



Kauno technologijos universitetas
Informatikos fakultetas

Testavimo planas

Tadas Laurinaitis
Projekto autorius

Doc. Šarūnas Packevičius
Dėstytojas

Kaunas, 2024

Turinys

1. Įvadas	3
1.1. Sistemos aprašymas.....	3
1.2. Panaudojimo atvejai	3
1.3. Testuojamos sistemos apimtis	3
2. Testavimo procesas ir jį sudarantys metodai	4
3. Testavimo strategija.....	4
4. Testavimo tikslai	5
5. Rankinis testavimas (<i>angl. Manual testing</i>)	5
6. Automatizuotas testavimas (<i>angl. Automated testing</i>)	5
7. Alfa ir Beta testavimas	5
7.1. Alfa testavimas	6
7.2 Beta testavimas	6
8. Testavimo elementai.....	6
10. Naudotojo sąsajos testavimas	7
11. Testavimo valdymas.....	7
12. Testavimo grafikas	8
13. Patvirtinimas.....	8
14. Testavimo rizika	8

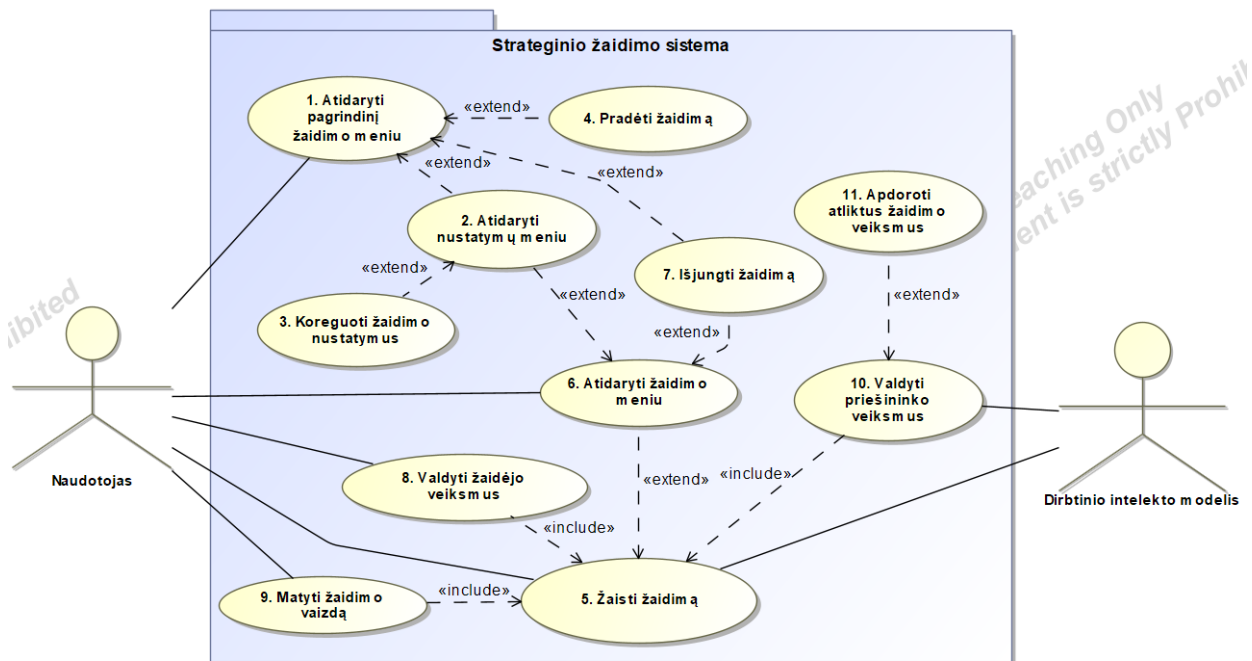
1. Įvadas

1.1. Sistemos aprašymas

Testavimo planas apibrėžiamas magistrinio projekto „Dirbtinio intelekto metodų taikymas strateginiams žaidimams“ metu kuriamam strateginiui žaidimui. Strateginio žaidimo metu žaidėjai galės naviguoti tiek pagrindiniame tiek žaidimo meniu, valdyti žaidėjo veiksmus ir stengtis įveikti dirbtinio intelekto modelių valdomą priešininką. Žaidėjo galimi atlikti veiksmai – gaminti ir duoti komandas padaliniams, statyti ir valdyti pastatus, rinkti ir valdyti resursus bei atakuoti priešininkus.

Testavimo plane apibrėžiamas testavimo procesas ir jį sudarantys metodai, strategija, tikslai, taip pat detalizuojami testavimo metodai, testavimo elementai bei naudotojo sąsajos testavimas. Sistemos testavimo atlikimo veiksmai ir veikimo korektiškumas bus nustatomas pagal šiame dokumente apibrėžtus kriterijus, procesus ir strategijas.

1.2. Panaudojimo atvejai



1 pav. Strateginio žaidimo panaudojimo atvejų diagrama

1.3. Testuojamos sistemos apimtis

Strateginis žaidimas padarytas naudojant Unity žaidimų variklį ir C# programavimo kalbą. Dirbtinio intelekto modeliai realizuoti tiek Python, tiek .NET runtime aplinkose ir su strateginiu žaidimu 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.

2. Testavimo procesas ir jį sudarantys metodai

Strateginio žaidimo sistemos testavimo procesą sudaro šios dalys:

- Vienetų testai (*angl. Unit tests*):
 - Testuojamas atskiras mažų sistemos dalių – vienetų, veikimas. Šio projekto kontekste vienetai grupuojami pagal klases, kurioms jie priklauso, o patys vienetai atitinka klasėse įgyvendintus metodus.
 - Vykdomi kartu su naujai atsirandčiomis ar besikeičiančiomis sistemos dalimis.
- Žaidžiamumo testavimas (*angl. Play testing*):
 - Testuojamas žaidimo veikimas žaidimo kūrimo metu žaidžiant žaidimą ir žiūrint ar realizuotas funkcionalumas atitinka panaudojimo atvejų diagramą ir projekto reikalavimuose apibrėžtas normas.
- Regresinis testavimas (*angl. Regression testing*):
 - Testuojama ar naujo žaidimo funkcionalumo įgyvendinimas nekenkia jau įgyvendintų žaidimo posistemių sąveikai ir greitaveikai.
- Našumo testavimas (*angl. Performance testing*):
 - Automatizuotas žaidimo našumo testavimas, siekiant užtikrinti, kad jis veikia sklandžiai įvairiomis sąlygomis.
 - Tikrinama, kaip žaidimas reaguoja į skirtingas apkrovas.
- Naudotojo sąsajos testavimas (*angl. UI testing*):
 - Automatiniai testai naudojami tikrinti naudotojo sąsajos elementų funkcionalumą.
 - Užtikrinama, kad mygtukai, meniu ir kiti UI elementai veikia taip, kaip turėtų.
- Alfa testavimas (*angl. Alpha testing*):
 - Išsamiai testuojamas įgyvendinamas funkcionalumas stengiantis atrasti defektus ir kitas problemas kuo anksčiau.
 - Šio testavimo metu atrandamos ir sutaisomos didžiausios problemos.
 - Vykdomas periodiškai po to kai įgyvendinamas esminis žaidimo funkcionalumas, tačiau prieš Beta testavimą.
- Beta testavimas (*angl. Beta testing*):
 - Vykdomas įgyvendinus didžiąją dalį pagrindinio žaidimo funkcionalumo užtikrinti, kad problemos, rastos alfa testavimo metu buvo tikrai sutaisytos.
 - Randami defektai, kuriuos ištaisius pagerės žaidžiamumas, stabilumas ir švarumas. Prireikus, bus vykdomas keliomis stadijomis.
 - Tikrinama, ar žaidimas yra tinkamai subalansuotas, įvertinama žaidėjo patirtis žaidžiant žaidimą.

3. Testavimo strategija

Dėl ribotų žmoniškųjų resursų, strateginio žaidimo sistemos testavimas bus atliekamas ją kuriančio programuotojo. Sistemos testai ir jų korektiškumas bus sprendžiamas įvertinant jų atitikimą nurodytiems panaudojimo atvejams.

Sistemos testavimas pradedamas nuo vienetų testų, kurie realizuojami kiekvieną kartą įgyvendinus naują arba pakeitus jau esančią sistemos dalį. Atsiradus pakankamam įgyvendintų sistemos dalių kiekiui, lygiagrečiai su vienetų testais bus pradedamas žaidžiamumo testavimas, šitaip užtikrinant, kad naujos žaidimo dalys veikia gerai ne tik iš techninės pusės, bet ir dera iš žaidėjo perspektyvos. Kartu su žaidžiamumo testavimu pradedami įgyvendinti našumo ir (jeigu įgyvendintas naujas UI funkcionalumas) naudotojo sąsajos testai, šitaip užtikrinant kad žaidimo našumas bus numatytoje ribose, o naudotojo grafinė sąsaja bus korektiška. Įgyvendinus kelias didesnes žaidimo sistemos dalių grupes, pradedamas regresinis testavimas žaidimo posistemių sąveikos korektiškumui ir greitaveikai užtikrinti. Turint veikiantį žaidimo prototipą ir vėliau įgyvendinus papildomas žaidimo posistemas, vykdomas Alfa testavimas, kurio metu randamos ir sutaisomos didžiausios problemos ir defektai. Alfa testavimas gali būti kartojamas kelis kartus. Atlikus Alfa testavimą ir turint didžiąją dalimi panaudojimo atvejus ir techninius reikalavimus atitinkantį žaidimą, vykdomas Beta testavimas. Šio testavimo metu įvertinama ar Alfa testavimo metu atrastos problemos buvo tikrai ištaisytos ir toliau ieškoma problemų ir defektų, kurie yra taisomi iškart, užtikrinant žaidimo žaidžiamumą, stabilumą ir švarumą. Beta testavimas kaip ir alfa testavimas, gali būti vykdomas kelis kartus atradus kritines problemas ar defektus, reikalaujančius esminių sistemos pakeitimų.

4. Testavimo tikslai

Pagrindinis testavimo tikslas – užtikrinti sklandžiai veikiančio ir reikalavimus atitinkančio strateginio žaidimo sistemos sukūrimą.

5. Rankinis testavimas (*angl. Manual testing*)

Rankinis testavimas bus vykdomas siekiant užtikrinti, kad žaidimas veikia pagal numatytus reikalavimus iš neautomatizuojamos perspektyvos.

Rankinis testavimas susideda iš dviejų procesų:

- Žaidžiamumo testavimas:
 - Tikrinama, ar visi žaidimo komponentai (padalinių gaminimas ir valdymas, pastatų statymas, resursų rinkimas ir valdymas) veikia pagal specifikacijas.
 - Tikrinama, ar žaidimo meniu ir nustatymai yra intuityvūs ir lengvai naudojami.
 - Patikrinama, ar žaidimo logika yra teisinga ir ar žaidėjas gali pasiekti pergalę arba pralaimėjimą pagal numatytus scenarijus.
- Našumo testavimas:
 - Patikrinama, ar žaidimas nesugenda arba nesustoja atliekant įprastus veiksmus.
 - Patikrinama, ar žaidimas veikia sklandžiai ir nėra našumo problemų, tokių kaip lėtėjimas ar vėlavimai.

6. Automatizuotas testavimas (*angl. Automated testing*)

Automatinis testavimas bus naudojamas siekiant efektyviai tikrinti žaidimo posistemes ir jų tarpusavio sąveiką, sumažinti žmogiškųjų klaidų tikimybę ir pagreitinti testavimo procesą.

Automatinis testavimas apima šiuos metodus:

- Vienetų testai:
 - Testuojami atskiri žaidimo komponentai, tokie kaip metodai ir funkcijos. Užtikrinama, kad kiekvienas žaidimo komponentas veiktų tinkamai ir grąžintų tikėtus rezultatus.
- Regresiniai testai:
 - Testuojama ar nauji pakeitimai žaidime nepažeidžia anksčiau veikusio funkcionalumo. Užtikrinama, kad po kiekvieno kodo pakeitimo žaidimas vis dar veikia pagal numatytus reikalavimus.
- Našumo testai:
 - Testuojamas žaidimo našumas, siekiant užtikrinti, kad jis veikia sklandžiai įvairiomis sąlygomis.
 - Testuojama, kaip žaidimas reaguoja į skirtingas apkrovas.
- Naudotojo sąsajos testavimas:
 - Testuojamas naudotojo sąsajos elementų funkcionalumas - užtikrinama, kad mygtukai, meniu ir kiti interaktyvūs elementai veikia taip, kaip tikėtasi.

7. Alfa ir Beta testavimas

Alfa testavimas ir beta testavimas yra svarbios žaidimo kūrimo proceso dalys, kurios padeda užtikrinti žaidimo kokybę ir funkcionalumą. Šios testavimo fazės leidžia atrasti ir ištaisyti problemas bei optimizuoti žaidimo našumą prieš galutinį išleidimą. Alfa testavimas gali vykti keliais etapais, kol didžioji dalis problemų yra išspręsta ir žaidimas yra pakankamai stabilus tolesniam testavimui – Beta testavimui. Beta testavimas taip pat gali vykti keliais etapais, kad būtų užtikrinta, jog žaidimas yra kuo labiau išbaigtas ir paruoštas galutiniam išleidimui. Tiek Alfa, tiek Beta testavimas yra būtini žingsniai, kurie padeda užtikrinti, kad strateginio žaidimo sistema būtų stabili ir aukštos kokybės.

7.1. Alfa testavimas

Alfa testavimas vykdomas ankstyvame žaidimo kūrimo etape, kai didžioji dalis pagrindinio funkcionalumo jau įgyvendinta, tačiau yra neišbaigtų ar neoptimizuotų dalių.

Pagrindiniai Alfa testavimo tikslai:

- Defektų atradimas ir taisymas: Išsamiai tikrinama visa žaidimo sistema, siekiant atrasti klaidas, defektus ir kitus trūkumus. Randamos problemos registruojamos ir taisomos kuo greičiau, kad būtų užtikrintas stabilus žaidimo veikimas.
- Funkcionalumo tikrinimas: Tikrinama, ar visos pagrindinės žaidimo funkcijos veikia pagal numatytus reikalavimus. Vertinama, ar žaidimo mechanikos logiškai siejasi ir dera tarpusavyje.
- Našumo optimizavimas: Tikrinama, ar žaidimas veikia sklandžiai ir ar nėra našumo problemų, tokių kaip trikdžiai ar vėlavimai (*angl. Lag*).
- Atliekami optimizavimo darbai, siekiant pagerinti žaidimo greitaveiką.

7.2 Beta testavimas

Beta testavimas vyksta po Alfa testavimo, kai žaidimas jau yra pakankamai stabilus ir didžioji dalis funkcionalumo yra įgyvendinta.

Pagrindiniai Beta testavimo tikslai:

- Likusių defektų atradimas: Tikrinama, ar visos alfa testavimo metu atrastos problemos buvo tinkamai išspręstos. Randamos ir taisomos naujai atrastos problemos, kurios gali turėti įtakos žaidimo kokybei ir stabilumui.
- Žaidimo balanso tikrinimas: Tikrinama, ar žaidimas yra tinkamai subalansuotas - įvertinama, ar žaidimo mechanikos ir taisyklės yra teisingos ir sąžiningos visiems žaidėjams.
- Paskutiniai našumo testai: Atliekami paskutiniai našumo testai, siekiant užtikrinti, kad žaidimas veikia sklandžiai įvairiomis sąlygomis ir prie įvairių apkrovų.

8. Testavimo elementai

1 Lentelė. Testavimo elementų testavimo būdai, panaudoti testai ir prioritetai.

Sistemos elementas	Testavimo būdai	Panaudoti testai	Prioritetas
Padalinių gaminimo ir valdymo posistemė	Rankinis ir automatizuotas	Vienetų testai, Regresiniai testai, Našumo testai, Žaidžiamumo testavimas	Aukštas
Pastatų statymo ir valdymo posistemė		Vienetų testai, Regresiniai testai, Našumo testai, Žaidžiamumo testavimas	Aukštas
Resursų valdymo posistemė		Vienetų testai, Regresiniai testai, Našumo testai, Žaidžiamumo testavimas	Aukštas
Padalinių, pastatų ir resursų posistemių sąveika ir greitaveika		Vienetų testai, Regresiniai testai, Našumo testai, Žaidžiamumo testavimas	Vidutinis
Grafinė sąsaja		Žaidžiamumo testavimas, Naudotojo sąsajos testavimas, Našumo testavimas	Žemas

10. Naudotojo sąsajos testavimas

Naudotojo sąsajos testavimo metu tikrinama, ar visos grafinės sąsajos dalys veikia taip, kaip numatyta reikalavimuose. Šio testavimo metu tikrinami meniu ir jų mygtukai, taip pat kitokių tipų interaktyvūs grafinės sąsajos elementai. Žemiau pateiktoje lentelėje apibrėžiami naudotojo sąsajos testavimo scenarijai. Svarbu pastebėti, kad pagrindinio meniu išdėstymas ir veikimas yra identiškas žaidimo meniu, todėl testavimo scenarijų sąrašas yra identiškas.

2 Lentelė. Grafinės naudotojo sąsajos testavimo scenarijų sąrašas.

Testo nr.	Meniu	Grafinės naudotojos sąsajos elementas	Veiksmas	Tikėtinas rezultatas
1	Pagrindinis meniu	Mygtukas „Quit“	Paspaudžiamas mygtukas	Uždaromas žaidimas
2	Pagrindinis meniu	Mygtukas „Play“	Paspaudžiamas mygtukas	Pradedamas žaidimas
3	Pagrindinis meniu	Mygtukas „Settings“	Paspaudžiamas mygtukas	Atidaromas nustatymų meniu
4	Nustatymų meniu	Mygtukas „Back“	Paspaudžiamas mygtukas	Atidaromas pagrindinis meniu
5	Nustatymų meniu	Pažymėjimo elementas „Fullscreen“	Pažymimas elementas	Ekrano režimas pasikeičia iš pilno ekrano (<i>angl. Fullscreen</i>) į sumažinto ekrano (<i>angl. Windowed</i>) arba iš sumažinto ekrano į pilno ekrano režimą
6	Nustatymų meniu	Elementas su grafikos pasirinkimų sąrašu – „Low“, „Medium“, „High“, „Very High“	Pasirenkamas elementas iš sąrašo	Žaidimo grafikos nustatymai pakeičiami į atitinkamus priklausomai nuo pasirinkto varianto
7	Nustatymų meniu	Elementas su rezoliucijos pasirinkimų sąrašu	Pasirenkamas elementas iš sąrašo	Žaidimo rezoliucija pakeičiama į atitinkamą priklausomai nuo pasirinkto varianto
8	Nustatymų meniu	Slankiojantis elementas su garso valdymu „Volume“	Paslenkamas rutulio formos elementas į kairę arba dešinę	Pakeičiamas žaidimo garsas priklausomai nuo to, į kurią pusę buvo paslinktas rutulio formos elementas.

11. Testavimo valdymas

Testavimo valdymas strateginio žaidimo sistemai apima 6 pagrindinius žingsnius, kurie užtikrins sklandų ir efektyvų šios sistemos testavimo procesą. Kiekvienas žingsnis yra svarbus siekiant aptikti ir ištaisyti klaidas bei optimizuoti žaidimo veikimą.

- Testavimo planavimas: Nustatomi testavimo tikslai, apimtis ir metodai. Sukuriamas detalus testavimo grafikas, apibrėžiant testavimo etapus ir jų trukmę.
- Testavimo aplinkos paruošimas: Sukuriama ir konfigūruojama testavimo aplinka ir užtikrinama, kad visi reikalingi įrankiai būtų paruošti.
- Testų kūrimas ir dokumentavimas: Sukuriami testų scenarijai ir testavimo atvejai, atitinkantys nurodytus reikalavimus. Visi testai dokumentuojami, nurodant tikslus, metodus ir tikėtinus rezultatus.
- Testų vykdymas: Vykdomi suplanuoti testai pagal nustatytą grafiką. Fiksuojami visi testavimo rezultatai ir aptiktos klaidos.
- Defektų valdymas: Aptiktos klaidos registruojamos ir klasifikuojamos pagal prioritetą, pagal kurį po to yra taisomos.
- Rezultatų analizė: Atsižvelgiant į testų rezultatus nusprendžiamas sistemos reikalavimų ir panaudos atvejų išpildymas, ir aptiktas klaidas.

12. Testavimo grafikas

Žemiau pateiktas testavimo grafikas lentelės pavidalu (**3 lentelė**). Laikotarpiai Alfa ir Beta testavimui gali keistis pagal situaciją.

3 Lentelė. Testavimo grafikas

Testavimo metodas	Data nuo	Data iki
Vienetų testai	2024-05-01	2025-02-01
Žaidžiamumo testavimas	2024-06-01	2025-02-01
Regresinis testavimas	2024-07-01	2025-02-01
Našumo testavimas	2024-07-01	2025-02-01
Naudotojo sąsajos testavimas	2024-08-01	2025-02-01
Alfa testavimas	2025-02-01	2025-03-15
Beta testavimas	2025-03-15	2025-05-01

13. Patvirtinimas

4 Lentelė. Testavimo plano patvirtinimas

Vardas ir pavardė	Pareigos	Patvirtinimas
Šarūnas Packevičius	Vadovas	
Tadas Laurinaitis	Architektas, programuotojas ir testuotojas	

14. Testavimo rizika

Galimos rizikos:

- Testų įgyvendinimas, ypatingai komandoje sudarytoje iš vieno programuotojo, kuris tuo pačiu yra ir testuotojas, užtrunka daug laiko, kuris galėtų būti skirtas papildomam strateginio žaidimo funkcionalumui kurti.
- Alfa ir Beta testavimo metu gali likti nepastebėtų problemų, kurios vėliau įtakos naudotojų pasitenkinimą žaidžiant žaidimą.