



Kauno technologijos universitetas
Informatikos fakultetas

Architektūros specifikacija

Tadas Laurinaitis
Projekto autorius

Dr. Tomas Blažauskas
Vadovas

Kaunas, 2024

Turinys

1.	Įvadas	3
1.1	Dokumento paskirtis	3
1.2	Apibrėžimai ir sutrumpinimai	3
1.3	Apžvalga	3
2.	Architektūros pateikimas	3
3.	Architektūros tikslai ir apribojimai	3
4.	Panaudojimo atvejų vaizdas	4
5.	Sistemos statinis vaizdas	7
5.1	Apžvalga	7
5.2	Paketų detalizavimas	8
5.2.1	Paketas „Kameros valdymas“	8
5.2.2	Paketas „Įvesties valdymas“	8
5.2.3	Paketas „Scenų valdymas“	8
5.2.4	Paketas „UI“	9
5.2.5	Paketas „Logikos valdymas“	10
5.2.6	Paketas „Dirbtinio intelekto modeliai“	10
6.	Sistemos dinaminis vaizdas	11
6.1	Veiklos diagramos	12
6.2	Būsenų diagramos	14
6.3	Sekų diagramos	14
7.	Išdėstymo (deployment) vaizdas	17
8.	Duomenų vaizdas	17
9.	Kokybė	17
10.	Nuorodos	18

1. Įvadas

1.1 Dokumento paskirtis

Šio dokumento paskirtis yra apibrėžti detaliją, kuriame projekto – strateginio žaidimo, architektūros specifikaciją. Šiame dokumente pateikiamas projekto išanalizuotos reikiamos logikos architektūrinis aprašas tolimesniems projekto vykdymo darbams. Dokumento vartotojai – projektą vykdančios programuotojai ir architektai. Dokumentas vartotojams padės sklandžiai komunikuoti tarpusavyje dėl architektūrinių sprendimų.

1.2 Apibrėžimai ir and sutrumpinimai

NPC (angl. *Non-playable character*) – kompiuterinio žaidimo veikėjas, kurį valdo kompiuteris.

AI (angl. *Artificial Intelligence*) – dirbtinis intelektas.

1.3 Apžvalga

Šis architektūros specifikacijos dokumentas pateikia projekto metu atliktus architektūrinius sprendimus.

2. Architektūros pateikimas - nurodomi architektūros pateikimo aprašymai ir reikalingi elementai.
3. Architektūros tikslai ir apribojimai - aprašomi projekto programinės įrangos tikslai, reikalavimai ir apribojimai.
4. Panaudojimo atvejų vaizdas – pateikiama panaudojimo atvejų diagrama ir aprašomas kiekvienas panaudojimo atvejis lentelių forma.
5. Sistemos statinis vaizdas – sistemos išskaidymas paketų diagramomis. Kiekvienas paketas turi esminių klasių diagramą ir trumpą aprašymą.
6. Sistemos dinaminis vaizdas – sistemos esminio funkcionalumo aprašymas bei specifikavimas naudojantis veiklos, būsenų ir sekų diagramomis.
7. Išdėstymo vaizdas – aprašymas ir sistemos komponentų diagrama.
8. Duomenų vaizdas – tuščias skyrius, kadangi strateginio žaidimo sistema nenaudos tradicinių duomenų bazių.
9. Kokybė – nurodomas galimas projekto metu kuriamos strateginio žaidimo sistemos išplečiamumas ir kokybės užtikrinimo būdai.

2. Architektūros pateikimas

Architektūros specifikacija pateikiama naudojantis žmonėms skaitomu tekstu, aprašytu Microsoft Word įrankiu, bei UML kalbos diagramomis, naudojant MagicDraw grafinio atvaizdavimo įrankį.

Šiuo būdu bus pateikiami žemiau nurodyti vaizdai su diagramomis:

- Panaudojimo atvejų vaizdas:
 - Panaudojimo atvejų diagrama
- Statinis vaizdas:
 - Paketų diagrama
 - Klasių diagramomis.
- Dinaminis vaizdas:
 - Veiklos diagramomis
 - Būsenų diagrama
 - Sekų diagramomis.
- Išdėstymo vaizdas:
 - Išdėstymo diagrama.

3. Architektūros tikslai ir apribojimai

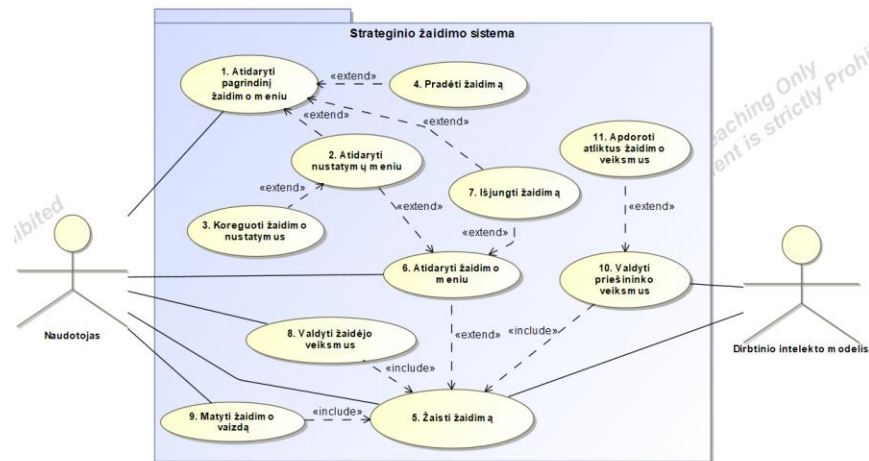
Reikalavimai, įtakojantys projekto architektūrą:

- Sistema turi būti pritaikyta įrenginiams naudojančioms Windows 10 arba naujesnę Windows operacinę sistemą
- Žaidimo grafinėje sąsajos elementai turi įsijungti per mažiau nei 1s nuo paspaudimo.
- Dirbtinio intelekto modeliai ar jų kombinacijos sugeba priimti tinkamus sprendimus žaidimo metu.

- Žaidimo valdymas turi būti greitas ir sklandus, o naudotojo valdomi objektai turi valdytis be trikdžių.
- Žaidimą turi būti patogu žaisti tiek naudotojams naudojantiems nešiojamus kompiuterius, tiek stacionarius kompiuterius.
- Du pirmus žaidimo gyvavimo metus bus leidžiami atnaujinimai bei žaidimo sistemos tvarkymai.
- Žaidimo duomenys neturi būti galimi pakeisti iš išorės (neturi veikti sukčiavimo programos)

4. Panaudojimo atvejų vaizdas

Žemiau pateiktame paveikslėlyje (1 pav.) pateikiami esminiai projekto sistemos panaudojimo atvejai ir jų ryšiai. Po paveikslėliu pateiktose lentelėse (1-11 Lentelės), tie patys panaudojimo atvejai ir ryšiai tarp jų yra aprašomi detaliau.



1 pav. Strateginio žaidimo panaudojimo atvejų diagrama

1 Lentelė. Panaudojimo atvejis nr. 1

1. PANAUDOJIMO ATVEJIS: Atidaryti pagrindinį žaidimo meniu

Vartotojas/Aktorius: Naudotojas

Aprašas: Atidaromas pagrindinis žaidimo meniu.

Prieš sąlyga: Naudotojas įjungė žaidimą.

Sužadinimo sąlyga: Naudotojas paspaudė pagrindinio žaidimo meniu įjungimo mygtuką.

Po-sąlyga: Atidaromas pagrindinis žaidimo meniu

2 Lentelė. Panaudojimo atvejis nr. 2

2. PANAUDOJIMO ATVEJIS: Atidaryti nustatymų meniu

Vartotojas/Aktorius: Naudotojas

Aprašas: Atidaromas nustatymų meniu langas.

Prieš sąlyga: Naudotojas atidarė pagrindinį žaidimo meniu.

Sužadinimo sąlyga: Naudotojas paspaudė nustatymų meniu įjungimo mygtuką.

Po-sąlyga: Atidaromas nustatymų meniu langas.

3 Lentelė. Panaudojimo atvejis nr. 3

3. PANAUDOJIMO ATVEJIS: Koreguoti žaidimo nustatymus

Vartotojas/Aktorius: Naudotojas

Aprašas: Koreguojami žaidimo nustatymai – garso, grafikos ir sunkumo lygiai.

Prieš sąlyga: Naudotojas atidarė nustatymų meniu langą.

Sužadinimo sąlyga: Naudotojas pakeitė garso, grafikos arba sunkumo lygį naudodamasis grafine sąsaja.

Po-sąlyga: Žaidimo nustatymai pakeičiami į naujai pasirinktus.

4 Lentelė. Panaudojimo atvejis nr. 4

4. PANAUDOJIMO ATVEJIS: Pradėti žaidimą

Vartotojas/Aktorius: Naudotojas

Aprašas: Pradedamas žaidimas.

Prieš sąlyga: Naudotojas atidarė pagrindinį žaidimo meniu.

Sužadinimo sąlyga: Naudotojas paspaudė žaidimo pradėjimo mygtuką.

Po-sąlyga: Įjungiamas pagrindinis žaidimo langas ir pradedamas žaidimas.

5 Lentelė. Panaudojimo atvejis nr. 5

5. PANAUDOJIMO ATVEJIS: Žaisti žaidimą

Vartotojas/Aktorius: Naudotojas, Dirbtinio intelekto modelis

Aprašas: Žaidžiamas žaidimas tarp žaidėjo ir dirbtinio intelekto.

Prieš sąlyga: Naudotojas atidarė pagrindinį žaidimo meniu.

Sužadinimo sąlyga: Naudotojas paspaudė žaidimo pradėjimo mygtuką.

Po-sąlyga: Žaidžiamas žaidimas tarp žaidėjo ir dirbtinio intelekto.

6 Lentelė. Panaudojimo atvejis nr. 6

6. PANAUDOJIMO ATVEJIS: Atidaryt žaidimo meniu

Vartotojas/Aktorius: Naudotojas

Aprašas: Atidaromas meniu žaidimo metu, leidžiantis keisti žaidimo nustatytumus ir išjungti žaidimą.

Prieš sąlyga: Naudotojas atidarė pagrindinį žaidimo meniu.

Sužadinimo sąlyga: Naudotojas paspaudė žaidimo pradėjimo mygtuką.

Po-sąlyga: Įjungiamas žaidimo meniu žaidimo metu.

7 Lentelė. Panaudojimo atvejis nr. 7

7. PANAUDOJIMO ATVEJIS: Išjungti žaidimą

Vartotojas/Aktorius: Naudotojas

Aprašas: Išjungiamas žaidimas.

Prieš sąlyga: Naudotojas atidarė žaidimo meniu.

Sužadinimo sąlyga: Naudotojas paspaudė žaidimo išjungimo mygtuką.

Po-sąlyga: Žaidimas išjungiamas.

8 Lentelė. Panaudojimo atvejis nr. 8

8. PANAUDOJIMO ATVEJIS: Valdyti žaidėjo veiksmus

Vartotojas/Aktorius: Naudotojas

Aprašas: Valdomi žaidėjo veiksmai, naudojantis įvesties įrenginiais ir atsižvelgiant į matomą žaidimo vaizdą.

Prieš sąlyga: Žaidimas yra pradėtas.

Sužadinimo sąlyga: Naudotojas paspaudė bent vieną iš registruotų žaidėjo valdymo klavišų klaviatūroje arba pelėje.

Po-sąlyga: Valdomi žaidėjo veiksmai priklausomai nuo paspaustų įvesties įrenginių klavišų.

9 Lentelė. Panaudojimo atvejis nr. 9

9. PANAUDOJIMO ATVEJIS: Matyti žaidimo vaizdą

Vartotojas/Aktorius: Naudotojas

Aprašas: Matomas žaidimo vaizdas išvesties įrenginyje (monitoriuje).

Prieš sąlyga: Naudotojas įjungė žaidimo aplikaciją.

Sužadinimo sąlyga: -

Po-sąlyga: Matomas visas žaidimo vaizdas bei pasinkti meniu.

10 Lentelė. Panaudojimo atvejis nr. 10

10. PANAUDOJIMO ATVEJIS: Valdyti priešininko veiksmus

Vartotojas/Aktorius: Dirbtinio intelekto modelis

Aprašas: Valdomi priešininko veiksmai atsižvelgiant į apdorotus jau atliktus žaidimo veiksmus.

Prieš sąlyga: Žaidimas yra pradėtas.

Sužadinimo sąlyga: Naudotojas paspaudė žaidimo pradėjimo mygtuką.

Po-sąlyga: Priešininkas yra valdomas atsižvelgiant į veiksmus, kuriuos reikia atlikti.

11 Lentelė. Panaudojimo atvejis nr. 11

11. PANAUDOJIMO ATVEJIS: Apdoroti atliktus žaidimo veiksmus

Vartotojas/Aktorius: Dirbtinio intelekto modelis

Aprašas: Apdorojami žaidėjo ir dirbtinio intelekto modelio atlikti veiksmai bei apskaičiuojami geriausi ir naudingiausi sekantys veiksmai.

Prieš sąlyga: Žaidimas yra pradėtas.

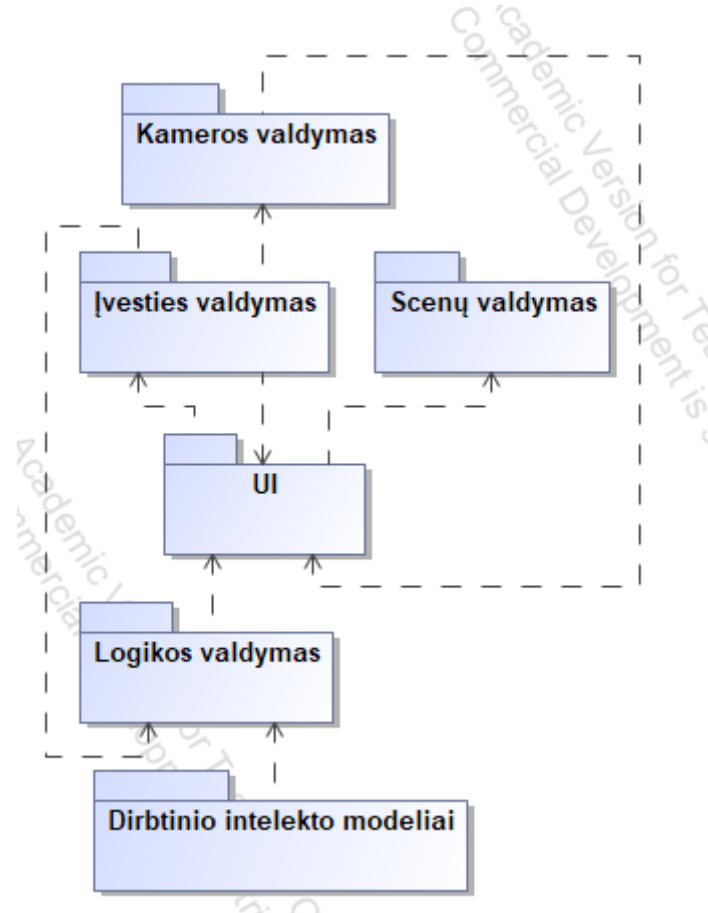
Sužadinimo sąlyga: Naudotojas paspaudė žaidimo pradėjimo mygtuką.

Po-sąlyga: Pastoviai apdorojama žaidėjo ir dirbtinio intelekto modelio veiksmų informacija ir parenkami geriausi ir naudingiausi sekantys veiksmai.

5. Sistemos statinis vaizdas

5.1 Apžvalga

Projekto statinis vaizdas skiriamas į paketus, matomus žemiau pateiktame paveikslėlyje (**2 pav.**). 5.2 skyriuje “Paketų detalizavimas”, šie paketai bus detalizuojami aprašymais ir klasių diagramomis.

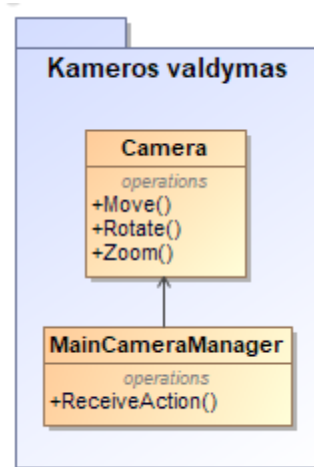


2 pav. Sistemos paketų diagrama

5.2 Paketų detalizavimas

5.2.1 Paketas „Kameros valdymas“

Šio paketo klasės yra atsakingos už pagrindinės žaidimo kameros valdymą. Paketo klasių struktūra pateikiama žemiau esančiame paveikslėlyje (3 pav.).

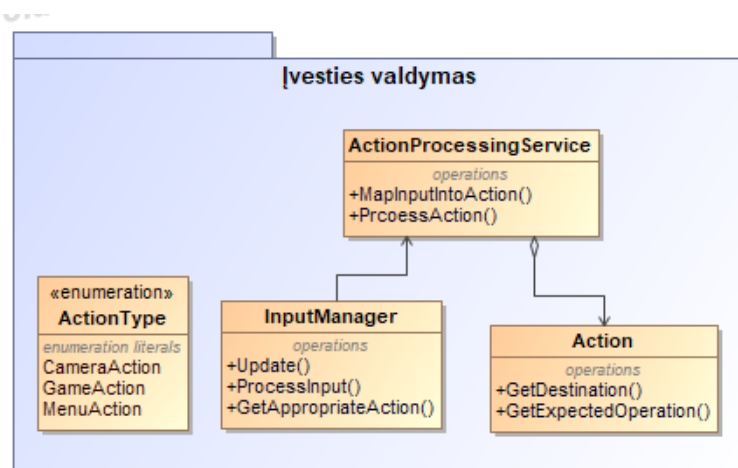


3 pav. Paketo „Kameros valdymas“ klasių diagrama

Paveikslėlyje parodytos klasės atlieka šias funkcijas: žaidėjo įvesties gavimas, kameros pozicijos keitimas, kameros rotacijos keitimas ir kameros vaizdo padidinimas.

5.2.2 Paketas „Įvesties valdymas“

Šio paketo klasės yra atsakingos už žaidėjo įvesties signalų, pateikiamų įvesties įrenginiais, apdorojimą. Paketo klasių struktūra pateikiama žemiau esančiame paveikslėlyje (4 pav.).



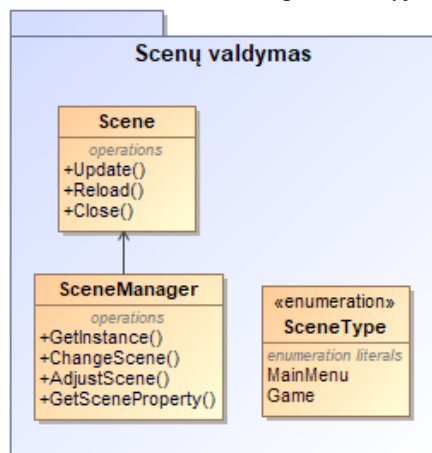
4 pav. Paketo „Įvesties valdymas“ klasių diagrama

Aukščiau pateiktame paveikslėlyje parodytos klasės atlieka šias funkcijas: gauna žaidėjo įvestį, atskiria įvesties tipą, prilyginą šį tipą norimam atlikti veiksmui ir galiausiai siunčia veiksmus į kituose paketuose esančius valdiklius.

5.2.3 Paketas „Scenų valdymas“

Šio paketo klasės yra atsakingos žaidimo scenų, tokių kaip pagrindinis meniu ar pagrindinis žaidimo langas,

valdymą. Paketo klasių struktūra pateikiama žemiau esančiame paveikslėlyje (**5 pav.**).

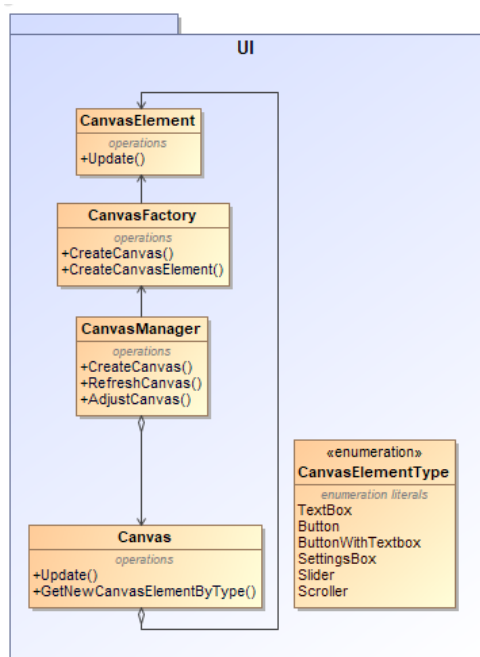


5 pav. Paketo „Scenų valdymas“ klasių diagrama

Aukščiau pateiktame paveikslėlyje parodytos klasės atlieka šias funkcijas: pakeičia žaidimo scenas, pakeičia scenų turinį, gauna scenų turinį, atnaujina arba uždaro scenas. Išskiriami du scenų tipai: pagrindinis meniu (*angl. MainMenu*) ir žaidimo pagrindinė scena (*angl. Game*).

5.2.4 Paketas „UI“

Šio paketo klasės yra atsakingos vartotojo grafinės sąsajos valdymą. Unity žaidimų variklyje, grafinės vartotojo sąsajos įgyvendinimas vykdomas drobių (*angl. Canvas*) pagrindu. Kiekviena drobė turi savo elementus, kurie gali būti skirtingų tipų. Paketo klasių struktūra pateikiama žemiau esančiame paveikslėlyje (**6 pav.**).

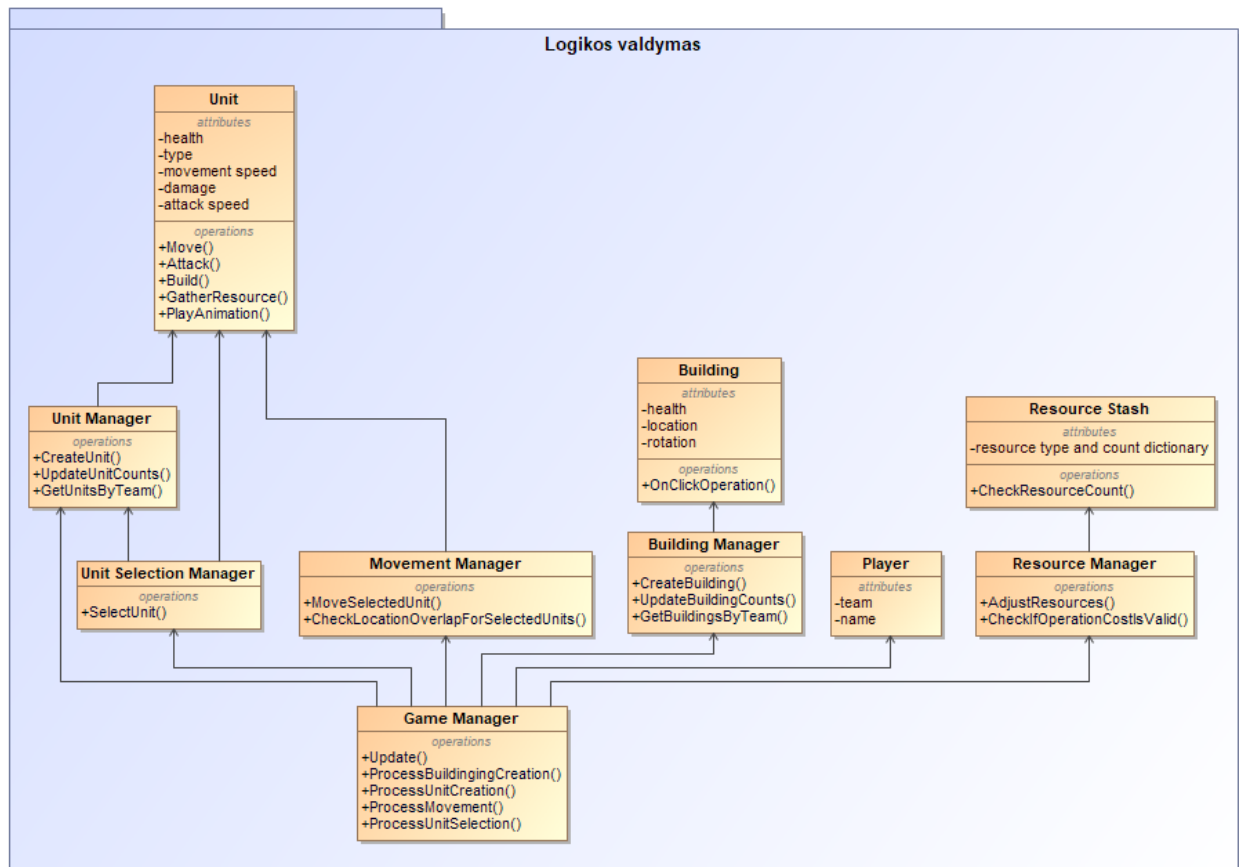


6 pav. Paketo „UI“ klasių diagrama

Aukščiau pateiktame paveikslėlyje parodytos klasės atlieka šias funkcijas: atlieka drobių kūrimą, atnaujinimą bei keitimą, drobių elementų kūrimą ir atnaujinimą. Išskiriami šeši drobių elementų tipai: tekstinis langas (*angl. TextBox*), mygtukas (*angl. Button*), mygtukas su tekstiniu langu (*angl. ButtonWithTextBox*), nustatymų langas (*angl. SettingsBox*), slankiojimo elementas (*angl. Slider*) ir slinkimo elementas (*angl. Scroller*).

5.2.5 Paketas „Logikos valdymas“

Šio paketo klasės yra atsakingos už bendros žaidimo logikos valdymą. Šiame pakete esančios klasės atlieka įvairius veiksmus su pagrindiniais žaidimo elementais – padaliniais, pastatais ir resursais. Paketo klasių struktūra pateikiama žemiau esančiame paveikslėlyje (7 pav.).



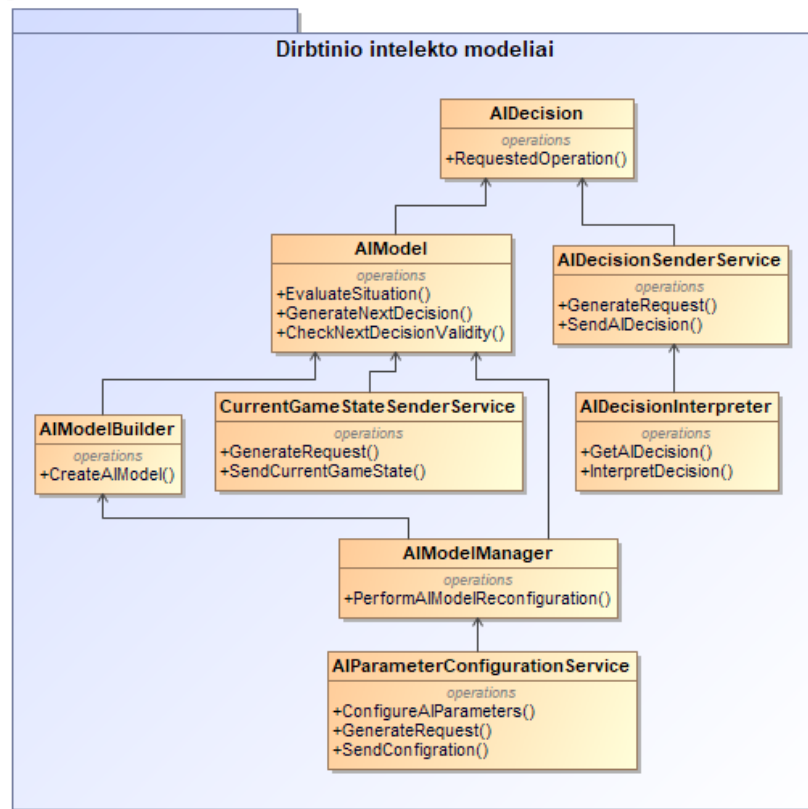
7 pav. Paketo „Scenų valdymas“ klasių diagrama

Aukščiau pateiktame paveikslėlyje parodytos klasės atlieka šias funkcijas:

- Padalinių sukūrimą, jų valdymą (judėjimą, atakavimą, resursų rinkimą, pastatų statymą) ir jų animacijų valdymą.
- Pastatų sukūrimą ir jų valdymą.
- Resursų valdymą (resursų pridėjimas ir atėmimas, operacijų su resursais tikrinimas, resursų kiekio tikrinimas).
- Žaidėjo valdymą

5.2.6 Paketas „Dirbtinio intelekto modeliai“

Šio paketo klasės yra atsakingos už dirbtinio intelekto modelių dalyvavimo žaidime valdymą. Paketo klasių struktūra pateikiama žemiau esančiame paveikslėlyje (8 pav.).



8 pav. Paketo „Dirbtinio intelekto modeliai“ klasių diagrama

Aukščiau pateiktame paveikslėlyje parodytos klasės atlieka šias funkcijas:

- Naujo dirbtinio intelekto modelio sukūrimą, sukurtą modelio konfigūraciją ir rekonfigūraciją.
- Žaidimo stadijos siuntimą sukurtam modeliui.
- Dirbtinio intelekto modelio situacijos įvertinimą, veiksmų priėmimą ir jų patikrinimą pagal dabartinę žaidimo stadiją.
- Dirbtinio intelekto modelio priimtų sekančių veiksmų siuntimą.
- Gautų dirbtinio intelekto sekančių veiksmų interpretavimą.

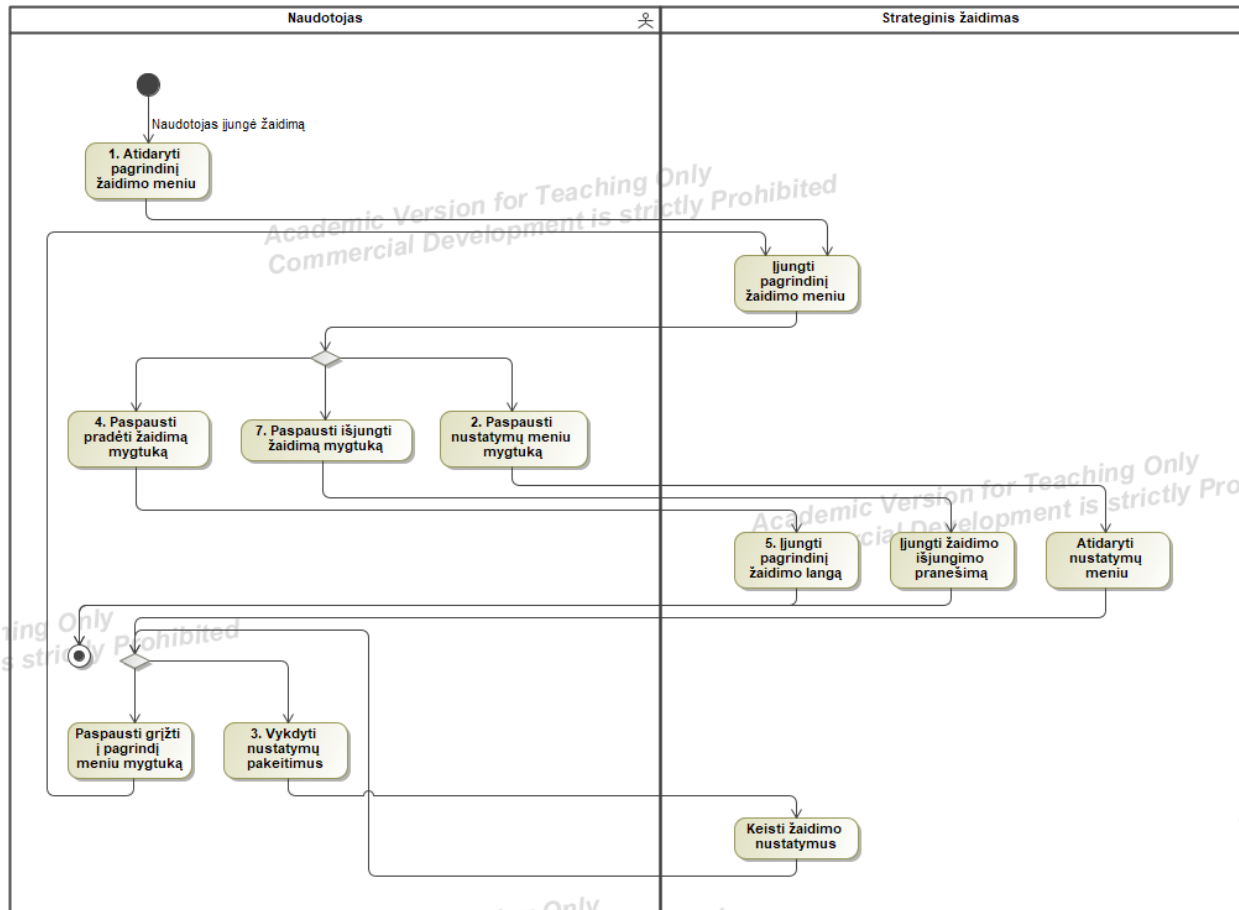
Pastaba: kai kurie dirbtinio intelekto metodai gali būti įgyvendinti Python programavimo kalba, tačiau klasių struktūra bus išlaikoma ta pati.

6. Sistemos dinaminis vaizdas

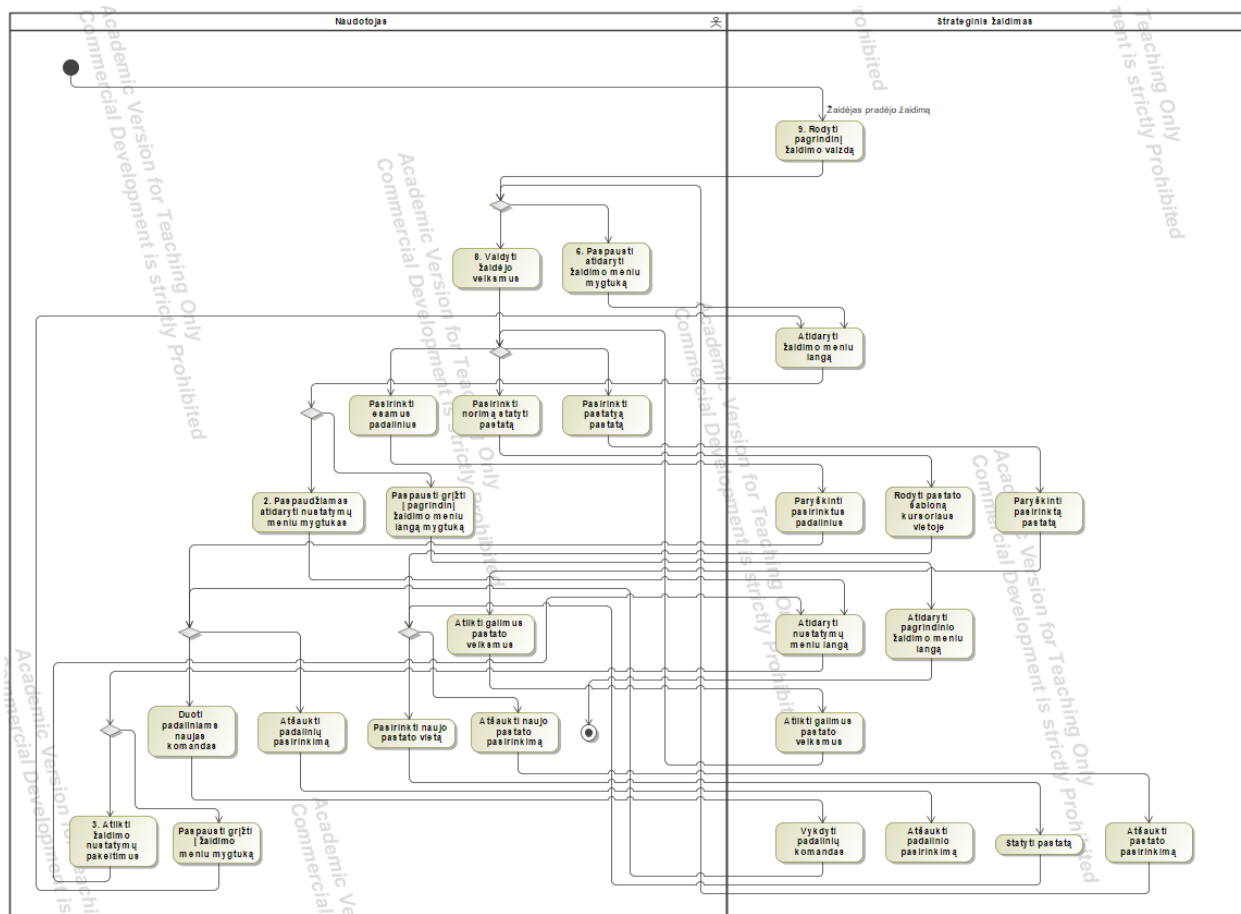
Šiame skyriuje pateikiamos veiklos, būsenų ir sekų diagramos. Diagramos palengvina bendro strateginio žaidimo veikimo proceso supratimą ir leidžia greičiau perprasti ryšį tarp skirtingų sistemos dalių.

6.1 Veiklos diagramos

Žemiau pateiktos veiklos diagramos pasirinktiems esminiams panaudojimo atvejams – žaidimo pagrindinio meniu navigacijai ir pagrindiniam žaidimo funkcionalumui, sutinkamam žaidimo metu. (9 ir 10 pav.).



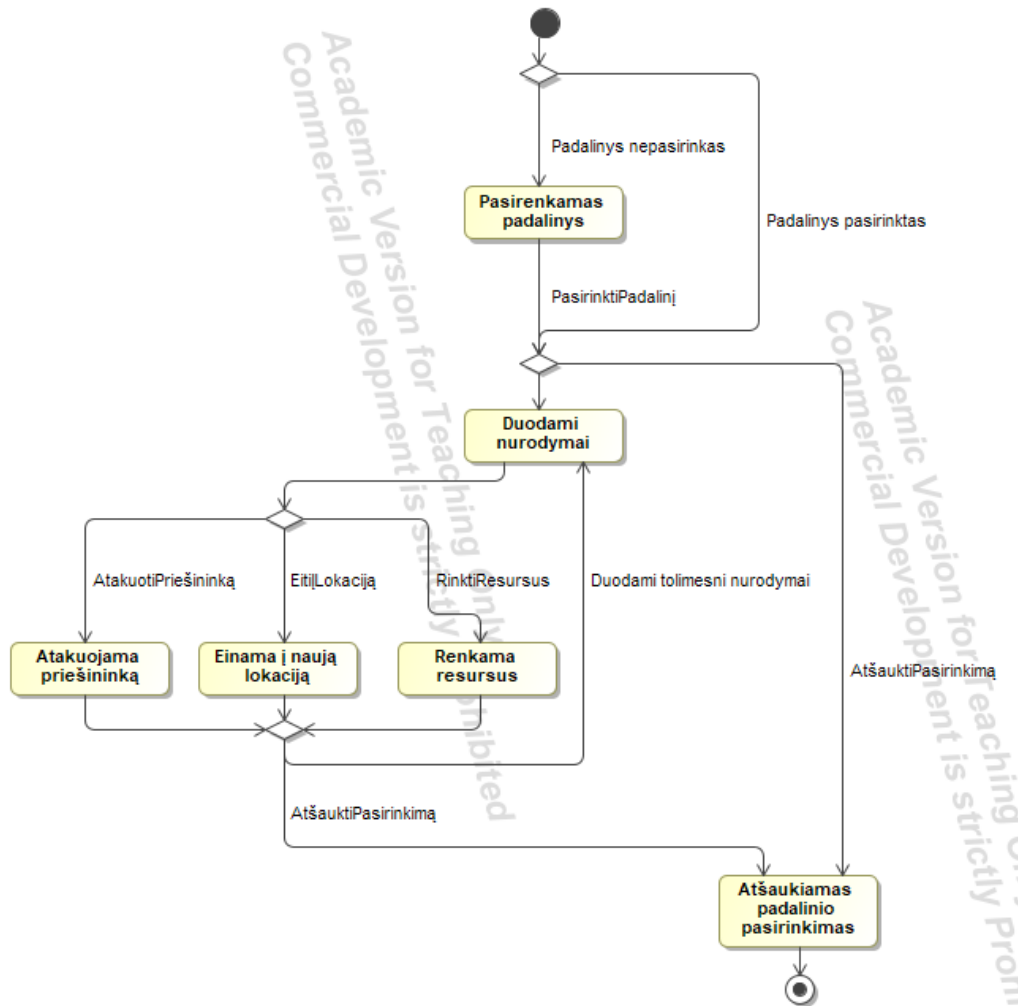
9 pav. Žaidimo pagrindinio meniu navigavimo veiklos diagrama



10 pav. Žaidimo pagrindinio funkcionalumo, sutinkamo žaidimo metu, veiklos diagrama

6.2 Būsenų diagramos

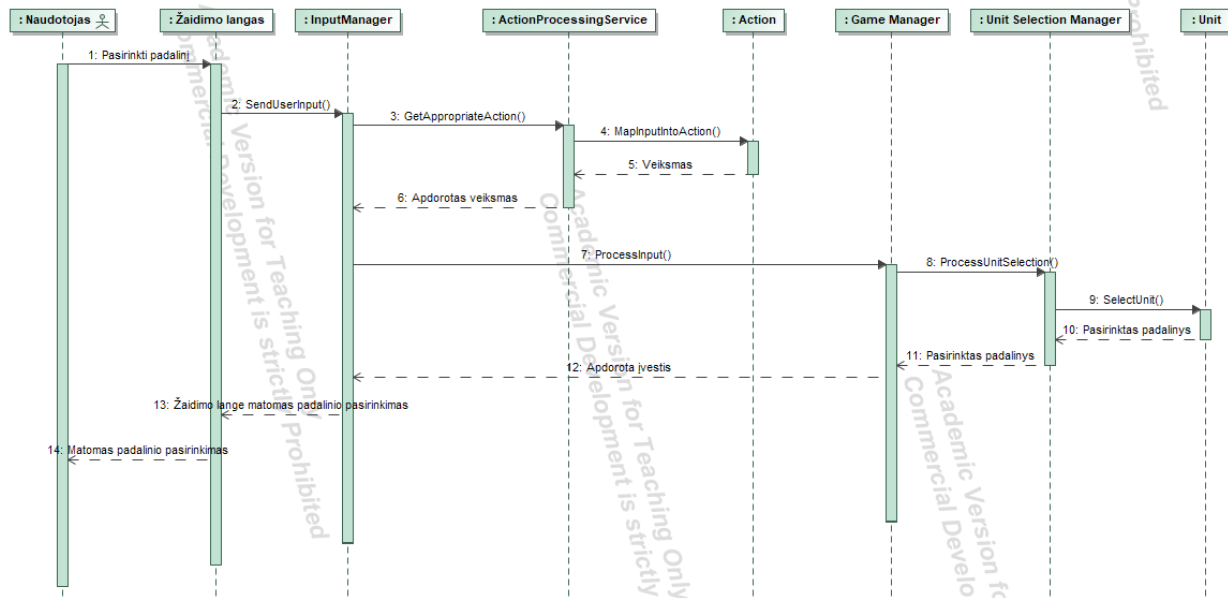
Žemiau pavaizduota būsenų diagrama (11 pav.) apibrėžia kiekvieno padalinio būsenas žaidimo metu. Padalinio būsenos kinta priklausomai nuo žaidėjo pasirinktų veiksmų su tuo padaliniu. Padalinio būsenos visada prasideda padalinio pasirinkimu, ir visada baigiasi padalinio pasirinkimo atšaukimu.



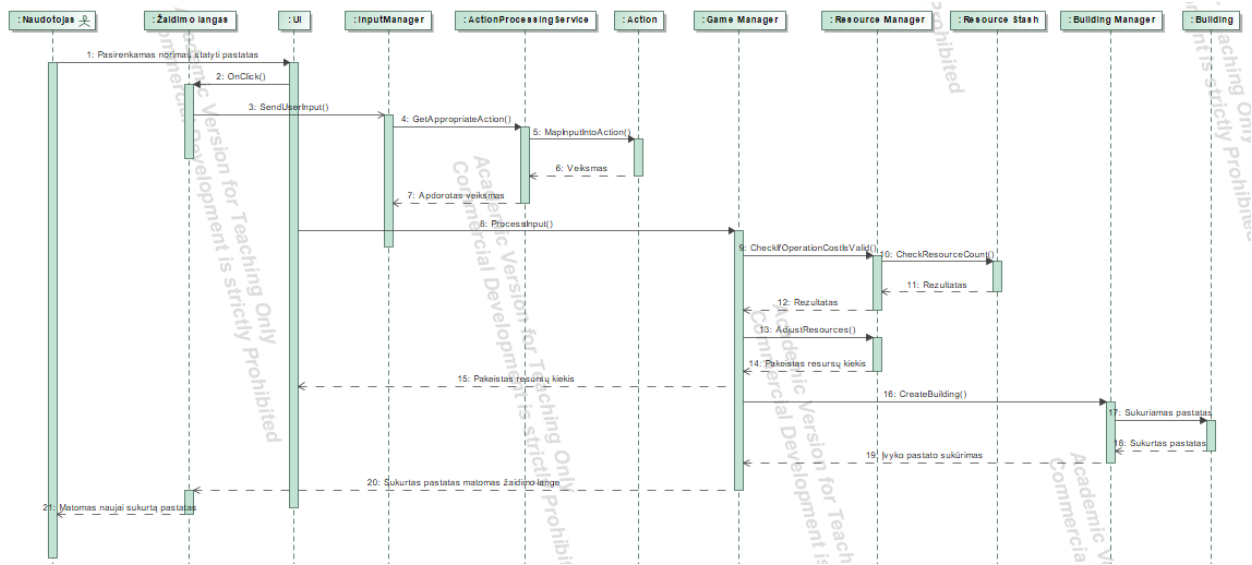
11 pav. Padalinio būsenų diagrama

6.3 Sekų diagramos

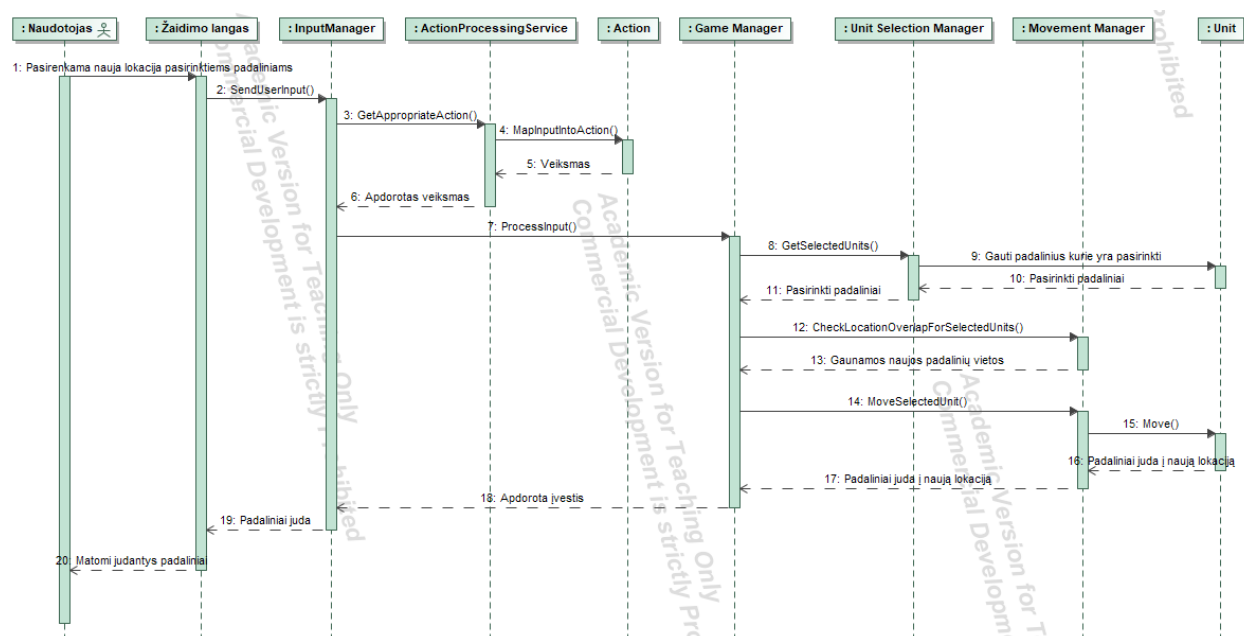
Žemiau esančiuose paveikslėliuose (12-14 pav.) pateikiamos sekų diagramos pasirinktam strateginio žaidimo funkcionalumui. Svarbu pastebėti, kad sekų diagramose neapibrėžiami elementarūs strateginio žaidimo panaudojimo atvejai, o apibrėžiamas sudėtingesnis funkcionalumas, iš kurio susideda panaudojimo atvejai „5. Valdyti žaidėjo veiksmus“ ir „5. Žaisti žaidimą“. Funkcionalumas, kuris turi beveik identišką sekų diagramas (galbūt su tam tikrais nežymiais pakitimais), kaip ir žemiau apibrėžtos sekų diagramos, taip pat nėra apibrėžtas.



12 pav. Padalinio pasirinkimo sekų diagrama



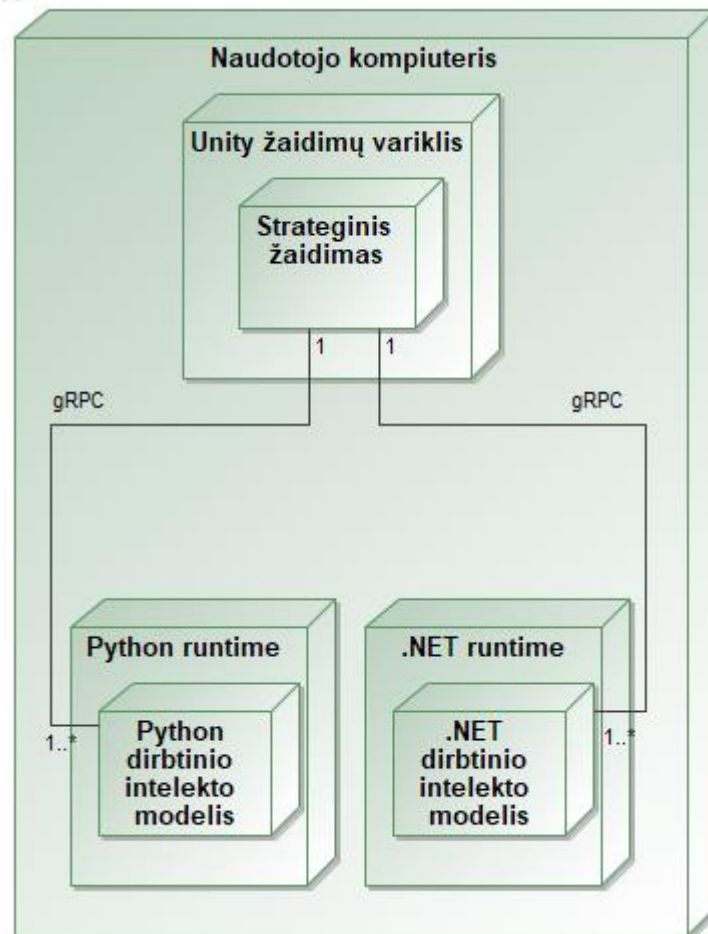
13 pav. Pastatų statymo sekų diagrama



14 pav. Padalinių judėjimo į pasirinktą lokaciją sekų diagrama

7. Išdėstymo (deployment) vaizdas

Projekto sistemos išdėstymo vaizdas pateikiamas išdėstymo diagrama. Visos žaidimo dalys veikia naudotojo kompiuteryje – strateginis žaidimas veikia Unity žaidimų variklyje, o dirbtinio intelekto modeliai – Python ir .NET runtime aplinkose. Strateginis žaidimas su dirbtinio intelekto modeliais komunikuoja naudojantis gRPC technologija duomenų perdavimui ir gavimui žaidimo metu. Strateginis žaidimas atsakingas už visą žaidimo logiką ir jos valdymą, o dirbtinio intelekto modeliai atsakingi už esamos žaidimo stadijos interpretavimą, sprendimų priėmimą ir savo priimtų sprendimų siuntimą žaidimo logikai apdoroti. Dirbtinio intelekto modeliai išskiriami į dvių tipų – Python ir .NET, kadangi projekto tikslas yra ištirti skirtingų tipų dirbtinio intelekto modelių naudojimą žaidimuose, todėl bus padengtas didesnis kiekis modelių.



8 pav. Strateginio žaidimo išdėstymo diagrama

8. Duomenų vaizdas

Projekto metu kuriama sistema – strateginis žaidimas nenaudos duomenų bazės. Žaidimo duomenys žaidimo metu bus saugomi darbinėje atmintyje (RAM) Unity žaidimo variklio pagalba automatiškai būdu. Pabaigus žaidimą, žaidimo duomenys bus išsaugomi atsikarame faile, užfiskuosiančiame paskutinę žaidimo būseną žmonėms neskaitomu formatu – sugeneruotu artefaktu.

9. Kokybė

Strateginis žaidimas leidžia bet kokių dirbtinio intelekto modelių naudojimą, su vieninteliu reikalavimu – dirbtinio

intelekto modelis turi sekti šiame dokumente apibrėžtą klasių ir išdėstymo struktūrą. Šios struktūros sekimo užtikrinimas yra būtinas dirbtinio intelekto modelio naudojimui žaidime, kadangi vidinė žaidimo logika bus tiesiogiai susieta su šia struktūra.

Strateginis žaidimas su tam tikrais nedideliais sisteminio kodo pakeitimais, nekeičiant žaidimo architektūros, taip pat teoriškai gali būti padarytas žaisti ir ant kitų platformų, palaikančių Unity žaidimo variklį, tačiau šis žingsnis nėra planuojamas.

Strateginio žaidimo kodo kokybei užtikrinti bus naudojami tiek statinės, tiek dinaminės kodo analizės įrankiai.

10. Nuorodos

[1] „Reikalavimų specifikacija“, Tadas Laurinaitis, 2024-05-03, KTU Moodle