



VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS INSTITUTAS
KOMPIUTERINIO IR DUOMENŲ MODELIAVIMO KATEDRA

Programų sistemų inžinerijos projektas

VŪkis

Atliko:

Agnė Butkutė

Tomas Dunčikas

Edvinas Želnys

Tomasz Kosa

Vadovas:

Lekt. Virgilijus Krinickij

Vilnius
2019

Turinys

1. Verslo planas	3
1.1. Komanda	3
1.2. Terminų žodynės	3
1.2.1. Tikslas ir misija	3
1.2.2. Sistemos aprašymas iš naudotojo perspektyvos	3
1.2.3. Sistemos aprašymas iš kūrėjų perspektyvos	4
1.2.4. Programos išvaizda	4
2. Sistemos reikalavimai	9
2.1. Funkciniai reikalavimai	9
2.2. Ne funkciniai reikalavimai	10
3. Projekto planas	11
3.1. Pagrindinės rizikos	11
3.2. Darbo paskirstymas	11
3.3. Grafikas	12
4. Sistemos dizainas	14
4.1. Sistemos užduočių diagrama	14
4.2. UML diagrama	15
4.3. UML komponentų diagrama	16
5. Technologijos	18
5.1. Technologijų paketas	18
5.2. Veikimo aplinka	18
6. Kūrimo įrankiai	19
6.1. Versijų kontroles sistema	19
6.2. Programinio kodo apžvalga	19
6.3. Kūrimo ir implementavimo modelis	19
6.4. Testavimo automatizavimas	20
6.5. Darbo planavimas	20
7. Paleidimas	20
7.1. Paleidimo diagrama	21

1. Verslo planas

1.1. Komanda

Komandos pavadinimas – ETTA @. Komandos nariai:

1. Agnė Butkutė(Vadovė)
2. Tomas Dunčikas(UML makeduotojas)
3. Edvinas Želnys(Testuotojas)
4. Tomasz Kosa(Programuotojas)

1.2. Terminų žodynas

- „**Windows 7**“ - Microsoft kompanijos sukurta operacinė sistema, pakeitusi savo pirmtakę Windows Vista operacinę sistemą. Windows 7 skirta asmeniniams kompiuteriams, išskaitant namų ir verslo stalinius kompiuterius, nešiojamus kompiuterius, planšetinius kompiuterius ir kt.
- **IDE (liet. Integravota kūrimo aplinka)** - programos, turinčios daug priemonių, palengvinančių ir pagreitinančių programinės įrangos kūrimo procesą.
- **Unity** – įvairiaplatformis žaidimų kūrimo įrankis, sukurtas bendrovės „Unity Technologies“. Jį sudaro žaidimų variklis ir integravota kūrimo aplinka (IDE).
- Esant poreikiui, terminai bus pildomi.

1.2.1. Tikslas ir misija

Mūsų tikslas – sukurti linksmą, informatyvų, paprastą ir įdomų žaidimą ne tik jauniems žmonėms, bet ir kiekvieno amžiaus piliečiui.

Mūsų misija – sugalvoti įdomią, linksmą, pamokančią istoriją. Sukurti emociją, jog tai nėra tik skaitmeninis žaidimas, bet lyg tikra animacinė istorija, kurioje dalyvauja ir pats naudotojas. Naudotojo sėsą sukurti patogią, lengvai suprantamą, kad žaidimo paleidimo, įjungimo funkcijos būtų aiškios visiems. Kadangi žaidimas bus ganėtinai paprastas, valdymas taip pat liks lengvas, intuityvus ir puikiai suprantamas. Galimybė ateityje diegti papildymus ir atnaujinimus, kad naudotojai žaidimą pamėgtų vis iš naujo. Taip pat moksliniai tyrimai rodo, kad vaizdo žaidimai gerina koordinacija bei loginį mąstymą. Norint išnaudoti šias mokomąsias galimybes, norime sukurti paprastą žaidimą, kuris reikalautų susikaupimo ir dėmesingumo.

1.2.2. Sistemos aprašymas iš naudotojo perspektyvos

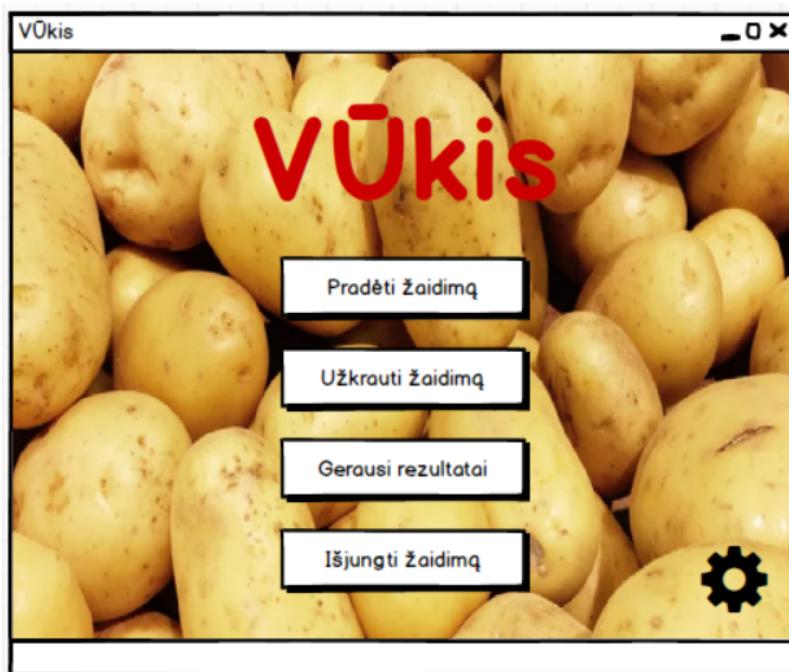
Naudotojai yra skirtingo amžiaus žmonės, kuriems žaidimo pagalba suteikiama galimybė iškūnyti į žaidimo personažą bei nukeliauti į skaitmeninį herojaus pasaulį. Žaidimas išnaudoja žmogaus kūriningumą bei reikalauja tikslumo ir susikaupimo. Žaidimas turės subtilias, subalansuotas spalvas, kurios nebūtų per ryškios ar nepriimtinios mūsų akims. Taip pat sukurs linksmą ir gerą nuotaiką. Fonas turės turėti gražias, malonias iliustracijas, kurios sukurtų jaukią ir malonią aplinką bei neatitrauktų dėmesio nuo pačio žaidimo. Turės bent kelis nepasikartojančius lygius ir boso

vietas, kuriose, norėdamas pereiti į kitą lygi, naudotojas turės įveikti bosą. Tačiau svarbiausia - gerai praleisti laiką. Žaidimas bus sukurtas vienam naudotojui. Šis žaidimas bus nemokamas.

1.2.3. Sistemos aprašymas iš kūrėjų perspektyvos

Sukurti žaidimą, kuris būtų prieinamas visiems, bet tikslinė auditorija yra jaunimas. Mūsų komanda specializuojasi žaidimų kūrimo srityje. I rinką norime paleisti startuolį „VŪkis“. Privalomas pradžios ekranas turi būti lengvai suprantamas ir patrauklus. Žaidimas bus geresnis už panašius konkurentus, tokius kaip „Super Mario“, nes turės naują, negirdėtą ir linksmą istoriją. Žaidimo herojus galės įgyti naujų, nematyti, įspūdingų super galių. Žaidimo eigoje bus galima susstoti, atsipalauduoti ir pasigrožėti kraštovaizdžiu ir fonu. Nauji, įdomūs, vidutinio sunkumo lygai. Geriausių žaidėjų rezultatai bus saugomi duomenų bazėje ir prieinami, bet kuriuo metu mūsų sukurame internetiniame puslapyje. Prieinamumas ateis iš to, jog žaidimą vartotojai bet kada galės atsisiusti iš interneto.

1.2.4. Programos išvaizda



1 pav. Pagrindinis meniu

Tai yra pagrindinis žaidimo meniu puslapis, kuriame galima pasirinkti iš penkių skirtingų parinkčių: "Pradėti žaidimą", "Užkrauti žaidimą", "Geriausi rezultatai", "Išjungti žaidimą" ir nustatymų piktogramos.



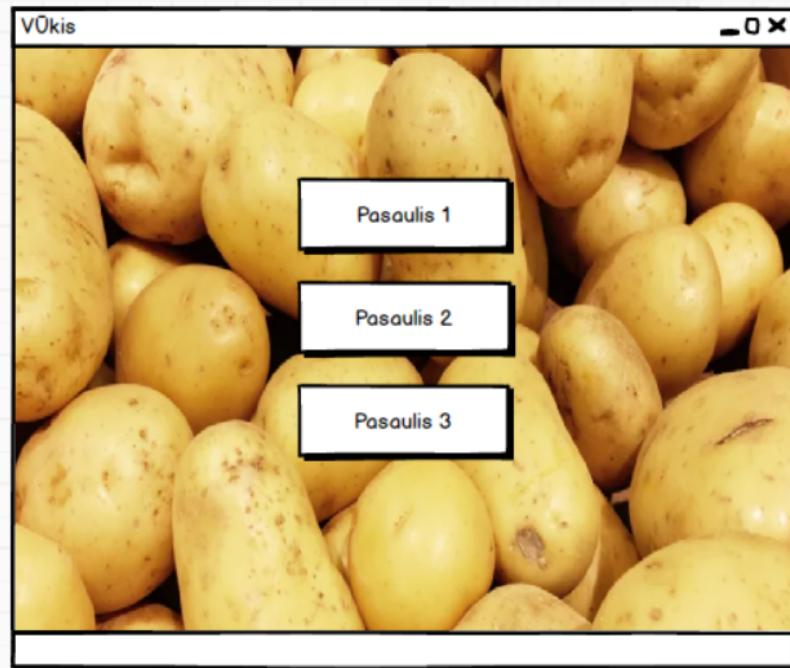
2 pav. Įvesk savo vardą

Paspaudus mygtuką "Pradėti žaidimą" atsirandame šiame puslapyje, kuriame turime išrašyti savo vardą ir paspausti mygtuką "Pradėti", kad žaidimas prasidėtų.

Vardas	Laikas	Mirtys	Bulvės
Bulvius	10min 55sec	40	56
Potatoe	12min 34sec	53	48

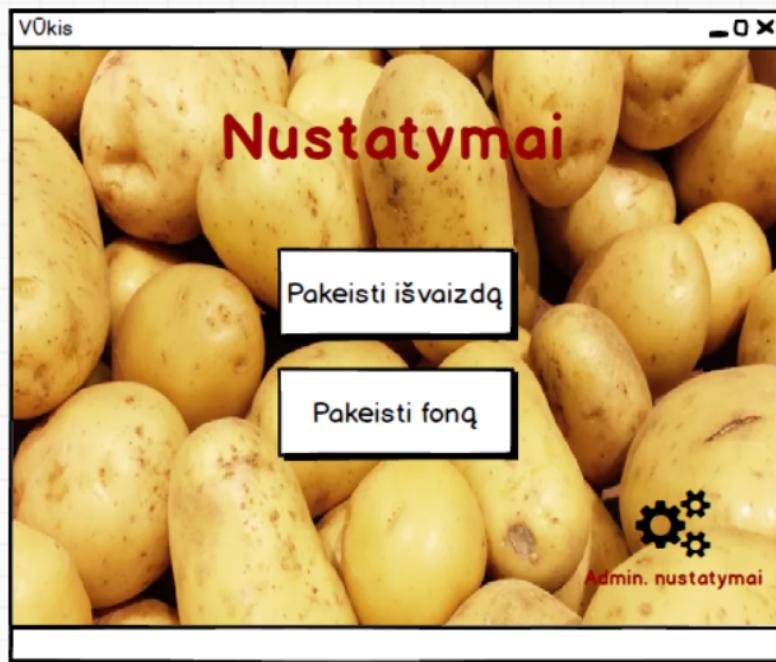
3 pav. Rezultatai

Paspaudus mygtuką "Rezultatai" pagrindiniame meniu puslapyje matome visus rezultatus, kurie yra sudėlioti pagal žaidimo baigimo laiką (greičiausias laikas - 1, t.t)



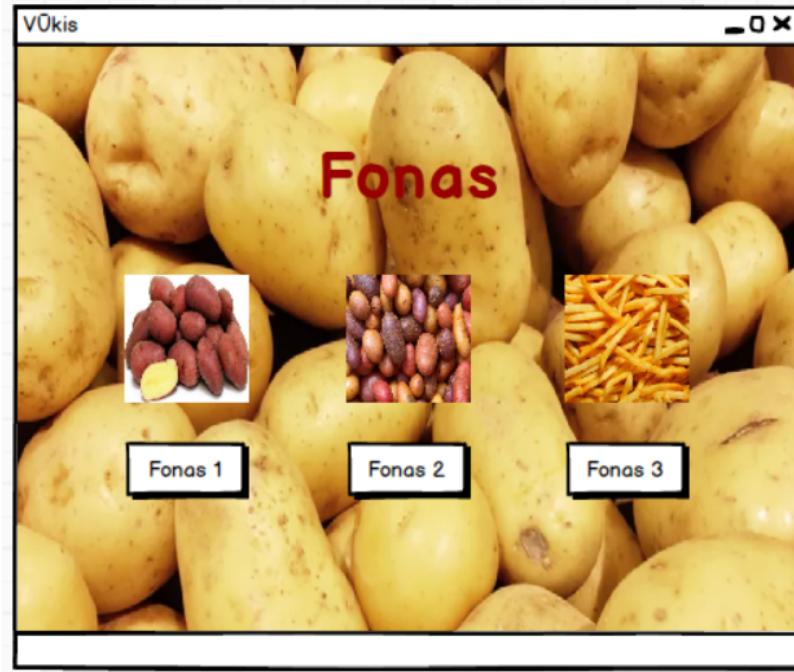
4 pav. Íkrauti žaidimą

Pagrindiniame meniu puslapyje pasirinkus "Užkrauti žaidimą" turime galimybę pasirinkti iš trijų galimų žaidimo pasaulių, paspaudę mygtukus "Pasaulis 1", "Pasaulis 2", "Pasaulis 3".



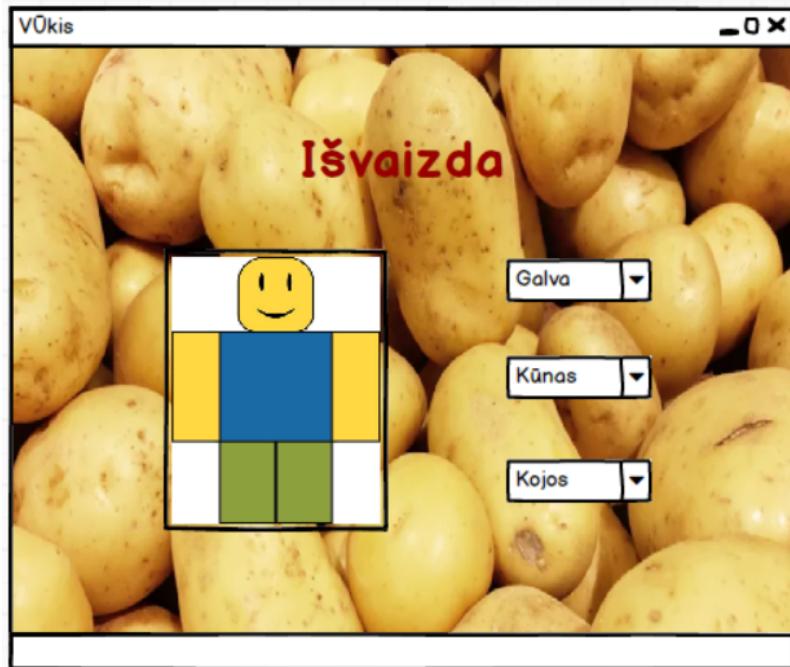
5 pav. Nustatymai

Pasirinkus pagrindiniame meniu puslapyje nustatymų piktogramą, nukreipiama į puslapį "Nustatymai", kuriamo turime galimybę pasirinkti pakeisti pagrindinio veikėjo išvaizdą, žaidimo foną arba, paspaudus ant "Admin. nustatymai" pitogramos, gauti prieigą prie administratoriaus nustatymų.



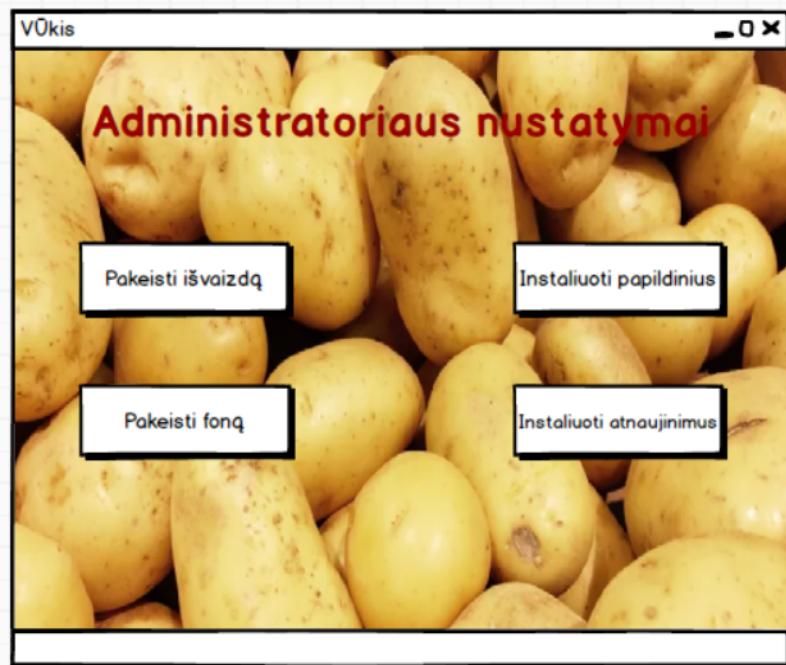
6 pav. Fono keitimas

Šiame puslapyje, į kurį patenkame iš puslapio "Nustatymai" (pav. 5), galima pasirinkti žaidimo foną iš trijų pateikiamų.



7 pav. Išvaizdos keitimas

Šiame puslapyje, į kurį patenkame iš puslapio "Nustatymai" (pav. 5), galima pasirinkti pagrindinio veikėjo išvaizdą, pakeičiant galvą, kūną ir kojas.



8 pav. Administratoriaus nustatymai

Funkcijos skirtos administratoriaus darbui.

2. Sistemos reikalavimai

Šiame punkte bus išdestomi programos funkciniai ir ne funkciniai reikalavimai. Pavaizduota funkcijų lentelė padės orientuotis kokios funkcijos prieinamos naudotojams ir administratoriams, su išsamesniu kiekvienos funkcijos aprašymu. Ne funkciuose reikalavimuose informacija apie žaidimo prieinamumą, tolesnę priežiūrą, saugumą ir naudingumą.

2.1. Funkciniai reikalavimai

Funkcijų lentelė (ang. User function matrix) - joje yra surašytos visos programos funkcijos ir kas gali jas atliki. Mūsų atveju aktoriai yra naudotojas ir administraorius.

Funkcijos/Aktoriai	Naudotojas	Administratorius
Papildinių diegimas		
Pakeisti foną		
Atnaujinimų diegimas		
Pašokti		
Judėti pirmyn		
Judėti atgal		
Irašyti savo vardą		
Pakeisti galvos išvaizdą		
Pakeisti kūno išvaizdą		
Pakeisti kojų išvaizdą		
Patikrinti rezultatus		
Užkrauti pradėtą žaidimą/istoriją		
Rinkti bulves		
Mygtukas išjungti žaidimą		
Mygtukas pajungti žaidimą		
Veikėjo ypatingos galios		
Kliūtys		

1 lentelė. Funkcijų lentelė

- Papildinių diegimas** - administratorius, tačiau ne paprastas žaidimo naudotojas, turi galimybę žaidimui įdiegti naujų papildinių, kurie, jeigu būtų poreikis, pagerintų žaidimą ir žaidimo kokybę;
- Pakeisti foną** - suteikiama galimybė bet kada pasikeisti žaidimo foną į vieną iš pateiktų;
- Atnaujinimų diegimas** - administratorius, tačiau ne paprastas žaidimo naudotojas, turi galimybę žaidimui įdiegti reikalingą atnaujinimą (naujesnė žaidimo versija/ defekto ištaisymas ir pan.);
- Pašokti** - žaidimo metu suteikiama galimybė su pagrindiniu veikėju atliki pašokimo veiksma;
- Judėti pirmyn** - žaidimo metu suteikiama galimybė su pagrindiniu veikėju atliki judėjimo pirmyn veiksmą;

6. **Judėti atgal** - žaidimo metu suteikiama galimybė su pagrindiniu veikėju atlikti judėjimo atgal veiksma;
7. **Irašyti savo vardą** - prieš pradedant žaidimą, žaidėjas turi galimybę išrašyti norimą vardą, kurio pagalba sistemoje bus fiksuojami rezultatai;
8. **Pakeisti galvos išvaizdą** - suteikiama galimybė pakeisti pagrindinio veikėjo galvos išvaizdą, iš bet kurią iš pateiktų galimybių;
9. **Pakeisti kūno išvaizdą** - suteikiama galimybė pakeisti pagrindinio veikėjo kūno išvaizdą, iš bet kurią iš pateiktų galimybių;
10. **Pakeisti kojų išvaizdą** - suteikiama galimybė pakeisti pagrindinio veikėjo kojų išvaizdą, iš bet kurią iš pateiktų galimybių;
11. **Patikrinti rezultatus** - suteikiama galimybė peržiūrėti sužaistų žaidimų rezultatus ir juos sekti ar gerinti;
12. **Užkrauti pradėtą žaidimą/istoriją** - suteikiama galimybė pratęsti pradėtą žaidimą, kurį žaidė prieš tai;
13. **Rinkti bulves** - suteikiama galimybė žaidimo metu rinkti bulves ir taip išveikti žaidimo lygi;
14. **Mygtukas išjungti žaidimą** - paspaudus šį mygtuką žaidimas išjungiamas;
15. **Mygtukas pajungti žaidimą** - paspaudus šį mygtuką įjungiamas žaidimas;
16. **Veikėjo ypatingos galios** - pagrindinis veikėjas turi galimybę žaidimo metu išgyti ypatingų galių, jeigu žaidimo metu suranda tai suteikiančią objektą;
17. **Kliūtys** - žaidimo metu pagrindinis veikėjas, pasitelkdamas aukščiau išvardintas kompetencijas, gali išveikti žaidimo metu atsirandančias kliūtis.

2.2. Ne funkciniai reikalavimai

Norint paleisti žaidimą, sistema turėtų turėti „Windows 7“ arba naujesnę operacinės sistemos versiją. Prieinamumas būtų labai paprastas, žaidimą atsišusti iš interneto. Žaidimo dokumentacija prieinama naudotojams. Efektyvumas atsižvelgiant į turimus resursus. Emociniai faktoriai paliekantys naudotojui įspūdį. Galimybė toliau diegti papildinius ir atnaujinimus. Susidorojimas su trūkumais, trikdžiais (ang. bug). Galimybė pakeisti kodą, pridėti savo norimas funkcijas. Kokybė – testuotojų patikrintos visos žaidimo galimybės ir funkcijos, arba trukdžių šalinimas. Saugumas – saugomi žaidėjų rezultatai. Charakteristika – priklausys nuo to, ant kokio galingumo sistemos bus žaidžiamas žaidimas.

3. Projekto planas

Šiame punkte pateikta informacija apie rizika ir problemas, su kuriomis galime susidurti programos kūrimo metu. Taip pat darbo paskirstimas kur punktais surašyti visi planuojami darbai, o toliau darbo grafike - darbai ir užduotys pavaizduoti laike.

3.1. Pagrindinės rizikos

Pagrindinė rizika - įgūdžių stoka, todėl darbai gali būti neatlikti iki projekto pabaigos. Taip pat, sugeneruotų idėjų neišpildymas, pagal mūsų planą. Rizika, jog nepavyks prijungti rezultatų sekimo funkcijos prie duomenų bazės. Nepavyks pridėti veikėjo super galių. Funkcija, kuri pakeičia foną, neveiks ir fonas liks tas pats. Nepavyks pridėti administratoriaus lango, kurio pagalba bus diegiami papildymai.

3.2. Darbo paskirstymas

- Išanalizuoti projekto reikalavimus;
- Pristatyti komandą ir pagrindinę idėją;
- Paruošti dokumentaciją;
- Išsidiegti „Unity 3D“ programą ir IDE (angl. Intergrated development enviroment)
- Sukurti interneto svetainę, kurioje:
 - patalpintas žaidimas, kurį galima atsisusti;
 - galima sekti geriausius rezultatus;

* rezultatai prijungti duomenų bazėje;
- Veikėjo sukūrimas;
- Veikėjo funkcijų sukūrimas;
- Žaidimo aplinkos kūrimas;
- Žaidimo grafikos kūrimas;
- Žaidimo istorijos kūrimas;
- Nubraižyti UML diagramą;
- Administratoriaus nustatymų kūrimas;
- Ištestuoti žaidimą:
 - ištaisyti atsiradusius trigdžius;
 - ištaisyti rastus trūkumus;
- Galutinis žaidimo testavimas;
- Paruošti trumpą filmuką žaidimo pristatymui su „Balsamiq“ programa;
- Projekto pristatymas

3.3. Grafikas

2019 RUGSĒJIS						
SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4 Susipažinimas su PSI	5	6	7
8	9	10 Projekto reikalavimų analizavimas	11	12	13	14
15	16	17	18	19 Komandos ir pagrindinės idėjos pristatymas	20	21
22	23 Dokumentacijos ruošimas	24	25	26	27	28
29	30					

9 pav. Darbo grafikas 1

2019 SPALIS						
SUN	MON	TUE	WED	THU	FRI	SAT
		1	2	3 "Unity" diegimas	4	5
6	7	8 Internetinės sveitainės kurimas	9	10	11	12
13	14	15	16	17	18	19 Veikėjo kūrimas
20	21 Veikėjo funkcijų kūrimas	22	23	24	25	26
27	28	29	30 Aplinkos išvaizdos kūrimas	31		

10 pav. Darbo grafikas 2

2019 LAPKRITIS						
SUN	MON	TUE	WED	THU	FRI	SAT
					1	2
3	4 Žaidimo grafikos kurimas	5	6	7	8	9
10	11	12	13	14	15 Istorijos kurimas	16
17	18	19	20 Funkciju lentelės kurimas	21	22	23
24	25	26	27	28 Admin. nustatymų kurimas	29	30

11 pav. Darbo grafikas 3

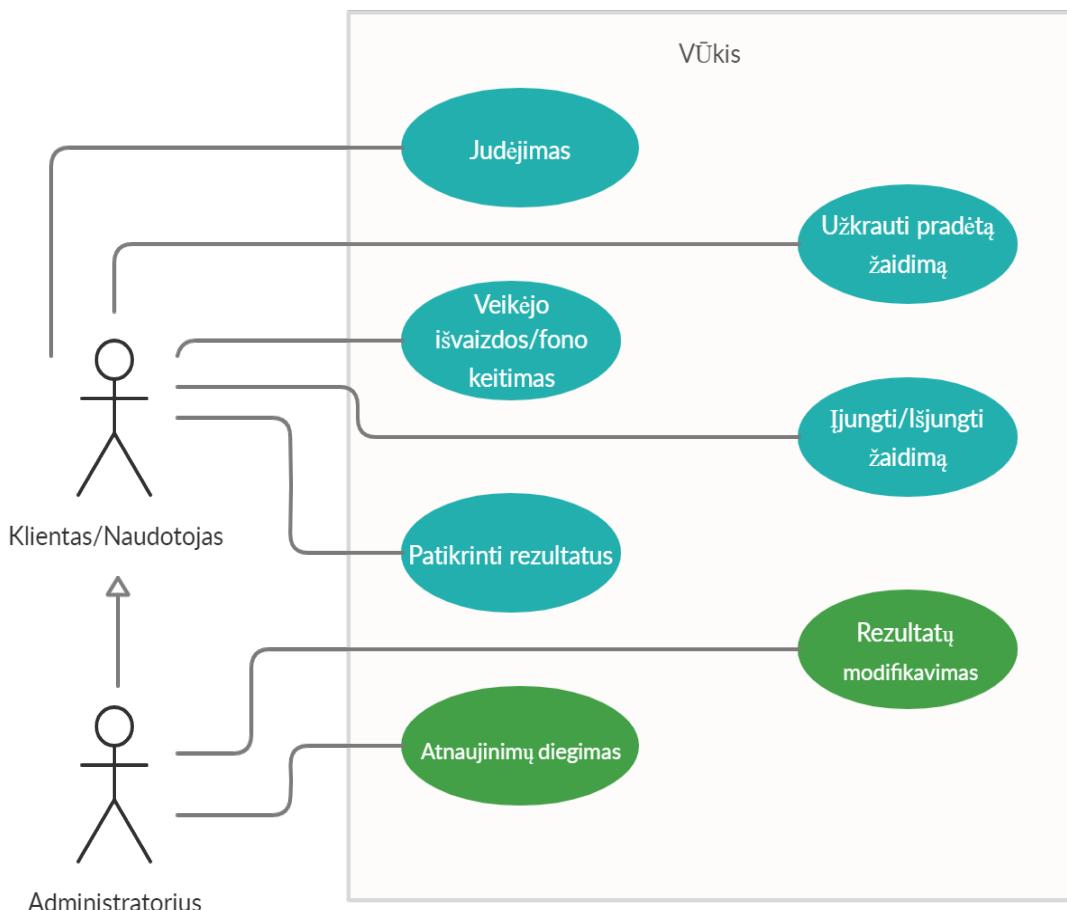
2019 GRUODIS						
SUN	MON	TUE	WED	THU	FRI	SAT
1	2 I žaidimo išbandymas	3	4	5	6 Klaidų taisymas	7
8	9	10	11 II žaidimo išbandymas	12	13	14
15	16	17	18	19	20 Galutinis žaidimo koregavimas	21
22	23 Projekto pristatymas	24	25	26	27	28
29	30	31				

12 pav. Darbo grafikas 4

4. Sistemos dizainas

Šiame punkte trumpai apžvelgsime kuriamo žaidimo architektūrą.

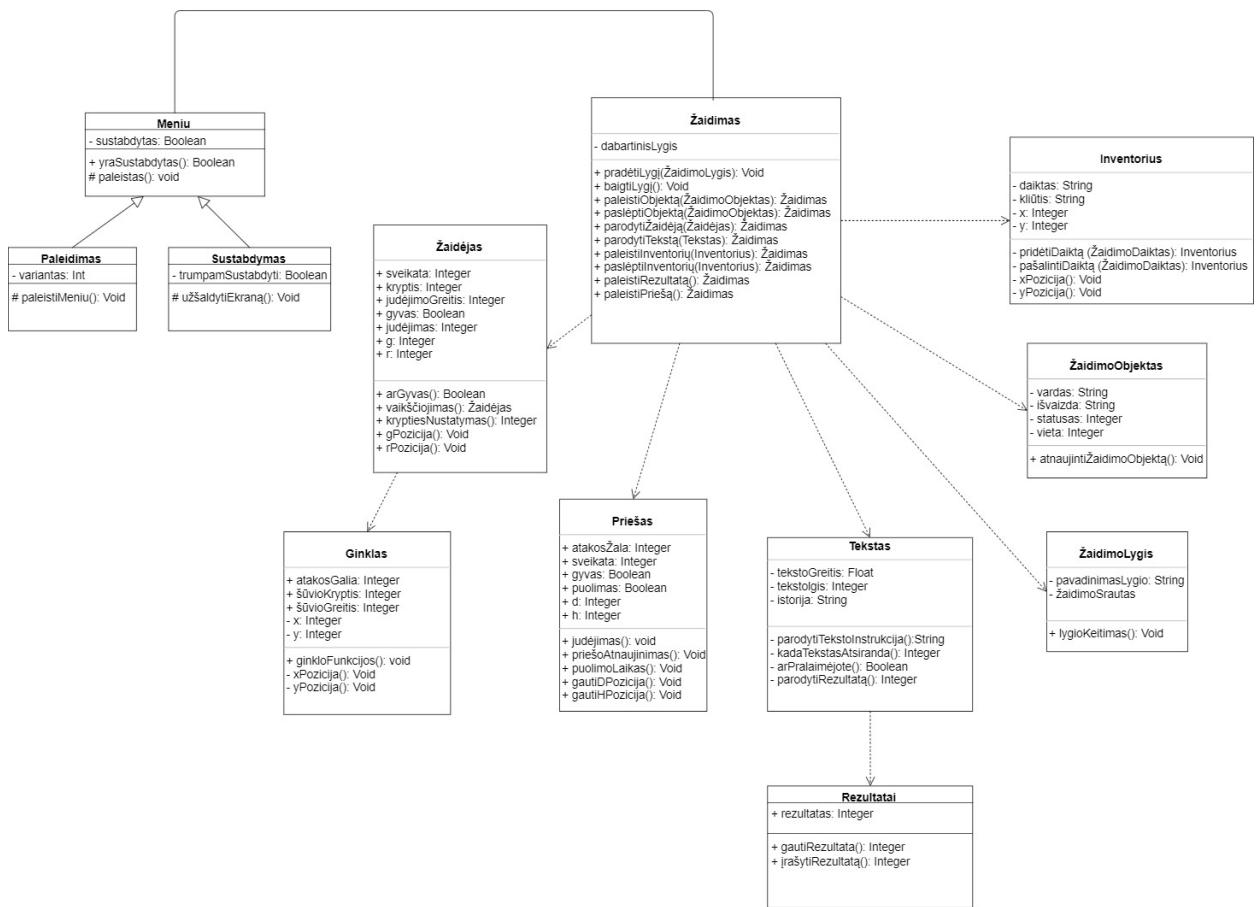
4.1. Sistemos užduočių diagrama



13 pav. Sistemos užduočių diagrama

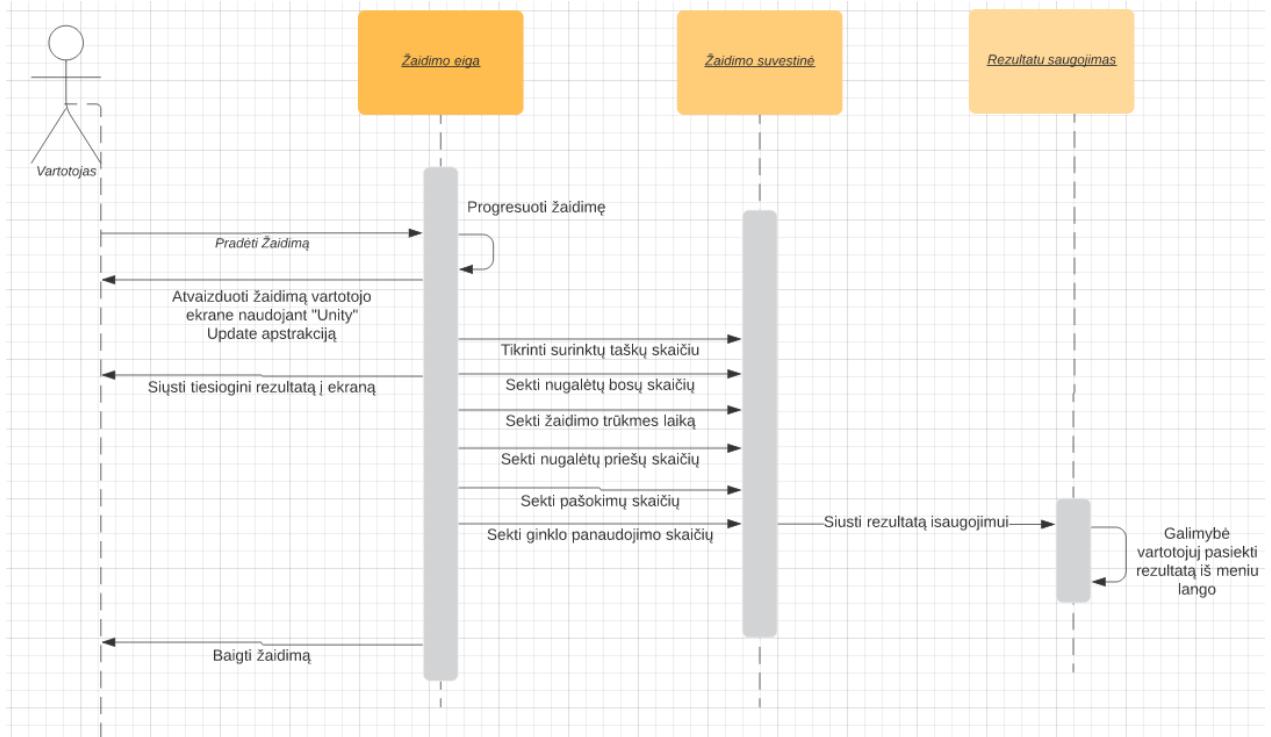
Pateikiam UML sistemos užduočių diagrama, aprašanti, ką projektuojama sistema gali atlikti. Taip pat kurioje aiškiai matosi kiekvieno aktoriaus sąveika su sistema. Būtent šioje sistemos užduočių diagramoje galime matyti, kad aktorius „Klientas/Naudotojas“ papildo „Administratorius“ aktorių.

4.2. UML diagrama



14 pav. UML klasių diagrama

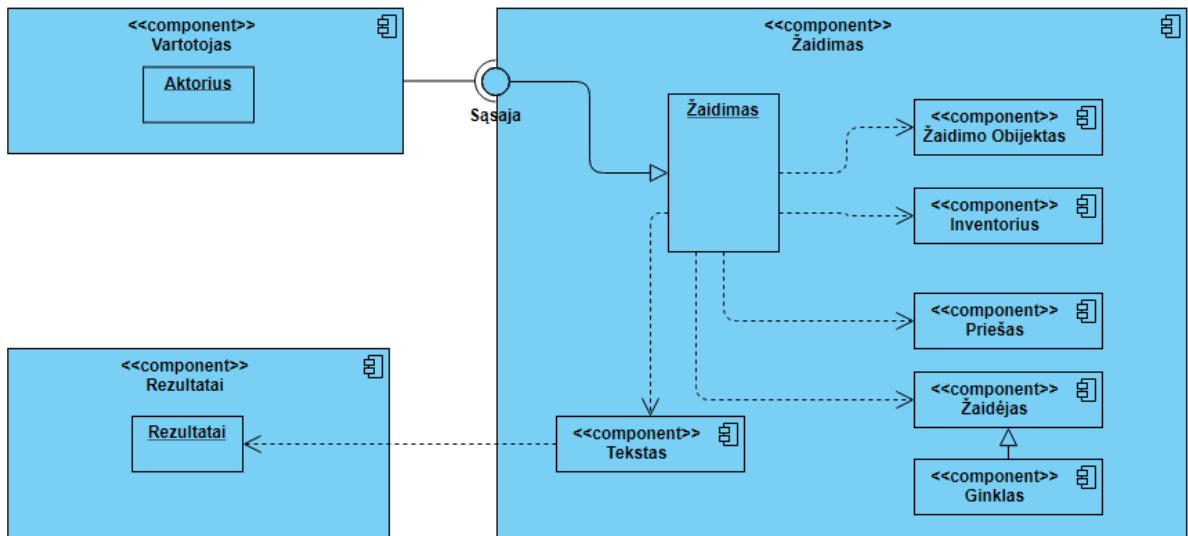
Šioje UML klasių diagramoje vaizduojami klasių rinkiniai ir sąsajos, kartu su ryšiais, esančiais tarp klasių ir sąsajų. Pateiktoje UML klasių diagramoje galime matyti, kad tarp „Meniu“ klasės ir „Žaidimo“ klasės yra asociacija. Klasė „Inventorius“, kuri priklauso nuo „Žaidimas“ klasės, parodo, „Unity3D“ specifiką, nes yra aprašomi koordinačių atributai ir metodai, kaip keisis inventorius, esančio žaidėjo rankose, pozicija. Taip pat dar viena „Unity3D“ specifika, kad turint public atributus ar metodus, juos galima keisti „Unity3D“ programos inspektoriuje, kas yra išties patogu. Dar šioje UML klasių diagramoje galime matyti, kad klasės „Paleidimas“ ir „Sustabdymas“ yra klasės „Meniu“ vaikai, kurie papildo parent klasę.



15 pav. Sekos diagrama

Sekos diagrama yra sukurta taip, kad atspindėtų laiko juostą, kuri prasideda diagramos viršuje ir nuosekliai atvaizduoja visą sąveiką seką iki pabaigos. Taip pat ji rodo perduodamas žinutes iš vieno dalyvio į kitą.

4.3. UML komponentų diagrama



16 pav. UML komponentų diagrama

UML komponentų diagrama suteikia sistemos fizinį vaizdą, rodo struktūrą bei priklausomybes tarp sistemos komponentų, įtraukiant pirminio kodo komponentus, dvejetainio kodo komponentus

bei vykdomuję programų komponentus.

5. Technologijos

Šiame punkte pateikta informacija apie tai, kodėl pasirinkome kurti programą su vienais ar kaitais ištekliais, kodėl jie yra naudingi būtent mūsų kuriamam žaidimui. Taip pat yra kalbama apie pasirinktą veiklos aplinką.

5.1. Technologijų paketas

Žaidimą kurti nusprendėme su „Unity3D“ programa, nes šioje platformoje yra lengva adaptuotis, suprasti visus pateikiamus ypatingumus. Taip pat, jeigu iškiltų kokių nors neišsprendžiamų bėdų su programa, visuomet galima kreiptis į „Unity“ techninį aptarnavimo skyrių. Kaip kompliatorių pasirinkome programą „Visual Studio“, nes šią programą rekomenduoja naudoti pats "Unity". Taip pat ši programa turi labai patogią programavimo aplinką. Kaip programavimo kalbą pasirinkome C# kalbą, nes šia kalba yra parašytas „Unity“ žaidimų variklis, todėl ir žaidimo skriptai turi būti parašyti C# kalba.

Programoje naudojamos bibliotekos:

- **UnityEngine** - be šios pagrindinės „Unity“ bibliotekos nebūtų įmanoma naudoti visų suteikiamu funkcijų ar palengvinimų;
- **UnityEngine.TextCore** - ši biblioteka leidžia naudoti teksto redaktorių, pakeisti teksto poziciją, dydį;
- **UnityEditor** - ši biblioteka leidžia naudotis kitomis, papildomomis UnityEditor bibliotekomis, tokiomis kaip UnityEditor.UI, UnityEditor.Animations;
- **System.Collections** - ši biblioteka turi sąsajų ir klasių, kuriose yra begalės objektų kolekcijų, sąrašų, masyvų, direktorių, dėstymo lentelių (angl. hash tables);

5.2. Veikimo aplinka

Kuriant žaidimą koncentruosimės į tai, kad jis veiktu „Windows“ operacinėje sistemoje, ne senesnėje nei „Windows 7“ versijoje, nes senesnių versijų nepalaiko pats „Microsoft“. Šis sprendimas yra grįstas tuo, kad beveik visose bibliotekose esančiuose kompiuteriuose, naujai nupirktuose nešiojamuose kompiuteriuose dažniausiai yra įdiegta būtent „Windows“ operacinė sistema, o mūsų tikslas yra sukurtą produktą padaryti kuo įmanoma labiau prieinamą vaikams bei moksleiviams.

6. Kūrimo īrankiai

Kaip kūrimo īrankius naudojame „Visual Studio“ ir „Unity“ bendradarbiavimo variklį. Pagrindiniam žaidimo elementų programavimui naudojame „Visual Studio“, nes ši aplinka padeda greit aptikti klaidas kode ir pagerina jo kokybę. Labai naudingas īrankis yra "Unity" bendradarbiavimo (angl. collab) variklis, kuris mums padeda dirbtį prie žaidimo kartu nuotoliniu būdu ir vis dalintis progresu su kitais komandos nariais be didelių problemų ir atskirų failų siuntinėjimų ar naudojant kitą panašaus veikimo programą. Daug laiko nereikalaujantiems klausimams, paprasčiams planams, susitarimams ar paprastai komunikacijai naudojame socialinio tinklo „Facebook“ programelę „Messenger“.

6.1. Versijų kontroles sistema

Kaip versijų kontrolės sistemą naudojame jau minėtą „Unity“ bendradarbiavimo variklį, kuris padeda suprasti kas, kada ir kokius pakeitimus atliko. Todėl iškilus problemoms ar atsiradus naujų idėjų galime keisti bei tobulinti žaidimą, nes ši sistema mums puikiai padeda grįžti prie senų ar naujų žaidimo versijų. Taip pat komandoje dirbame ant vienos kodo atšakos, o ne kelių. Tai reiškia jog atskirų projekto atšakų nėra. Ateityje planuojamieems atnaujinimams taip pat norėtumėm likti prie tokios pat sistemas.

6.2. Programinio kodo apžvalga

Programinį kodą, kurį implementuojame į žaidimą, peržvelgiame visi, taip bandydamai dar labiau jį optimizuoti ir pasiekti geresnių rezultatų. Išsiaiškinti kas, kiek ir ką pridėjo ar pakeitė kode - naudojame „Facebook“ programelę „Messenger“, kurioje visada pasidaliname padarytais atnaujinimais bei pakeitimais. Kodą stengiamės rašyti tvarkingai ir suprantamai (žaidimo failuose galima rasti tekstinį failą, kuriame yra informacija apie mūsų kodo rašymo taisykles ir standartus. Taip prisijungus naujam nariui jau sukurtas mūsų kodas bus lengvai suprantamas ir skaitomas, padės greičiau aptikti ir išspręsti problemas). Komentarai kode nėra privalomi, todėl juos naudojame tik svarbiems aspektams pabrėžti. Pasiekti kodą galés kiekvienas, kuris bus atsiuntęs mūsų žaidimą, visi failai bus žaidimo aplanke. Taip skatinime naudotojus kurti, galbūt pridėti jų norimą funkcionalumą. Norime atkreipti dėmesį, jog kodas, priešingai nei pats žaidimas, parašytas anglų kalba, nes kode naudojamos funkcijos iš „Unity“ bibliotekos naudoja angliškus raktažodžius. Todėl, kad nebūtų sunku orientuotis kode, komentarus taip pat nusprendėme rašyti anglų kalba.

6.3. Kūrimo ir implementavimo modelis

Mūsų projektas sukurtas naudojant „Unity“ kūrimo mechanizmą, kuris naudotojo sukurtą funkcionalamą bei aplinką paverčia į naudotojui lengvai paleidžiamą programą, kuri padeda žaidimui mėgautis greitai ir patogiai. Žaidimą taip pat galima sukurti ir startuoti iš komandinės eilutės, tačiau tam naudotojas turi tūrėti atsisuntes „Unity“ įrangą ir prieigą prie projekto aplanko, kurį galimybę atsiisiusti turi tik komandos nariai. Tai reiškia, jog paprastam naudotojui ši galimybė nebus prieinama, nes naudotojai atsiisiųs jau sukurtą žaidimo versiją. Tačiau prisijungus naujam komandos nariui ši galimybė bus prieinama. Tam, jog naujas komandos narys turėtų prieigą prie projekto turinio, jis turi susikurti „Unity“ anketą arba prisijungti prie „Unity“ su savo „Google“ paskyra ir

komandos lyderis pridėtų jį prie komandos narių sąrašo. Komandinės eilutės kodas bus patalpinamas projekto aplanke. Projektui atsinaujinus t.y. įdiegus atnaujinumus, procesą reiktu pakartoti iš naujo.

6.4. Testavimo automatizavimas

Sistemą testuojame ją kurdami, t.y. stengiamės vis naujai įdėtą funkcionalumą pritaikyti, kad nepriekaištingai veiktu iš karto. „Unity“ neturi labai lengvai prieinamo automatinių testų kūrimo funkcijų, todėl testus atliekame patys rankiniu būdu bandydamis žaidimą ir šalindami kliūtis, su kuriomis susiduriame. Kad žinotumėme, kaip tinkamai atliki programos testus ar nieko ne-pamirštume, esame susikūrė trumpą testavimo atmintinę (ją galima rasti žaidimo failuose arba app.anasa.com). Tačiau ateityje planuojame išmèginti „Unity“ siūlomą „Test runner“ funkcionalumą ir pritaikyti automatizuotą testavimą.

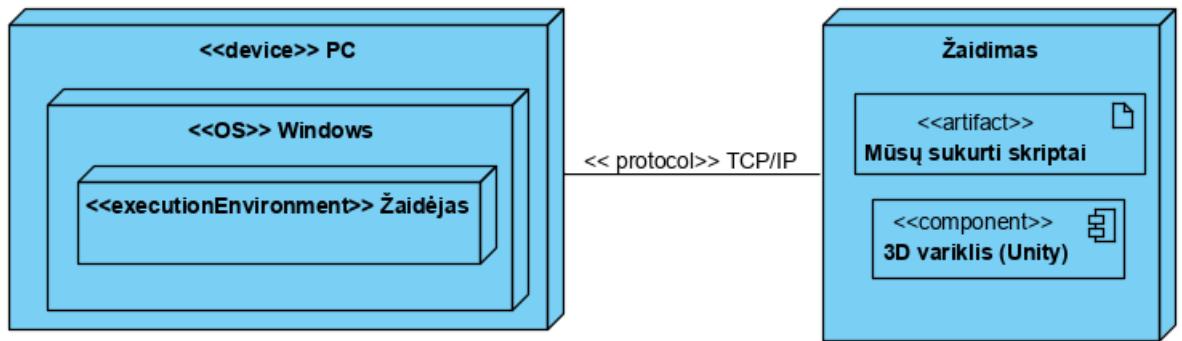
6.5. Darbo planavimas

Darbo planavimui naudojame „Asana“ platformą, kuri yra skirta projektų užduočių planavimui, bendravimui, komentavimui ir progreso stebėjimui. Ten mes kuriame užduotis, kurios yra matomas visiems komandos nariams su trumpais aprašymais, kaip tą užduotį ivykdyti. Nors naujodame nemokamą programos versiją, ji suteikia daug reikalingų ir naudingų funkcijų. Kiekvienas iš komandos narių yra informuojamas el.laišku, kai kuri nors iš užduočių yra atlanka. Taip progresą galime sekti visada. Prieinamos funkcijos duoda galimybę matyti jau įvykdytas užduotis, todėl naujam komandos nariui prisijungus būtų galima greit susipažinti su tuo, kas jau padaryta ir ką dar reikia atlankti.

7. Paleidimas

Šiame punkte pateikta paleidimo diagrama (angl. "Deployment diagram"), kuri parodo kaip sistema veikia ja paleidus ir kaip saveikauja vykdomojo laiko apdorojimo mazgai su nuo jų priklausomais komponentais.

7.1. Paleidimo diagrama



17 pav. UML paleidimo diagrama