatoru sarakstā.

Izvaddati:

* Lietotājam tiek attēlots paziņojums par darba veiksmīgu pievienošanu un parbaudīšanu (statusu), var kļūdas paziņojums. Darbs tiek attēlots lietotāja profilā.

**2.2.17. Mākslas darbu atlasīšana pēc cenas intervāla**

Ievaddati:

* Mākslas darbu saraksts.
* Cenas intervāls.

Apstrāde:

* Datubāzē tiek atlasīts mākslas darbu saraksts, kuru cena iekļaujas lietotāja izvēlētajā intervālā.

Izvaddati:

* Lietotājam jāattēlo jaunu darbu sarakstu.

**2.2.18. Mākslas darba apstiprināšana**

Ievaddati:

* Mākslas darbs ar statusu “Uploaded”.

Apstrāde:

* Statusa piešķiršana “Verified”.
* Pievienošana datubāzē un publicēšana visiem lietotājiem.

Izvaddati:

* Atbilstošam moderatoram un lietotājam jāattēlo jaunu statusu.

**2.2.19. Mākslas darba atteikšana**

Ievaddati:

* Mākslas darbs ar statusu “Uploaded”.

Apstrāde:

* Statusa piešķiršana “Rejected”.
* Atteikšanas iemesla pievienošana no moderācijas puses.

Izvaddati:

* Atbilstošam moderatoram un lietotājam jāattēlo jaunu statusu un atteikšanas iemeslu.

**2.2.20. Mākslas darba pirkšana**

Ievaddati:

**11. tabula**

**Mākslas darba pirkšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Māsklas darba ID | Jā |  |
| Lietotāja ID | Jā |  |

Apstrāde:

* Tiek veikta transākcija.
* Izveidots pasūtījums un pasūtījuma ID.
* Pirkumam tiek pievienots statuss – “Uzsākts pirkuma process”.

Izvaddati:

* Jāattēlo pircējam un pārdedvējam paziņojumu par pirkuma uzsākšanu

**2.2.21. Mākslas darba pievienošana pirkumu grozā.**

Ievaddati:

**12. tabula**

**Mākslas darba pievienošana pirkumu grozā**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Māsklas darba ID | Jā |  |
| Lietotāja ID | Jā |  |

Apstrāde:

* Konkrēts darbs tiek paņemts no datubāzes un tiek attēlots lietotāja pirkumu grozā.

Izvaddati:

* Jāattēlo lietotājam pievienoto produktu pirkuma grozā.

**2.2.22. Mākslas darba “Like” funkcija.**

Ievaddati:

**13. tabula**

**Mākslas darba “Like” funkcija**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Māsklas darba ID | Jā |  |
| Lietotāja ID | Jā |  |

Apstrāde:

* Konkrēts darbs tiek paņemts no datubāzes un tiek attēlots lietotāja “Like” sadaļā.

Izvaddati:

* Jāattēlo lietotājam pievienoto produktu “Like” sadaļā.

**2.2.23. Moderatora tiesību piešķiršana lietotājam**

Ievaddati:

**14. tabula**

**Moderatora tiesību piešķiršana lietotājam**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Lietotāja ID | Jā |  |

Apstrāde:

* Lietotājam tiek piešķirtas jaunas moderatora tiesības (Tiek nomainīta lietotāja loma datubāzē).

Izvaddati:

* Apstiprinājums par veiksmīgu tiesību piešķiršanu vai kļūdas paziņojums.

### 2.2.24. Pasūtījuma attēlošana moderatoriem

Ievaddati:

**15. tabula**

**Pasūtījuma attēlošana moderatoriem**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Pasūtījuma ID | Jā |  |

Apstrāde:

* Moderators saņema informāciju no datubāzes par jaunu pasūtījumu.

Izvaddati:

* Informācija par jaunu pasūtījumu tiek attēlota moderatoru lapā.

**2.2.25. Pasūtījuma statusa mainīšana**

Ievaddati:

**16. tabula**

**Pasūtījuma statusa mainīšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Pasūtījuma ID | Jā |  |

Apstrāde:

* Tiek nomainīts pasūtījuma statuss uz “Nosūtīts”, “Piegādāts” vai “Atcelts” un dati tiek atjaunoti datubāzē.

Izvaddati:

* Paziņojums par veiksmīgo statusa mainīšanu vai kļūdas paziņojums.
* Jāatjauno pasūtījuma statusu pircējam.

**2.2.25. Pasūtījuma atcelšana**

Ievaddati:

**16. tabula**

**Pasūtījuma statusa mainīšana**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Pasūtījuma ID | Jā |  |

Apstrāde:

* Tiek nomainīts pasūtījuma statuss uz “Atcelts” un dati tiek atjaunoti datubāzē. Pasūtījums vairs nav derīgs un nav iespējams ar viņu darboties. Pasūtījumu var atcelt, tikai tad, ja statuss ir “Uzsākts pirkuma process”.

Izvaddati:

* Paziņojums par veiksmīgo pasūtījuma atcelšanu vai kļūdas paziņojums.
* Jāatjauno pasūtījuma statusu pircējam un moderatoriem.

**2.2.26. Moderatora tiesību atņemšana lietotājam**

Ievaddati:

* Lietotāja ID.

**17. tabula**

**Moderatora tiesību atņemšana lietotājam**

|  |  |  |
| --- | --- | --- |
| **Nosaukums** | **Obligāts** | **Piezīmes** |
| Lietotāja ID | Jā |  |

Apstrāde:

* Lietotājam tiek noņemtas moderatora tiesības (Tiek nomainīta lietotāja loma datubāzē).

Izvaddati:

* Apstiprinājums par veiksmīgu tiesību noņemšanu vai kļūdas paziņojums.

**2.2.26. “Atcerēties mani” funkcija**

Ievaddati:

* Lietotāja lietotājvārds un parole. (Pareizi dati).
* Tiek nospiesta poga “Remember me”.

Apstrāde:

* Sistēma ģenerē ilglaicīgu JWT.
* Šis žetons tiek saglabāts localStorage.

Izvaddati:

* Turpmākajos lietojumprogrammas apmeklējumos sistēma automātiski nosaka autentifikācijas žetona klātbūtni.
* Ja žetons ir derīgs, lietotājam tiek piešķirta automātiska piekļuve bez atkārtotas akreditācijas datu ievadīšanas.

**2.2.27. Lietotāju attēlošana moderatora lapā**

Ievaddati:

* Tiek atvērta sadaļa “Users” moderatora lapā.

Apstrāde:

* Visi dati par visiem lietotājiem tiek paņemti no datubāzes.

Izvaddati:

* Jāattēlo datus moderatoram.

**2.2.28. Attēlošana populārākos māksliniekus pēc pārdošanas apjoma (Top)**

Ievaddati:

* Lietotāju saraksts

Apstrāde:

* Lietotāji tiek sakārtoti dilstoši pēc pārdošanas apjoma.
* No datubāzes tiek paņemti 5 lietotāji ar vislielāko pārdošanas apjomu.

Izvaddati:

* Jāattēlo 5 populārākos māksliniekus galvēnajā lapā.

## 2.3. Sistēmas nefunkcionālās prasības

**2.3.1. Veiktspēja**

* Platformai jābūt spējīgai apstrādāt vienlaikus notiekošas lietotāju darbības bez būtiskas aizkaves. Minimāli 10.

**2.3.2. Drošība**

* Lietotāju datiem, tostarp personiskā informācija un maksājumu detaļas, jābūt šifrētām, lai nodrošinātu datu drošību.
* Platformai jāievēro nozares standarta šifrēšanas protokoli, un drošības pārbaudes jāveic gadu gaitā.

**2.3.3. Lietojamība**

* Platformai jābūt intuitīvai un lietotājam draudzīgai saskarnei, nodrošinot pozitīvu lietotāja pieredzi.

**2.3.4. Responsīvs (pielāgojams) dizains**

* Platformai jāpielieto pielāgojama dizaina principi, nodrošinot optimālas lietotāja pieredzes visās ierīcēs, tostarp galddatoros, planšetdatoros un viedtālruņos.
* Vietnei jāpielāgojas dažādu ekrāna izmēru un izšķirtspēju prasībām, nodrošinot konsistentu un vizuāli pievilcīgu pieredzi visās ierīcēs.

**2.3.5. Sadarbība ar dažādām pārlūkprogrammām**

* Platformai jābūt saderīgai ar plašu pārlūku klāstu, iekļaujot Google Chrome, Mozilla Firefox, un Microsoft Edge, nodrošinot līdzvērtīgu lietotāja pieredzi visās šajās pārlūkprogrammās.

**2.3.6. Ātrgaitas pārejas starp lappusēm**

* Platformai jānodrošina ātra navigācija un pārejas starp dažādām lappusēm, lai lietotājiem nebūtu jāgaida ilgi uz iekšējām lappusēm vai funkcijām. Ne ilgāk nekā 1 sekunde.

**2.3.7. Izstrādes valoda un tehnoloģijas**

* Izvēlētajai izstrādes valodai un tehnoloģijām jābūt modernām un atbilstošām, lai nodrošinātu sistēmas ilgtermiņa uzturējamību un paplašināmību.
* Jānodrošina regulāra izstrādes rīku un tehnoloģiju atjaunināšana, lai izmantotu jaunākās iespējas un novērstu drošības problēmas.

## 2.4. Gala lietotāja raksturiezīmes

Gala lietotāja raksturiezīmes "Artify" projektam ietver dažādu cilvēku grupu, tai skaitā viesus, lietotājus(pircējs/pārdevējs), moderatorus un administratoru. Platforma ir izstrādāta, lai apmierinātu katra lietotāju segmenta vajadzības un preferences, nodrošinot pielāgotu un lietotājam draudzīgu pieredzi. Šeit ir konkrētas īpašības un iespējas katrai galējā lietotāju grupai:

**Viesis (Neautorizēts lietotājs):**

* Var piereģistrēties un pieteikties.
* Var apskatīt mākslas darbus sākumlapā vai cita lietotāja profilā.
* Var apskatīt citu lietotāju profilus.

**Lietotājs (Pircējs/Pārdevējs):**

Var izmantot visas lietotnes funkcijas (izņemot moderatora un administratora funkcionālitāti):

* Var izrakstities.
* Var apskatīt mākslas darbus sākumlapā vai cita lietotāja profilā.
* Var pievienot mākslas darbus grozā vai “Like” sadaļā.
* Var izveidot pasūtījumu un apskatīt savus pasūtījumus.
* Var apskatīt savu vai cita lietotāja profiku, rediģēt sava profila un konta datus.
* Var filtrēt un kārtot mākslas darbus.
* Var pievienot mākslas darbus pārdošanai.
* Var sekot citiem lietotājiem.

**Moderators:**

Sistēmā var būt vairāki moderatori,viņi izmantot visas Lietotāja (Pircējs/Pārdevējs) funkcijas, papildus moderatoriem ir īpašas funkcijas:

* Var apskatīt lietotāju reģistru.
* Var rediģēt, dzēst lietotāju datus un lomu.
* Var apskatīt mākslas darbu reģistru.
* Var rediģēt, dzēst mākslas darbu datus un verificēt tos.
* Var apskatīt pasūtījumus un rediģēt statusu.

**Administrators:**

Var izmantot visas Lietotāja (Pircējs/Pārdevējs) un Moderatora funkcijas, bet administrators sistēmā ir tikai viens:

* Var piešķirt vai atņemt moderatora tiesības lietotājam.

# 3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums

Šajā sadaļā ir aprakstītas izvēlētās programmēšanas valodas un izstrādes rīki. Tajā aprakstīti arī alternatīvi risinājumi, kurus varēja izmantot.

## 3.1. Izvēlēto risinājuma līdzēkļu un valodu apraksts

Šajā sadaļa ir aprakstītas izvēlētās programmēšanas valodas un izstrādes rīki.

### 3.1.1. Front-End tehnoloģijas

**HTML:**

* **Par tehnoloģiju:** HTML kalpo kā pamats tīmekļa lapu struktūrēšanai, definējot izkārtojumu, virsrakstus, formas un citus būtiskus komponentus, kas nepieciešami lietotāja saskarnes veidošanai.
* **Pamatojums:** HTML tiek izvēlēts tā fundamentālās nozīmes dēļ tīmekļa izstrādē un tā spējas nodrošināt standartizētu struktūru tīmekļa saturam. Marketplace projektā HTML ir būtisks, lai izveidotu produktu sarakstus, lietotāju profilus un citus interfeisa elementus. Lai gan ir alternatīvas, piemēram, JSX (React) vai šablona dzinēji, piemēram, Handlebars, HTML tiek izvēlēts tā vienkāršības, lasāmības un plašās atbalsta dēļ. Marketplace projektā, kur saturam jābūt viegli pieejamam un saprotamam gan attīstītājiem, gan meklētājprogrammām, HTML piedāvā vienkāršu risinājumu, neiekļaujot nepieciešamo sarežģītību.

**JavaScript:**

* **Par tehnoloģiju**: JavaScript ir daudzpusīga programmēšanas valoda, kas tiek izmantota gan priekšējās, gan aizmugurējās puses izstrādei. Tā ļauj izveidot dinamiskas un reaģējošas lietotāja saskarnes (piemēram, React), kā arī servera puses loģiku (izmantojot Node.js un Express).
* **Pamatojums**: JavaScript tiek izvēlēts tā vienotas valodas un plašās kopienas atbalsta dēļ, kas padara to par ideālu izvēli gan priekšējās, gan aizmugurējās puses izstrādei marketplace platformai. Tas nodrošina vienkāršu koplietošanu kodā un paplašinājumos, kā arī vienotu sintaksi abās lietotāja saskarnes un servera puses kodā. Salīdzinot ar citām programmēšanas valodām, piemēram, Python vai Ruby, JavaScript ir populāra un plaši izplatīta, ar lielu kopienas atbalstu un daudzām bibliotēkām un rīkiem. Tā kā projektā tiek izmantots React front-endā un Node.js un Express back-endā, JavaScript izmantošana vienotas valodas dēļ vienkāršo izstrādes procesu un nodrošina efektīvu datu pārvietošanu starp priekšējo un aizmugurējo pusi, padarot to par labu izvēli marketplace platformai.

**React:**

* **Par tehnoloģiju:** React ir front-end bibliotēka, kas paredzēta lietotāja saskarnes veidošanai, piedāvājot komponentu bāzētu arhitektūru, lai izveidotu pārstrādājamus un interaktīvus UI komponentus.
* **Pamatojums:** React tiek izvēlēts tā spējas dēļ izveidot dinamiskas un reaģējošas lietotāja saskarnes, veicinot modernu un interaktīvu marketplace platformas izveidi. Tā komponentu bāzētā pieeja veicina koda atkārtotību, padarot to vieglāk uzturēt un skalot aplikāciju. Lai gan ir alternatīvas, piemēram, Angular vai Vue.js, React tiek izvēlēts tā vienkāršības, veiktspējas un spēcīgā ekosistēmas dēļ. Marketplace projektā, kur lietotāja pieredze un interaktivitāte ir galvenā, React virtuālais DOM un komponentu bāzētā arhitektūra nodrošina būtiskus priekšrocības veiktspējas un attīstības efektivitātes ziņā.

**Tailwind CSS:**

* **Par tehnoloģiju:** Tailwind CSS ir utilītu pirmā CSS struktūra, kas vienkāršo stilizācijas procesu, nodrošinot iepriekš definētas utilītu klases parastajiem CSS īpašību.
* **Pamatojums:** Tailwind CSS tiek izvēlēts tā lietošanas viegluma un uzturēšanas dēļ, ļaujot ātri stilizēt React komponentus, izmantojot iepriekš definētas utilītu klases. Marketplace projektā, kur konsistences un skalējamības nodrošināšana ir būtiska, Tailwind CSS veicina vienotas dizaina sistēmas izveidi ar minimālu piepūli. Lai gan ir CSS ietvaru, piemēram, Bootstrap vai Foundation, Tailwind CSS piedāvā elastīgāku un pielāgojamāku pieeju stilizēšanai. Atšķirībā no tradicionālajiem CSS ietvariem, kas balstās uz iepriekš definētām komponentēm un stiliem, Tailwind CSS ļauj iegūt lielāku kontroli pār stilizāciju, padarot dizainu piemērotāku konkrētajiem platformas prasījumiem.

**AG-Grid:**

* **Par tehnoloģiju**: AG-Grid ir tabulu komponente, kas tiek izmantota manā projektā, lai attēlotu datus moderatoriem par lietotājiem, mākslas darbiem un pasūtījumiem. Tas nodrošina iespēju efektīvi pārvaldīt un filtrēt lielu datu apjomu, vienlaikus piedāvājot dažādas funkcijas, piemēram, kārtošanu, filtrēšanu un grupēšanu.
* **Pamatojums**: AG-Grid ir izvēlēts, lai nodrošinātu elastīgu un uzlabotu datu attēlošanas risinājumu mājaslapas administratoriem un moderatoriem. Tas ļauj ērti pārvaldīt lielu datu apjomu, kas nepieciešams moderēšanai un darbību veikšanai mājaslapā. Salīdzinot ar citām tabulu komponentēm, AG-Grid piedāvā plašu funkciju klāstu un izcilu veiktspēju, pat ar lielu datu apjomu. Tas ļauj dinamiski pārvaldīt un attēlot datus, nodrošinot iespēju viegli pielāgot tabulas saskaņā ar specifiskajām vajadzībām un prasībām. Tas padara to par ideālu risinājumu marketplace vietnei, kur ir nepieciešama efektīva datu pārvaldība un attēlošana.

### 3.1.2. Back-End tehnoloģijas

**Node.js:**

* **Par teholoģiju**: Node.js ir servera puses JavaScript izpildmehānisms, kas ļauj izpildīt JavaScript kodu serverī. Tas nodrošina servera darbību, apstrādā HTTP pieprasījumus, izpilda servera puses loģiku un pārvalda kopējo lietotnes back-end daļu.
* **Pamatojums**: Node.js tiek izvēlēts tā vienkāršības un elastības dēļ, ļaujot izstrādātājiem veidot efektīvas un mērojamās tīmekļa lietojumprogrammas, izmantojot vienu valodu gan priekšējās, gan aizmugurējās puses izstrādei. Salīdzinot ar citām back-end tehnoloģijām, piemēram, PHP vai Ruby on Rails, Node.js piedāvā lielāku veiktspēju un elastību, jo tas izmanto asinhrono programmatūru un nebloķējošus ievades/izejās datu procesus. Tā kā projektā tiek izmantots arī JavaScript front-endā, Node.js izmantošana vienotas valodas dēļ vienkāršo izstrādes procesu un atvieglo kodu koplietošanu starp priekšējo un aizmugurējo pusi.

**Express:**

* **Par tehnoloģiju**: Express ir tīmekļa lietojumprogrammu struktūra Node.js, kas vienkāršo uzticamu API izveidi. Tas definē API maršrutus, starpprogrammatūru un apstrādā HTTP pieprasījumus, nodrošinot vienkāršotu pieeju efektīviem un mērojamajiem back-end pakalpojumiem.
* **Pamatojums**: Express tiek izvēlēts tā minimalistiskās un elastīgās dēļ, kas ļauj ātri izveidot un pielāgot back-end loģiku atbilstoši konkrētajām projektu prasībām. Tā viegli saprotamā struktūra un liela kopienas atbalsts padara to par ideālu izvēli marketplace platformai. Salīdzinot ar citām back-end struktūrām, piemēram, Django vai Ruby on Rails, Express piedāvā lielāku elastību un kontrolē par izstrādes procesu. Tas ir viegli pielāgojams un paplašināms, ļaujot izveidot API, kas atbilst konkrētajiem projektu prasībām un veicina produktivitāti.

**MongoDB:**

* **Par tehnoloģiju**: MongoDB ir NoSQL datu bāze, kas piedāvā elastīgu un mērojamu datu uzglabāšanas risinājumu, uzglabājot datus JSON līdzīgā formātā. Tas nodrošina vienkāršu mērojamību un pielāgojamību mainīgiem datu struktūrām.
* **Pamatojums:** MongoDB tiek izvēlēts tā elastīguma un skalējamības dēļ, kas ir būtiski, lai nodrošinātu, ka marketplace platforma var ātri pielāgoties mainīgajām. Salīdzinot ar tradicionālajām ER datu bāzēm, piemēram, MySQL vai PostgreSQL, MongoDB piedāvā elastīgāku datu modelēšanu un lielāku skalējamību, ļaujot ātri pielāgoties mainīgajām prasībām un saglabāt izstrādes procesu elastīvu.

**Mongoose:**

* **Par tehnoloģiju:** Mongoose ir objektu datu modelēšanas (ODM) bibliotēka MongoDB un Node.js, kas nodrošina vienkāršu veidu, kā mijiedarboties ar MongoDB datubāzēm, izmantojot JavaScript objektus.
* **Pamatojums:** Mongoose tiek izvēlēts, lai atvieglotu un standartizētu mijiedarbību ar MongoDB datubāzi, piedāvājot skaistu ORM sistēmu, kas padara datu operācijas vienkāršākas un izpratnējākas. Salīdzinot ar tīrām MongoDB operācijām, Mongoose piedāvā augstāku līmeni abstrakcijas, kas atvieglo datu modeļu definēšanu un datu manipulēšanu, vienlaikus nodrošinot drošību un izpildītspēju. Tas padara Mongoose par labu izvēli projektam, kur nepieciešama viegli izpratne un uzturēšana ilgtermiņā.

### 3.1.3. Papildus back-end rīki

**Bcrypt:**

* **Par tehnoloģiju**: Bcrypt ir kriptogrāfijas bibliotēka, kas paredzēta parolēm drošai saglabāšanai, izmantojot hešu. Tā aizsargā lietotāju paroles, hešojot un droši verificējot tās, pievienojot drošības slāni lietotāju autentifikācijai.
* **Pamatojums**: Bcrypt tiek izvēlēts kā svarīgs drošības elements platformā, nodrošinot, ka lietotāju paroles ir droši saglabātas un aizsargātas pret nelikumīgu piekļuvi. Salīdzinot ar citām kriptogrāfijas bibliotēkām, piemēram, SHA-256 vai MD5, Bcrypt tiek uzskatīts par drošāku risinājumu, jo tas izmanto vienmērīgu soli un ir izstrādāts, ņemot vērā drošības prasības, piemēram, lēnu hash funkciju. Tas padara to par labu izvēli projektam, kuram nepieciešams augsts drošības līmenis.

**Multer:**

* **Par tehnoloģiju**: Multer ir starpvaloda, kas apstrādā failu augšupielādes Node.js. Tas apstrādā un saglabā lietotāju augšupielādētus profila attēlus, mākslas darbu attēlus un citus failus, nodrošinot bezproblēmu failu apstrādes risinājumu.
* **Pamatojums**: Multer tiek izvēlēts kā efektīvs un uzticams risinājums failu augšupielādei platformā. Tas nodrošina, ka lietotāji varēs viegli augšupielādēt savus attēlus un failus bez sarežģītām procedūrām. Salīdzinot ar citām failu augšupielādes bibliotēkām Node.js, piemēram, Formidable vai Busboy, Multer ir vienkāršs lietojumprogrammas interfeiss un piedāvā plašas konfigurācijas iespējas. Tas ir viegli pielāgojams mūsu projektam un nodrošina efektīvu failu apstrādi, kas ir būtiska funkcionalitāte marketplace platformai.

**Yup:**

* **Par tehnoloģiju**: Yup ir JavaScript shēmu veidotājs, kas paredzēts vērtību parsēšanai un validācijai. Tas nodrošina datu integritāti, validējot un attīrot lietotāja ievadi servera pusē, novēršot nepareizu vai ļaunprātīgu datu iekļūšanu sistēmā.
* **Pamatojums**: Yup tiek izvēlēts kā svarīgs rīks platformā, lai nodrošinātu datu integritāti un drošību, validējot lietotāja ievadi. Tas palīdz novērst datu kļūdas un ļaunprātīgu datu iekļūšanu sistēmā, tādējādi uzlabojot kopējo drošības līmeni. Salīdzinot ar citiem validācijas un shēmu veidotājiem JavaScript, piemēram, Joi vai Validator.js, Yup tiek uzskatīts par viegli pielāgojamu un elastīgu risinājumu. Tas piedāvā intuitīvu sintaksi un plašas validācijas iespējas, kas padara to par labu izvēli projektam, kuram nepieciešama efektīva un uzticama datu validācija un attīrīšana.

### 3.1.4. Pārlūkprogramma

**Chrome:**

* **Par tehnoloģiju**: Chrome ir tīmekļa pārlūkprogramma, kurā tiek izpildīta tīmekļa lietojumprogramma. To izmanto, lai testētu, labotu kļūdas un nodrošinātu saderību ar klienta pusi.
* **Pamatojums**: Chrome tiek izvēlēts kā galvenais pārlūkprogrammas tests un kļūdu labošanas instruments platformai. Tā ir viena no vispopulārākajām pārlūkprogrammām un nodrošina plašas iespējas attīstītājiem, lai veiktu testēšanu un atklātu jebkādas kļūdas vai saderības problēmas. Lai gan ir pieejami citi pārlūkprogrammu testēšanas rīki, Chrome ir populārākais un plaši izmantotais pārlūkprogrammas risinājums. Tas piedāvā plašas rīku klāstu attīstītājiem, ieskaitot iebūvētu atkļūvošanu, konsoles rīkus un papildinājumus, kas palīdz uzlabot produktivitāti un efektivitāti testēšanas procesā. Tāpēc Chrome ir laba izvēle projektam, lai nodrošinātu optimālu saderību un veiksmīgu lietojumprogrammas darbību visās pārlūkprogrammās.

### 3.1.5. Koda redaktors un versionēšana

**Visual Studio Code:**

* **Par tehnoloģiju**: Visual Studio Code ir integrēta izstrādes vide, kas piedāvā plašu rīku klāstu, lai atvieglotu koda rakstīšanu, testēšanu un atkļūvošanu. Tā piedāvā ērtu un elastīgu darba vidi programmētājiem, nodrošinot papildu atbalstu paplašinājumu formātā, iespējas rediģēt kodu, integrētu atkļūvošanu un citus produktivitātes uzlabojumus.
* **Pamatojums**: Visual Studio Code tiek izvēlēts kā galvenais koda redaktors projekta izstrādei. Tas piedāvā plašas iespējas programmētājiem, lai veiktu koda rakstīšanu un atkļūvošanu, kā arī nodrošina papildu atbalstu dažādu programmēšanas valodu atbalstam un paplašinājumiem, kas padara darbu ar kodu ērtu un efektīvu. Lai gan ir pieejami citi koda redaktori, Visual Studio Code izceļas ar plašu funkciju klāstu, elastīgu konfigurāciju un aktīvu attīstības kopienu. Tas piedāvā integrētu atbalstu dažādām programmēšanas valodām, labu atkļūvošanas rīku un daudzus papildinājumus, kas padara to par ērtu un efektīvu rīku izstrādātājiem visās nozarēs.

**GitHub:**

* **Par tehnoloģiju**: GitHub ir platforma, kas nodrošina versiju kontroli un sadarbību kodu izstrādē. Tas ļauj izstrādātājiem kopīgot, atkļūdot un uzlabot savus projektus, nodrošinot centralizētu vietu kodu uzglabāšanai, versiju izsekošanai un sadarbībai starp izstrādātājiem, izmantojot git versiju kontroles sistēmu.
* **Pamatojums**: GitHub tiek izvēlēts kā galvenais versiju kontroles un sadarbības rīks projektam. Tā ļauj efektīvi pārvaldīt kodu, izsekojot versijas un veicot izmaiņas kopīgās izstrādes procesā. GitHub ir viena no populārākajām un plaši izmantotajām platformām kodu kopīgai, izsekošanai un sadarbībai. Tā piedāvā plašas iespējas projektu pārvaldībai, labu integrāciju ar citām izstrādes rīkiem un labu atbalstu lietotājiem. Tāpēc GitHub ir ideāls risinājums projektam, lai nodrošinātu efektīvu kodu pārvaldību.

### 3.1.6. Izstrādes un kvalitātes rīki

**Nodemon:**

* **Par tehnoloģiju**: Nodemon ir utilīta, kas uzrauga Node.js lietojumprogrammu izmaiņas un automātiski restartē serveri. Tā paātrina izstrādes procesu, izslēdzot nepieciešamību pēc manuāla servera restartēšanas pēc koda izmaiņām, uzlabojot izstrādātāja produktivitāti.
* **Pamatojums:** Nodemon tiek izvēlēts, lai uzlabotu izstrādes procesu un atvieglotu servera puses izmaiņu ieviešanu. Tas palīdz izstrādātājiem ātri redzēt izmaiņas un veikt eksperimentus bez manuālas servera restartēšanas. Salīdzinot ar citām līdzīgām utilītām, Nodemon ir vienkāršs un ērts rīks, kas piedāvā automātisko servera restartēšanu pēc izmaiņām. Tas padara izstrādes procesu efektīvāku un ērtāku, tāpēc tas ir ideāls risinājums projektam.

**ESlint:**

* **Par tehnoloģiju**: ESlint ir statiska koda analīzes rīks, kas identificē un palīdz novērst problēmas JavaScript kodā. Tas izpilda kodu formatēšanas standartus, atklāj potenciālās problēmas un uztur kodu kvalitāti, nodrošinot ievērošanu labāko prakšu ziņā.
* **Pamatojums**: ESlint tiek izvēlēts, lai uzlabotu koda kvalitāti un palīdzētu identificēt un novērst potenciālās problēmas. Tas ir svarīgs rīks, lai nodrošinātu vienotu kodu kvalitāti visā projektā. Salīdzinot ar citām statiskās koda analīzes rīkiem, ESlint ir viens no populārākajiem un plaši izmantotajiem, kas piedāvā plašas konfigurācijas iespējas un labu atbalstu JavaScript kodam. Tas padara to par ideālu risinājumu projektam.

**dotenv:**

* **Par tehnoloģiju**: Dotenv ir rīks, kas pārvalda vides mainīgos, ielādējot tos no .env faila process.env. Tas nodrošina ērtu un drošu veidu, kā konfigurēt vides mainīgos, API atslēgas, datu bāzes pieslēguma virknes, neizpaustot jutīgo informāciju kodā.
* **Pamatoujms**: Dotenv tiek izvēlēts, lai nodrošinātu drošu un ērtu vides mainīgo konfigurēšanu projektā. Tas palīdz izvairīties no jutīgas informācijas izpausmes un vienkāršo kodu konfigurācijas procesu. Lai gan ir citi veidi, kā pārvaldīt vides mainīgos, dotenv ir vienkāršs un plaši izmantots rīks, kas nodrošina drošu un ērtu vides mainīgo konfigurēšanu. Tas padara to par populāru risinājumu daudziem projektu izstrādātājiem.

**Prettier:**

* **Par tehnoloģiju**: Prettier ir kodu formatēšanas rīks, kas ievieš vienotu koda stilu. Tas automātiski formatē kodu, lai ievērotu iepriekš noteiktu stilu ceļvedi, uzlabojot kodu lasāmību un uzturot vienotu kodu visā projektā.
* **Pamatojums:** Prettier tiek izvēlēts, lai nodrošinātu vienotu kodu stilu un uzlabotu kodu lasāmību. Tas palīdz saglabāt koda kvalitāti un vienotu izskatu visā projektā. Salīdzinot ar citiem kodu formatēšanas rīkiem, Prettier ir viens no populārākajiem un plaši izmantotajiem. Tas piedāvā ērtu konfigurāciju un spēj automātiski pielāgot kodu atbilstoši iepriekš noteiktajiem stilu ceļvežiem. Tas padara to par ērtu un efektīvu risinājumu, lai nodrošinātu vienotu kodu stilu visā projektā.

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

Šajā sadaļa ir aprakstītas alternatīvas programmēšanas valodas un izstrādes rīki, kurus varēja izmantot.

### 3.2.1 Alternatīvas Front-End tehnoloģijas

**Alternatīva HTML:**

* **HTML:** HTML nodrošina struktūras marķējumu tīmekļa lapām, bet tas var grūti uzturams, it īpaši, ja ir daudz atkārtojošu elementu.
* **Pug:** Pug ar savu īso un lasāmo sintaksi atvieglo HTML struktūras izveidi. Tas padara kodu pārskatāmāku un mazāk jāraksta, kas ir noderīgi marketplace projektos ar daudzām lapām un sadaļām.

**Alternatīva Tailwind CSS:**

* **Tailwind CSS:** Tailwind CSS piedāvā lielu elastību un precīzu kontroli pār stilu, kas ir noderīgi, ja nepieciešams pielāgotas un unikālas dizaina sistēmas. Tas var būt īpaši noderīgi marketplace veida projektos, kur ir nepieciešama diferenciācija no citiem konkurentiem, nodrošinot unikālu un vizuāli pievilcīgu lietotāja pieredzi.
* **Bootstrap:** Bootstrap ir iepriekš definēta stila bibliotēka, kas piedāvā gatavus dizaina elementus un komponentus. Tas var būt noderīgs projektos, kur ir nepieciešams ātri ieviest dizainu, bet var ierobežot pielāgošanas iespējas, īpaši, ja unikāls dizains nav tik svarīgs projektam.

**Alternatīva React:**

* **React:** React piedāvā elastīgu un efektīvu veidu, kā pārvaldīt lietojumprogrammas stāvokli un veidot dinamiskas lietotāja saskarnes. Tas ir īpaši noderīgs marketplace veida projektos, kur ir nepieciešams dinamisms un liela datu apstrāde.
* **Angular:** Angular ir pilnīga struktūra, kas nodrošina visu nepieciešamo funkcionalitāti vienā ietvarā. Tas var būt noderīgs, ja ir nepieciešama integrēta risinājuma vienkāršība un pilnvaras.
* **Vue.js:** Vue.js piedāvā viegli izmantojamu un saprotamu struktūru, kas var būt laba izvēle, ja ir nepieciešama vienkārša un viegli pielāgojama tehnoloģija, īpaši mazākos vai vidējos marketplace veida projektos.

**Alternatīva JavaScript:**

* **JavaScript:** JavaScript ir neaizstājams marketplace veida projektos, jo tas nodrošina elastīgu un ātru izstrādi. Tomēr, tas var radīt problēmas lielākos projektos ar kodu bāzi.
* **TypeScript:** TypeScript piedāvā papildu drošības un efektivitātes līmeņus, kas var uzlabot koda uzturēšanu un veiktspēju. Tas ir noderīgi lielos projektos ar lielu kodu bāzi, kur ir nepieciešama labāka kodu pārvaldība un uzraudzība.

**Alternatīva AG-Grid:**

* **AG-Grid:** AG-Grid ir spēcīga, elastīga un viegli pielāgojama tabulu bibliotēka, kas ļauj efektīvi attēlot un pārvaldīt datus. Tas ir ideāls marketplace veida projektos, kur ir nepieciešama augsta datu attēlošanas veiktspēja, elastīgums un interaktivitāte.
* **React Table:** React Table ir alternatīva tabulu bibliotēka, kas piedāvā līdzīgu funkcionalitāti kā ag-Grid. Tomēr AG-Grid parasti ir priekšplānā ar plašākām iespējām, labāku veiktspēju un detalizētāku dokumentāciju, kas padara to par labāku izvēli marketplace veida projektos, kur ir nepieciešama augsta kvalitāte un pielāgojamība.

### 3.2.2. Alternatīvas Back-End tehnoloģijas

**Alternatīva Node.js:**

* **Node.js:** Node.js ir servera puses JavaScript izpildmehānisms, kas nodrošina ātru un elastīgu servera izstrādi. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama augsta veiktspēja un elastīga arhitektūra, piemēram, vairāku datu avotu integrācijai.
* **Django:** Django ir Python pamatota servera puses ietvara, kas nodrošina augstu produktivitāti un drošību. Tas var būt noderīgs marketplace veida projektos, kur ir nepieciešama integrēta autorizācija un lietotāju pārvaldība, piemēram, lietotāju autentifikācijai un autorizācijai.

**Alternatīva Express**:

* **Express:** Express ir minimalistisks un elastīgs Node.js ietvars, kas nodrošina vienkāršu API izveidi un maršrutēšanu. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama ātra un efektīva back-end izstrāde, piemēram, lietotāju informācijas apmaiņai un datu apstrādei.
* **Flask:** Flask ir Python pamatots mikro ietvars, kas piedāvā vienkāršu un pārskatāmu veidu, kā izveidot web lietojumprogrammas. Tas var būt noderīgs marketplace veida projektos, kur ir nepieciešams mazs un viegli uzturams back-end, piemēram, lietotāju darbu apmaiņai un datu uzglabāšanai.

**Alternatīva MongoDB:**

* **MongoDB:** MongoDB ir NoSQL datu bāze, kas nodrošina elastīgu un mērojamu datu uzglabāšanu. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama ātra un pielāgojama datu apstrāde, piemēram, lietotāju datu uzglabāšanai un mākslas darbu informācijai.
* **PostgreSQL:** PostgreSQL ir attiecīgo datu bāžu sistēma, kas nodrošina attiecīgo datu modelēšanu un tranzakciju atbalstu. Tas var būt noderīgs marketplace veida projektos, kur ir nepieciešama augsta datu konsistence un attiecīgu datu struktūru saglabāšana, piemēram, pasūtījumu vai darbu informācijai.

**Alternatīva Mongoose:**

* **Mongoose:** Mongoose ir MongoDB un Node.js bibliotēka, kas nodrošina vienkāršu mijiedarbību ar MongoDB, izmantojot JavaScript objektus. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešams vienkāršs un elastīgs datu modelēšanas risinājums.
* **TypeORM:** TypeORM ir TypeScript un JavaScript ORMapper, kas atvieglo datu bāzes mijiedarbību. Tas piedāvā tīru un deklaratīvu veidu, kā definēt datu modeļus un veikt datu bāzes darbības. Tas var būt noderīgs lielos veida projektos, kur ir nepieciešama augsta datu modeļu abstrakcija un labāka izpildīšanas drošība.

### 3.2.3. Papildus Back-End rīki

**Alternatīva bcrypt**:

* **bcrypt:** bcrypt ir kriptogrāfijas bibliotēka parolēm drošai saglabāšanai, izmantojot hešu. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama droša paroļu saglabāšana un autentifikācijas drošība.
* **Argon2:** Argon2 ir jaunākā paroles kriptogrāfijas bibliotēka, kas piedāvā vēl augstāku drošības līmeni nekā bcrypt. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama palielināta paroles drošība un izturība pret uzbrukumiem.

**Alternatīva Multer:**

* **multer:** multer ir Node.js pakalpojums, kas apstrādā failu augšupielādes. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama efektīva un droša failu augšupielādes apstrāde.
* **Formidable:** Formidable ir alternatīvs Node.js pakalpojums, kas piedāvā līdzīgu funkcionalitāti failu apstrādes jomā. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama vienkārša un uzticama failu augšupielādes apstrāde.

**Alternatīva Yup:**

* **Yup:** Yup ir JavaScript shēmas veidotājs vērtību parsēšanai un validācijai. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama datu integritātes nodrošināšana un lietotāja ievades validācija.
* **Joi:** Joi ir alternatīvs shēmas veidotājs, kas piedāvā līdzīgu funkcionalitāti yup. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama vērtību validācija un datu pārbaude.

### 3.2.4. Pārlūkprogramma

**Alternatīva Chrome:**

* **Chrome:** Chrome ir populāra tīmekļa pārlūkprogramma, kas nodrošina plašu funkcionalitāti un ātru veiktspēju. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešams ātrs un uzticams tīmekļa pārlūkošanas risinājums, piemēram, lietotāju saskarnes testēšanai un kļūdu labošanai.
* **Firefox:** Firefox ir cienījama tīmekļa pārlūkprogramma, kas piedāvā plašu papildu funkciju un atvērtu avotu kodu. Tas var būt noderīgs marketplace veida projektos, kur ir nepieciešams atklāts un drošs tīmekļa pārlūkošanas risinājums, piemēram, datu privātuma un drošības apsvērumu dēļ.

### 3.2.5. Koda redaktors un versionēšana

**Alternatīva Visual Studio Code:**

* **Visual Studio Code:** Visual Studio Code ir populāra integrēta izstrādes vide, kas piedāvā plašu rīku klāstu un augstu papildinājumu atbalstu. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama plaša rīku izvēle un papildu atbalsts, piemēram, koda rediģēšanas iespējas un integrēta atkļūvošana.
* **Atom:** Atom ir atvērtā koda koda redaktors, kas piedāvā pielāgojamu un plašu papildinājumu klāstu. Tas var būt noderīgs marketplace veida projektos, kur ir nepieciešams elastīgs un pielāgojams koda rediģēšanas risinājums, piemēram, ātra un vienkārša koda rediģēšana un testēšana.

**Alternatīva GitHub:**

* **GitHub:** GitHub ir populāra koda versiju kontroles platforma, kas piedāvā plašu sadarbības un projektu pārvaldības rīku klāstu. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama centralizēta koda uzglabāšana un sadarbība starp izstrādātājiem, piemēram, kodu izmaiņu izsekošana un atsauces uz kļūdām.
* **GitLab:** GitLab ir integrēta koda pārvaldības platforma, kas piedāvā līdzīgas funkcijas kā GitHub, bet ar papildu funkcionalitāti un uzstādījumiem. Tas var būt noderīgs marketplace veida projektos, kur ir nepieciešama pilnīga uzņēmējdarbības risinājuma integrācija, piemēram, uzstādījumu un atļauju pārvaldība.

### 3.2.6. Izstrādes un kvalitātes rīki

**Alternatīva Nodemon:**

* **Nodemon:** Nodemon ir utilīta, kas uzrauga Node.js lietojumprogrammu izmaiņas un automātiski restartē serveri. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešams ātrs un automatizēts servera restarts pēc koda izmaiņām.
* **Pm2:** Pm2 ir alternatīvs utilīts, kas nodrošina Node.js lietojumprogrammu pārvaldību un procesu uzraudzību. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama vispusīga Node.js lietojumprogrammu pārvaldība un izstrādātāja produktivitāte.

**Alternatīva ESlint:**

* **ESlint:** ESlint ir statiska koda analīzes rīks, kas identificē un novērš problēmas JavaScript kodā. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama kodu kvalitātes uzlabošana un labāko prakšu ievērošana.
* **JSHint:** JSHint ir alternatīvs JavaScript kodu pārbaudes rīks, kas piedāvā līdzīgu funkcionalitāti kā ESlint. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama JavaScript kodu kvalitātes uzlabošana un kļūdu atrašana.

**Alternatīva dotenv:**

* **dotenv:** dotenv ir pārvaldīšanas vides mainīgais rīks, kas ielādē vides mainīgos no .env faila. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama droša un ērta vides konfigurācija un konfidencialitāte.
* **config:** config ir alternatīvs vides konfigurācijas rīks, kas piedāvā plašu vides mainīgo pārvaldību un iestatījumu organizāciju. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama augsta konfigurācijas elastība un drošība.

**Alternatīva Prettier:**

* **Prettier:** Prettier ir koda formatēšanas rīks, kas ievieš vienotu koda stilu. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama vienota koda formatēšana un kodu lasāmības uzlabošana.
* **Beautify:** Beautify ir alternatīvs koda formatēšanas rīks, kas piedāvā līdzīgu funkcionalitāti kā prettier. Tas ir noderīgs marketplace veida projektos, kur ir nepieciešama koda skaidrības uzlabošana un vienota stila ieviešana.

# 4. Sistēmas modelēšana un projektēšana

## 4.1. Sistēmas struktūras modelis

**1. attēls. Sistēmas struktūra**

## 4.2. Datubāzes modelis

**2. attēls. Datubāzes datu modelis**

Datubāzes datu modelis (2. attēls) ir grafisks attēlojums, kas raksturo datu bāzes struktūru.

**Tabulas un to attiecības:**

**Lietotāji (User):** Šajā tabulā tiek glabāti visi lietotāju dati, tostarp vārds, uzvārds, e-pasts, parole u.c. Katrs lietotājs var veikt vairākus pasūtījumus un augšupielādēt vairākas mākslas darbu bildes.

**Mākslas darbi (Artwork):** Šī tabulā ir informācija par visiem mākslas darbiem, tostarp nosaukums, apraksts, kategorija, cena u.c. Katrs mākslas darbs var būt saistīts ar vienu pasūtījumu.

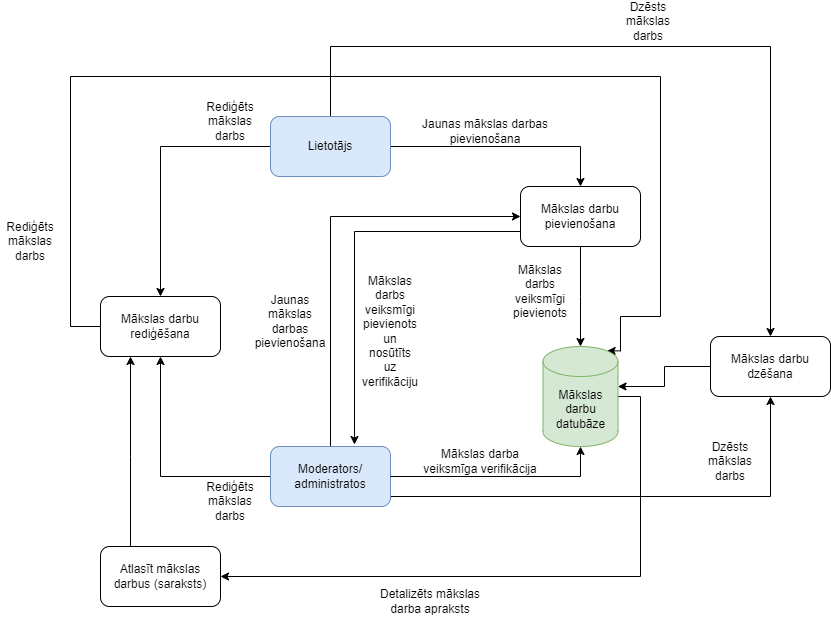
**Pasūtījumi (Order):** Šeit tiek glabāta informācija par katru pasūtījumu, tostarp pasūtījuma numurs, pasūtītāja dati, pasūtīto mākslas darbu saraksts, summa, piegādes adrese u.c.

**Maksājumi (Payment):** Šajā tabulā ir informācija par veiktajiem maksājumiem, tostarp statuss, summa, maksājuma datums un laiks u.c. Katrs maksājums ir saistīts ar konkrētu pasūtījumu.

**Bildes (Image):** Šī tabulā tiek glabātas bildes, kas saistītas ar mākslas darbiem. Katra bilde ir saistīta ar vienu mākslas darbu.

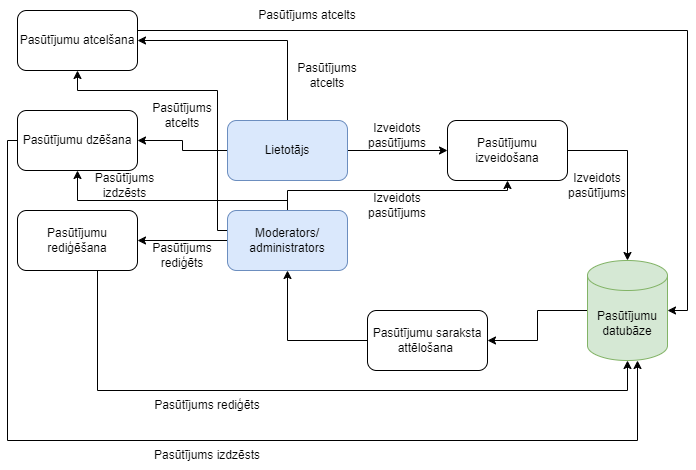
Šīs tabulas veido saistības starp dažādiem datu elementiem un ļauj efektīvi pārvaldīt platformu, ievērojot klientu pasūtījumus, pārvaldot mākslas darbu informāciju un apstrādājot maksājumus.

## 4.3. Funkcionālais un dinamiskais sistēmas modelis

Mākslas darbu datu plūsmas diagramma ir grafisks attēlojums, kas raksturo, kā dati par mākslas darbiem pārvietojas dažādas sistēmas komponentēs. Šī diagramma attēlo mākslas darbu datu plūsmu no sākuma līdz beigām, ietverot to pievienošanu, uzglabāšanu, dzēšanu, rediģēšanu un pārsūtīšanu starp dažādām sistēmas daļām. (sk. 3. attēls)

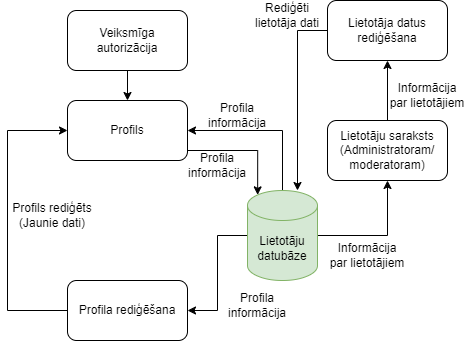
**3. attēls. Mākslas darbu datu plūsmas diagramma**

Pasūtījumu datu plūsmas diagramma ir grafisks attēlojums, kas raksturo, kā dati par pasūtījumiem pārvietojas dažādas sistēmas komponentēs. Šī diagramma attēlo pasūtījumu datu plūsmu no sākuma līdz beigām, ietverot to izveidošanu, uzglabāšanu, dzēšanu, rediģēšanu un pārsūtīšanu starp dažādām sistēmas daļām. (sk. 4. attēls)

**4. attēls. Pasūtījumu datu plūsmas diagramma**

Reģistrācijas datu plūsmas diagramma ir grafisks attēlojums, kas raksturo, kā reģistrācijas dati pārvietojas dažādas sistēmas komponentēs. Šī diagramma attēlo reģistrācijas datu plūsmu no sākuma līdz beigām, ietverot to ievadi, validāciju, uzglabāšanu un pārsūtīšanu starp dažādām sistēmas daļām. (sk. 5. attēls)

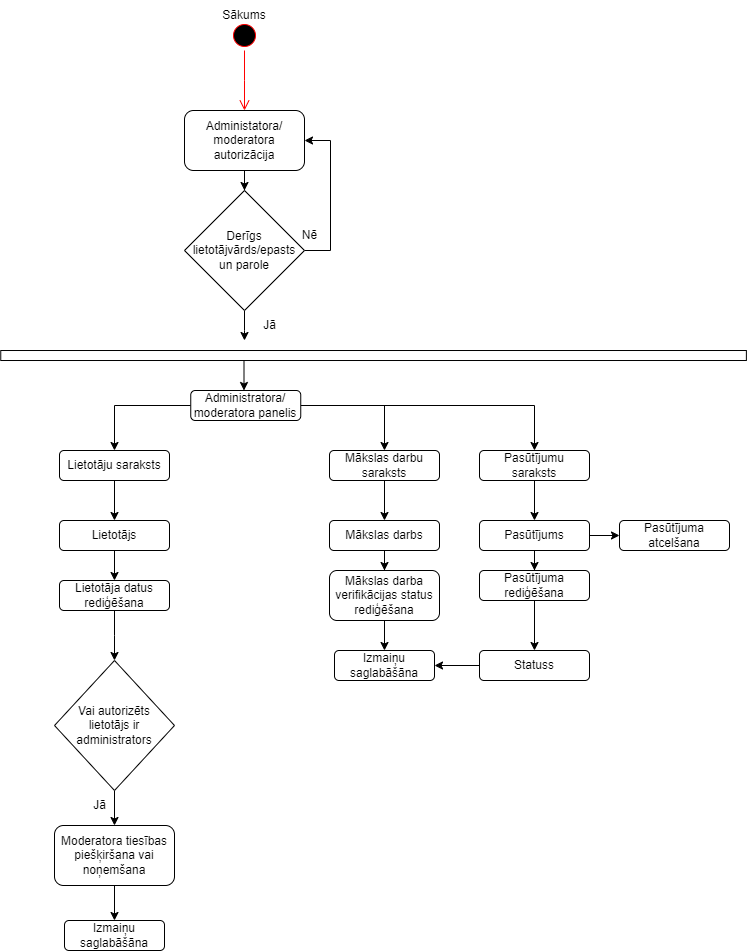
**5. attēls. Reģistācijas datu plūsmas diagramma**

Lietotāju datu plūsmas diagramma ir grafisks attēlojums, kas raksturo, kā profila dati pārvietojas dažādas sistēmas komponentēs. Šī diagramma attēlo reģistrācijas datu plūsmu no sākuma līdz beigām, ietverot to pievienošanu, rediģēšanu, uzglabāšanu un pārsūtīšanu starp dažādām sistēmas daļām. (sk. 6. attēls)

**6. attēls. Lietotāju datu plūsmas diagramma**

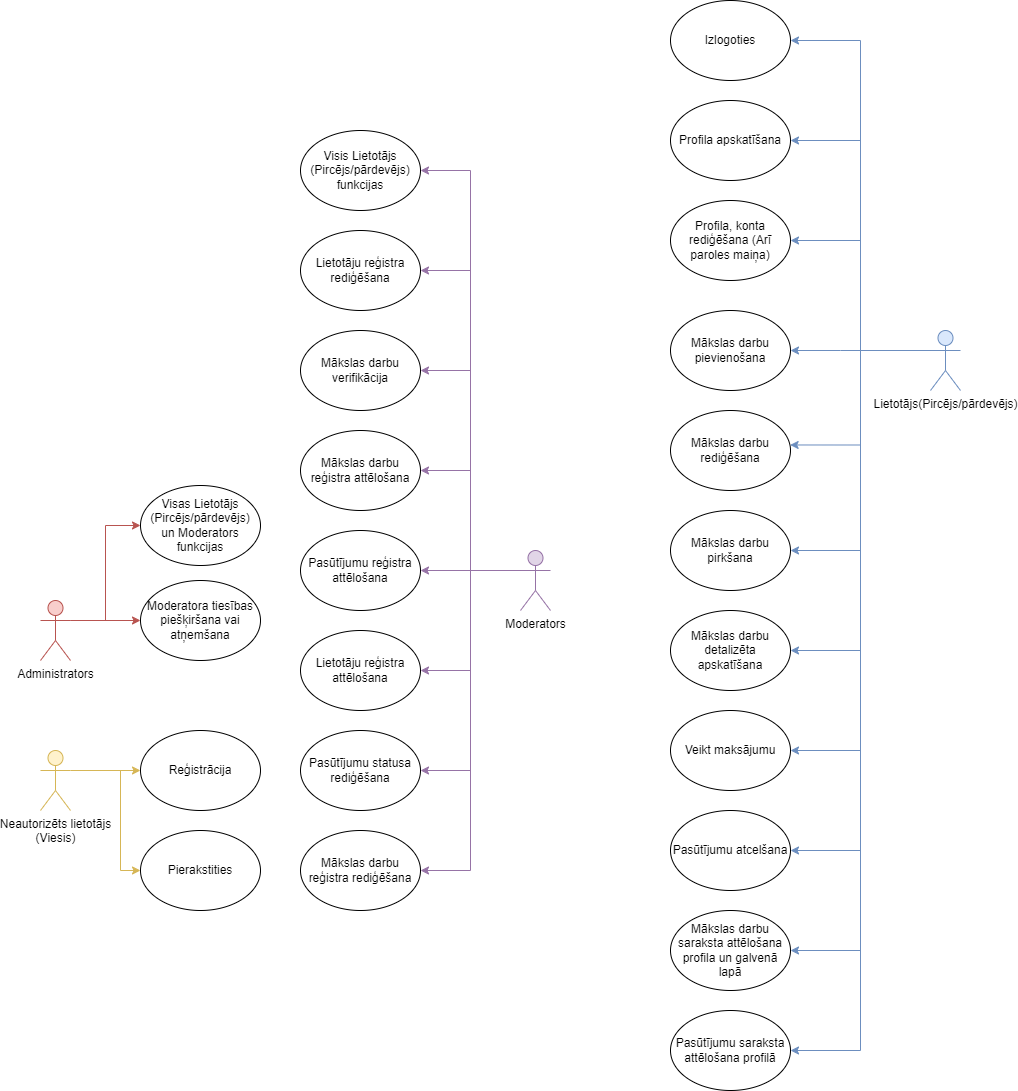
## 4.4. Aktivitāšu diagramma (Activity)

**7. attēls. Lietotāju aktivitāšu diagramma.**

Administratoram un moderatoriem ir visas tadas pašas funkcijas, kā parastājiem lietotājiem. Ši administratora/moderatora aktivitāšu diagrammā ir grafisks attēlojums, administrātora un moderatora funkcionālitātes (sk. 8. attēls).

**8. attēls. Administratora/moderatora aktivitāšu diagramma**

## 4.5. Lietojumgadījumu diagramma (Use Case)

Lietojumgadījumu diagramma vizuāli attēlo, ka lietotāji mijiedarbojas ar platfomu dažādos lietojumgadījumos.

**9. attēls. Lietojumgadījumu diagramma.**

## 4.6. Sistēmas moduļu apraksts un algoritmu shēmas

**Lietotāju pārvaldība:**

Šis modulis nodrošina lietotāju reģistrāciju, autentifikāciju un profilu pārvaldību. Tas ietver arī lietotāju atļauju un lomu pārvaldību, kas var ietekmēt piekļuvi noteiktām funkcijām un datiem.

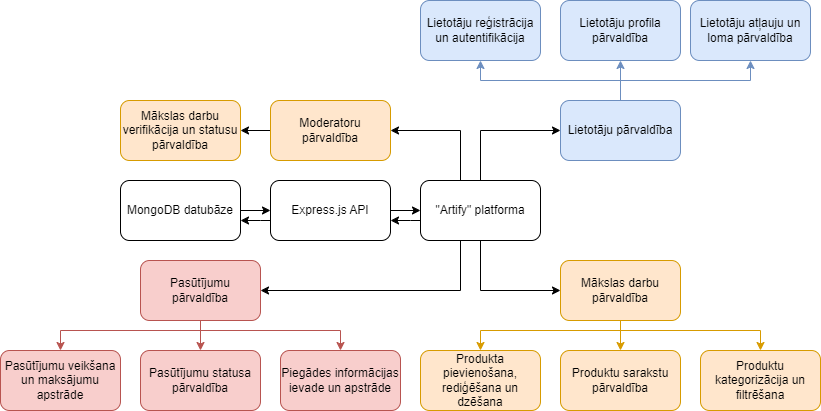
**Produktu pārvaldība:**

Produktu pārvaldības modulis ļauj lietotājiem pievienot, rediģēt un dzēst savus produktus. Tas ietver arī produktu sarakstu pārvaldību un meklēšanu, kā arī kategorizācijas un filtrēšanas iespējas, lai atvieglotu produktu atrašanu.

**Pasūtījumu pārvaldība:**

Šis modulis atbild par pasūtījumu veikšanu, maksājumu apstrādi un pasūtījumu statusu pārvaldību. Tas ietver pasūtījumu veikšanu, pasūtījumu statusu maiņu un piegādes informācijas ievadi un apstrādi.

**Moderatoru pārvaldība:**

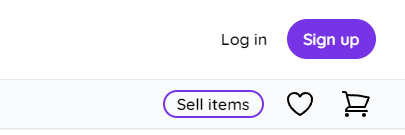
Šis modulis ir atbildīgs par darbu verifikāciju un pasūtījumu apstrādi. Moderatori pārskata jaunus produktus, pārbauda to atbilstību kvalitātes standartiem un veic pasūtījumu statusu maiņu atkarībā no pasūtījuma apstākļiem.

**10. attēls. Sistēmas moduļu diagramma**

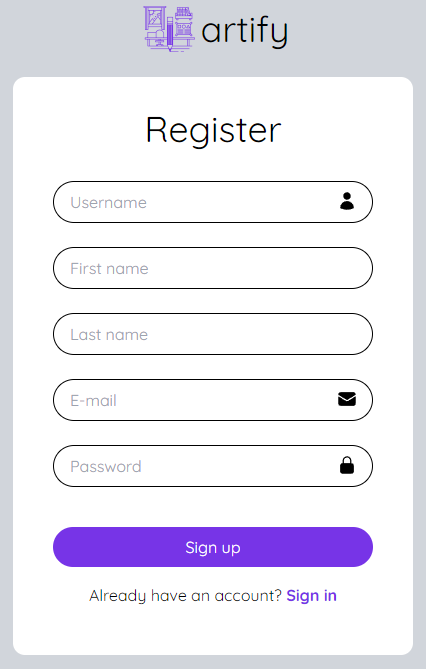
# 5. Lietotāju ceļvedis

Šī nodaļā ir informācija par to, kā izmantot platformu “Artify”.

## 5.1. Reģistrācija

1. Lai reģistrētos, lietotājam jānospiež pogu “Sign up” mājaslapas augšēja labaja stūrī (sk. 11. attēls).

**11. attēls. “Sign up” poga.**

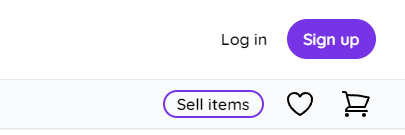
2. Lietotājs redzēs reģistrācijas formu, kur jāievadā datus. (sk. 12. attēls).

**12. attēls. Reģistrācijas logs.**

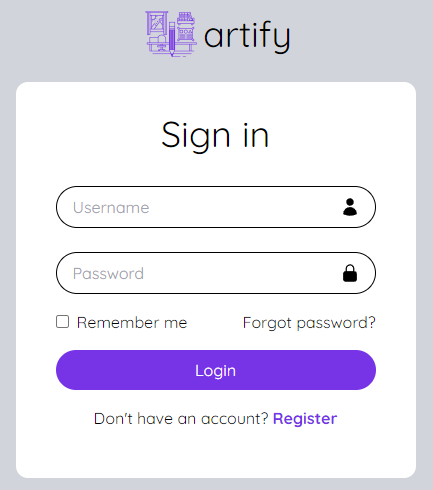
3. Pēc korektu datu ievadīšanas un pogas “Sign up” formas apakšeja daļā nospiešanas, reģistrācija ir veiksmīga un lietotājs tiek novirzīts uz savu profilu.

## 5.2. Ielogošanas

1. Lai ielogoties, lietotājam jānospiež pogu “Log in” mājaslapas augšēja labaja stūrī (sk. 13. attēls).



**13. attēls. “Log in” poga.**

****2. Lietotājs redzēs ielogošanas formu, kur jāievadā lietotājvārdu vai e-pastu un paroli. (sk. 14. attēls)**.**

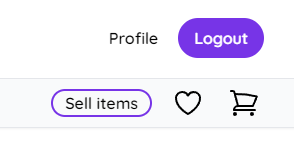
**14. attēls. Ielogošanas logs.**

3. Pēc korektu datu ievadīšanas un pogas “Login” formas apakšeja daļā nospiešanas, lietotājs veiksmīgi ielogojies un var apskatīt savu profilu un veikt citas darbības.

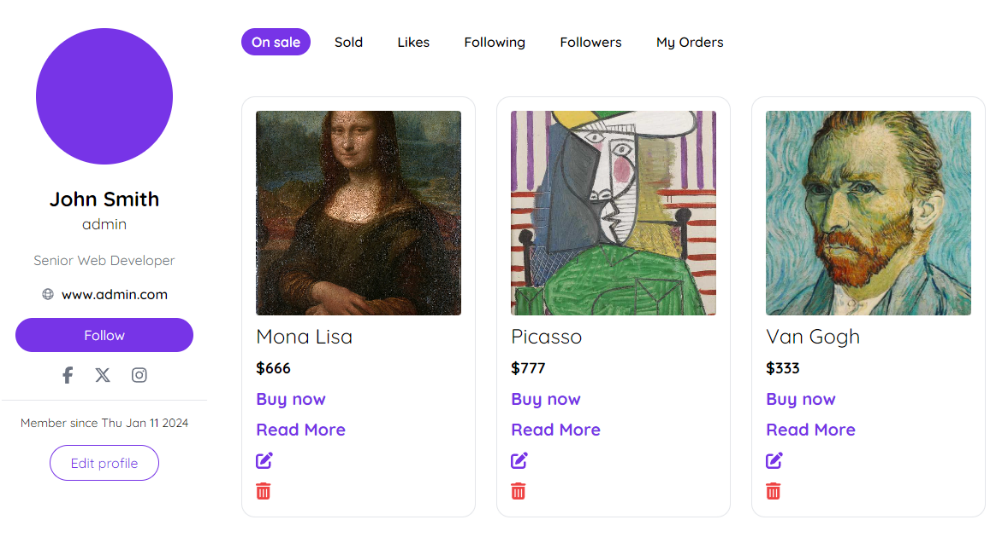
## 5.3. Profila apskatīšana

1. Jāielogojas (sk. 5.2. sadaļa).

2. Jānospiež pogu “Profile” mājaslapas augšēja labaja stūrī (sk. 15. attēls).



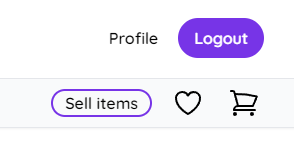
**15. attēls. “Profile” poga.**

3. Lietotājs tiek novirzīts uz savu profilu, kur viņs var pārslēgties starp profila sadaļam, klikšķinot uz atbilstošam pogam (sk. 16 attēls).

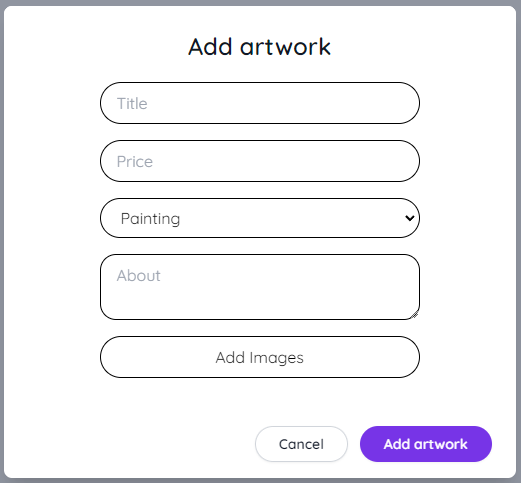
**16. attēls. Profils.**

## 5.4. Mākslas darba pievienošana

1. Jāielogojas (sk. 5.2. sadaļa).

2. Jānospiež pogu “Sell items” mājaslapas augšēja labaja stūrī (sk. 17. attēls).

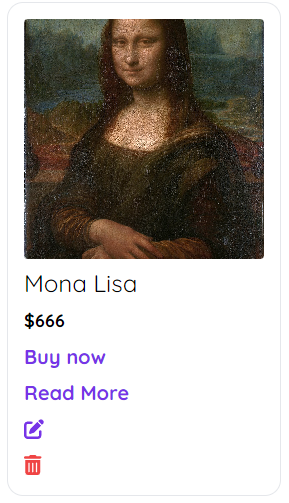
**17. attēls. “Sell items” poga.**

3. Atvērsies pievienošanas forma, kur jāievada nepieciešamos datus (sk. 18. attēls).

**18. attēls. Mākslas darbu pievienošanas logs.**

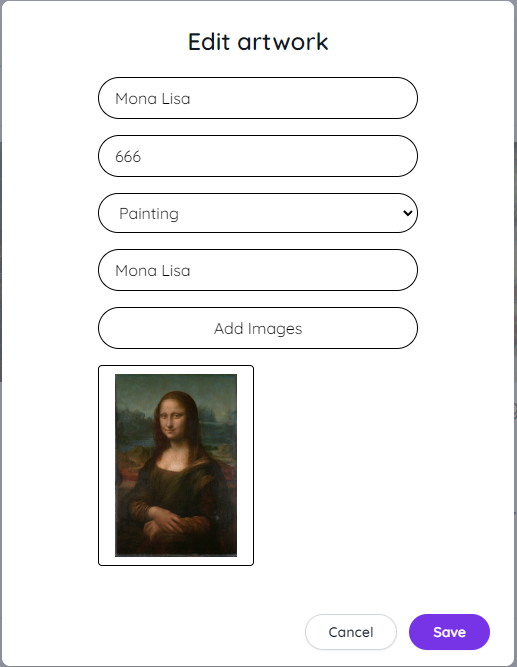
3. Pēc korektu datu ievadīšanas, mākslas darbs ir veiksmigi pievienots un lietotājs saņem paziņojumu par to, ka mākslas darbs tiek nosūtīts uz pārbaudi moderatoram.

## 5.5. Mākslas darbu rediģēšana

1. Jānospiež uz rediģēšnas pogu mākslas darba apakšeja kreisaja stūrī (sk. 19. attēls).

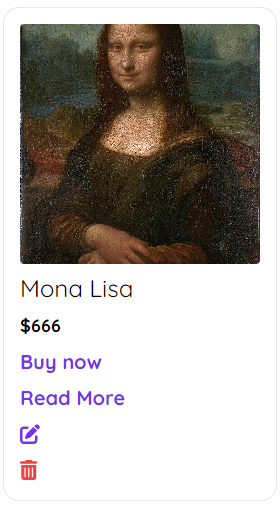
**19. attēls. Mākslas darba rediģēšanas poga.**

2. Atvērsies mākslas darba rediģēšanas logs, kur lietotājs var rediģēt nepieciešamos datus (sk. 20. attēls).

**20. attēls. Mākslas darba rediģēšanas logs.**

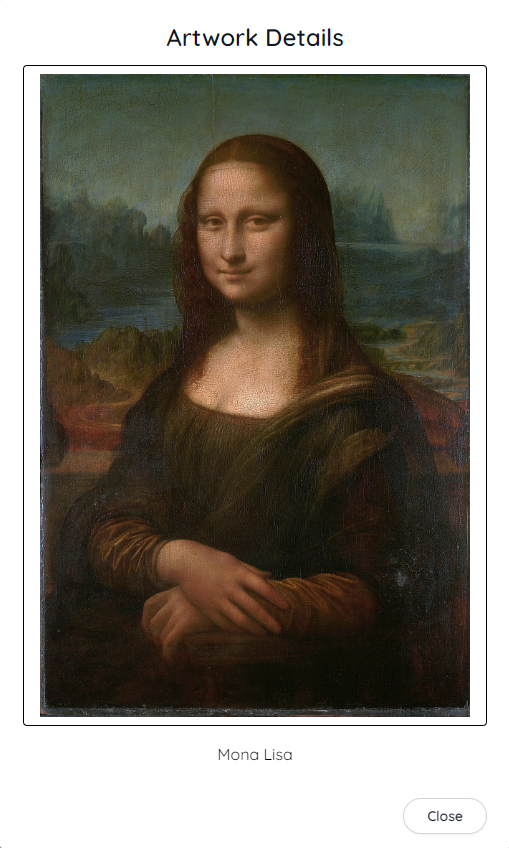
3. Pēc korektu datu ievadīšanas, mākslas darbs veiksmīgi rediģēts un lietotājs saņem paziņojumu.

## 5.6. Mākslas darbu detalizēta apskatīšana

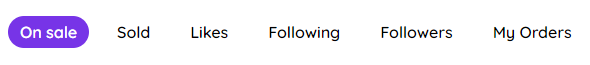
****1. Jānospiež pogu “Read More” mākslas darbā (sk. 21. attēls).

**21. attēls. “Read More” poga.**

2. Atvērsies mākslas darbu detalizētas apskatīšanas logs (sk. 22. attēls).

**22. attēls. Mākslas darbu detalizētas apskatīšanas logs.**

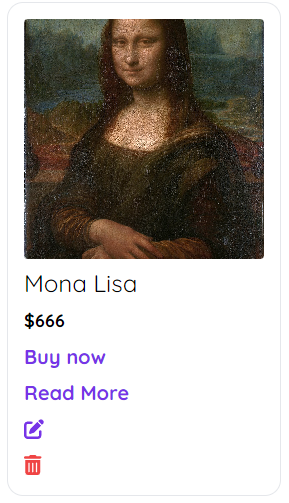
## 5.7. Pasūtījumu apskatīšana un atcelšana

1. Jānospiež uz pogu “My Orders” profilā (sk. 23. attēls).

**23. attēls. “My Orders” poga.**

2. Atvērsies sadaļa ar pasūtījumiem, kur pie pasūtījuma ir poga “Cancel Order”, pēc apstiprinājuma, pasūtījums būt atcelts un lietotājs saņems paziņojumu.

## 5.8. Mākslas darba dzēšana

****1. Jānospiež uz dzēšanas pogu mākslas darba apakšeja kreisaja stūrī (sk. 24. attēls).

**24. attēls. Mākslas datba dzēšanas poga.**

2. Pēc lietotāja apstiprinājuma mākslas darbs tiek izdzēsts un lietotājs saņems paziņojumu.

## 5.9. Mākslas darbu rediģēšana administratoram/moderatoram

1. Jānospiež pogu “Admin panel” un jāizvēlas sadaļu “Artwork list”.

2. Jānospiež uz mākslas darba rediģēšanas pogu.

3. Rediģēt datus vai verificēt mākslas darbu.

## 5.10. Lietotāju datus rediģēšana administratoram/moderatoram

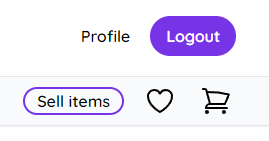
1. Jānospiež pogu “Admin panel” un jāizvēlas sadaļu “User list”.

2. Jānospiež uz lietotāju rediģēšana pogu.

3. Rediģēt datus vai piešķirt/atņemt moderatora tiesības.

## 5.11. Izlogoties

1. Jānospiež pogu “Logout” mājaslapas augšēja labaja stūrī (sk. 25. attēls).

**25. attēls. “Logout” poga.**

2. Lietotājs tiek izlogots un novirzīts uz galveno lapu.

## 5.12. Mākslas darbu pievienošana pirkumu grozā

1. Jānospiež uz groza pogu pie mākslas darba.

2. Mākslas darbs tiek pievienots grozā un pieejams sadaļa “Cart”.

## 5.13. Pirkumu groza apskatīšana

1. Jānospiež uz groza pogu mājaslapas “Header” daļa.

## 5.14. Mākslas darbu “Like”

1. Jānospiež uz “Like” pogu pie mākslas darba.

2. Mākslas darbs tiek pievienots un pieejams“Like” sadaļa.

## 5.15. “Like” mākslas darbu apskatīšana

1. Jānospiež uz “Like” pogu mājaslapas “Header” daļa.

## 5.16. Mākslas darbu pirkšana

1. Jānospiež uz pogu “Buy Now” pie mākslas darba vai pogu “Buy” sadaļā “Cart”.

2. Jāievada piegādes un maksājuma datus.

3. Pēc korektus datus ievadīšanas un apstiprināšanas, lietotājs saņems paziņojumu un pasūtījums parādīsies “”My Orders” sadaļa (sk. 5.7 sadaļa).

# 6. Testēšanas dokumentācija

Šī sadaļa ir informācija par projekta testēšanu, ieskaitot testpiemēru kopu un testēšanas žurnālu, kā arī informāciju par testēšanas metodem un rīkiem.

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

* Testēšanai izvēlēta manuāla testēšana, jo tas piedāvā elastīgu pieeju, ļaujot rūpīgi pārbaudīt katra funkcionalitātes aspektu un lietotāja pieredzi. Projekts nav pārāk liels, tāpēc manuālā testēšana ir praktiska un efektīva izvēle. Melnas kastes (black-box) metode ļauj testēt sistēmu no lietotāja perspektīvas, neievērojot tās iekšējo darbību, tādējādi nodrošinot, ka visas funkcijas strādā tā, kā paredzēts. No otras puses, baltas kastes (white-box) metode ļauj pārbaudīt sistēmu, zinot tās iekšējo darbību, kas var būt noderīgi, lai identificētu un novērstu iespējamās kļūdas.
* Manuālā testēšana ir arī pielāgojama un var tikt pielāgota konkrētajiem projektam un tā prasībām. Tā sniedz iespēju detalizēti pārbaudīt visus projektā iekļautos moduļus un funkcijas, kā arī nodrošina elastīgu pieeju problēmu identificēšanai un novēršanai. Šī pieeja ir piemērota individuāli veidotajam projektam, jo ļauj personīgi pārbaudīt katru aspektu, izmantojot savu zināšanu un izpratni par sistēmas darbību. Tas ir īpaši svarīgi, ņemot vērā, ka projektam ir viens izstrādātājs projektā, tāpēc ir pilnīga kontrole pār testēšanas procesu un rezultātiem.

## 6.2. Testpiemēru kopa

## 6.3. Testēšanas žurnāls

# 7. Secinājumi

Pēc platformas izstrādes, ir svarīgi padziļināti izanalizēt sasniegumus, secinājumus un nākotnes perspektīvas.

**Sasniegums un Rezultāta Novērtējums:**

Izveidotā mākslas darbu marketplace platforma nodrošina māksliniekiem un pircējiem efektīvu un pievilcīgu vidu mākslas darbu pārdošanai un iegādei tiešsaistē. Lietotāji var viegli reģistrēties, augšupielādēt darbus un veikt drošus maksājumus, radot veiksmīgu tirdzniecības vide. Viss bija izveidots ta, kā bija ieplānots no sākuma. Ieguvu ļoti daudz jaunas zināšanas par tīmekļa vietņu izstrādi, gar par front-end, gan par back-end.

**Izvirzīto Uzdevumu Analīze:**

Lai gan visi izvirzīti uzdevumi tika veiksmīgi pabeigti, bija dažādi izaicinājumi:

* integrēt maksājumu sistēmu un nodrošināt datu drošību;
* izveidot API;
* izveidot attēlus atbilstošu pievienošanu, rediģēšanu un glabāšanu;
* izdomāt un noformulēt visas prasības un projekta elementus.

Tomēr šīs problēmas tika atrisinātas, un platforma darbojas uzticami. Nebija tādas funkcionālitātes, kuru neizdevas pievienot projektam.

**Darba Apjoms:**

Izstrādāta plaša moduļu sistēma un vairāk nekā 25 funkcijas, kas veido kompaktu un funkcionālu platformu. Projekta mapē ir vairāk nekā 15 failus ar komponentiem, 4 faili ar datubāzi un 4 faili ar back-end funkcionālitāti. Projekts ir labi strukturēts un sadalīts pa daļam. Kopējais failu un moduļu skaits norāda uz apjomīgu darbu, kas veikts, lai nodrošinātu platformas darbību un funkcionalitāti. Kopīgais komandrindu skaits ir apmēram 3300 rinas. Funkcionālas komandrindu skaits (izņemot dizainu) ir apmēram 2000 rindas.

**Testēšana:**

Platforma tika rūpīgi testēta uz dažādām pārlūkprogrammām – Chrome, Microsoft Edge, Firefox un ierīcēm – dators, planšetdators, mobilais telefons, nodrošinot, ka tā darbojas stabili un pareizi visos apstākļos. Testi atklāja dažas nianses, saistītas ar responsīvo dizainu, kuras tika labotas, nodrošinot optimālu lietotāju pieredzi.

**Problēmas un Izaicinājumi:**

Sastapos ar dažādām problēmām, piemēram, sarežģītību integrēt maksājumu, izveidot API. Strādāju ar jaunām priekš manis tehnoloģijām, jo ļoti interesējos par tīmekļa izstrādi un mūsdienu standartiem un tas nebija tik viegli. Tomēr, pateicoties rūpīgai plānošanai un izpildei, šīs problēmas tika veiksmīgi atrisinātas.

**Nākotnes Ieceres:**

Plānots turpināt platformas attīstību, pievienojot jaunas funkcijas un uzlabojot esošās. Mērķis ir piedāvāt lietotājiem vēl plašākas iespējas un uzlabot mākslas darbu tirdzniecības pieredzi tiešsaistē. Var pievienot labāku meklēšanu, atsauksmes sistēmu, mākslas darbu rekomendācijas utt.

# 8. Lietoto terminu un saīsinājumu skaidrojumi

**18. tabula**

**Lietoto terminu un saīsinājumu skaidrojumi**

|  |  |
| --- | --- |
| **Termins vai saīsinājums** | **Skaidrojums** |
| JWT | JSON Web Token, kas tiek izmantots kā drošs veids, kā pārsūtīt informāciju starp front-end un back-end. Tas ir bieži izmantots autentifikācijai tīmekļa lietojumprogrammās un API. |
| Front-end | Lietotāja saskarne, kas sastāv no elementiem un mijiedarbības, ko lietotāji tieši redz un ar ko mijiedarbojas tīmekļa lietojumprogrammā. |
| Back-end | Servera puse lietojumprogrammā, kas atbild par pieprasījumu apstrādi, datubāzu pārvaldību un citām uzdevumu veikšanu, kas nav tieši redzami lietotājiem. |
| PPS | Programmatūras Prasību Specifikācija |
| localStorage | Tīmekļa pārlūka funkcija, kas ļauj tīmekļa vietnēm saglabāt datus lokāli uz lietotāja ierīces. |
| UI | Lietotāja saskarne; vizuālie elementi un izkārtojums, ar kuriem lietotāji mijiedarbojas tīmekļa vietnē vai lietojumprogrammā. |
| ID | Saīsinājums no "identifikators", tas ir unikāls vērtības, kas tiek izmantots, lai atšķirtu vai identificētu konkrētu entitāti, piemēram, lietotāju vai vienumu datubāzē. |
| API | API ir savstarpējas komunikācijas noteikumu un protokolu komplekts starp dažādām programmatūras lietotnēm. Tas ļauj šīm lietotnēm apmainīties ar informāciju un sazināties, definējot pieejamos datu formātus un metodes. |

# 9. Literatūras un informācijas avotu saraksts

1. <https://ag-grid.com/>

2. <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/Introduction>

3. <https://www.w3schools.com/mongodb/>

4. <https://mongoosejs.com/>

5. <https://builtin.com/articles/web-development-frameworks>

6. https://react.dev/learn

...

# PIELIKUMS