

LATVIJAS UNIVERSITĀTE  
DATORIKAS FAKULTĀTE

**REN'PY DZINĒJA SPĒLES KODA ANALĪZE UN  
LABOŠANA**  
KVALIFIKĀCIJAS DARBS

Autors: Agnese Griķe

Studenta apliecības Nr.: ag21201

Darba vadītājs: Mg. sc. comp. Ilvars Mizniks

RĪGA 2024

# ANOTĀCIJA

Kvalifikācijas darba “Ren’Py dzinēja spēles koda analīze un labošana” mērķis ir izpētīt Ren’Py dzinēja spēles “*Call of the Void*” pirmās sērijas kodu, kā arī veikt kļūdu labojumus un citus spēles lietotājam noderīgus uzlabojumus.

Darba ietvaros tiks izpētīta spēles koda struktūra un moduļi, izveidota koda dokumentācija un funkciju projektējums ar padziļinātiem funkcionalitātes aprakstiem, un veikti un testēti spēles labojumi.

Atslēgvārdi: Ren’Py, Python, spēle, analīze, izmaiņas

# ABSTRACT

## Ren'Py engine game code analysis and alteration

The purpose of the qualification work “Ren'Py engine game code analysis and alteration” is to investigate the code of the Ren'Py engine game’s “*Call of the Void*” first episode, as well as make bug fixes and other user-friendly improvements.

Within this work the author will investigate the game’s code structure and modules, create documentation and function design with in-depth description of functionality, and implement and test game improvements.

Key words: Ren'Py, Python, game, analysis, modifications

# SATURS

1.	IEVADS .....	9
1.1.	Nolūks.....	10
1.2.	Darbības sfēra .....	10
1.3.	Saistība ar citiem dokumentiem .....	10
1.4.	Pārskats.....	11
2.	VISPĀRĒJAIS APRAKSTS .....	12
2.1.	Esošā stāvokļa apraksts .....	12
2.2.	Pasūtītājs.....	13
2.3.	Perspektīva .....	13
2.4.	Darījumasprasības.....	14
2.5.	Sistēmas lietotāji.....	14
2.6.	Vispārējie ierobežojumi.....	15
2.7.	Pieņēmumi un atkarības .....	15
3.	PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA .....	16
3.1.	Funkcionālās prasības.....	16
3.1.1.	Spēles vispārējā gaita.....	16
3.1.2.	Spēles funkcijas .....	17
3.1.3.	Funkcionālie moduļi.....	18
3.1.4.	Stāstu moduļi .....	20
3.1.5.	Paziņojumi .....	21
3.1.6.	Pārbaudes.....	23
3.1.7.	Funkciju sadalījums pa moduļiem.....	24
3.1.8.	Sasniegumu modulis.....	25
3.1.9.	Cīņu modulis .....	28
3.1.10.	Smilšukastes modulis .....	35
3.1.11.	Skatu modulis .....	41
3.1.12.	Skripta modulis.....	44
3.1.13.	Teksta modulis.....	46
3.1.14.	Galvenais modulis .....	50
3.1.15.	Centra modulis.....	54
3.1.16.	Priekšpilsētas modulis .....	60

3.1.17. Ārpilsētas modulis .....	63
3.2. Nefunkcionālās prasības .....	67
3.2.1. Veiktspējas prasības .....	67
3.2.2. Pieejamības prasības .....	67
4. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS – KODA IZPĒTE .....	68
4.1. Lokālās datu krātuves struktūras projektējums .....	68
4.1.1. Funkcionālie mainīgie .....	69
4.1.2. Attiecību mainīgie .....	70
4.1.3. Vietu atrašanas karogi .....	71
4.1.4. Uzdevumu mainīgie.....	71
4.1.5. Inventāra mainīgie .....	74
4.1.6. Tēlu mainīgie.....	75
4.1.7. Karodziņi .....	77
4.2. Funkciju projektējums un apraksts .....	79
4.2.1. Sasniegumu modulis.....	79
4.2.2. Cīņas modulis .....	81
4.2.3. Smilšukastes modulis .....	89
4.2.4. Skatu modulis .....	95
4.2.5. Skripta modulis.....	97
4.2.6. Teksta modulis.....	98
4.2.7. Galvenais modulis .....	102
4.2.8. Centra modulis.....	107
4.2.9. Priekšpilsētas modulis .....	114
4.2.10. Ārpilsētas modulis .....	118
5. PROGRAMMATŪRAS KODĀ VEIKTĀS IZMAIŅAS.....	121
5.1. Izmaiņu saraksts .....	121
5.2. Sasniegumu modulis (ach.rpy) .....	122
5.2.1. Jauna sasnieguma apraksts un lodziņš.....	122
5.3. Cīņas modulis (combat.rpy) .....	123
5.3.1. Indes mainīgo implementēšana COM_001 .....	123
5.3.2. Mainītas formulas COM_005.....	123
5.3.3. Mainītas formulas COM_006.....	123
5.3.4. Indes efektu izraisīšana COM_006 .....	123
5.3.5. Pievienots glābšanas uzbrukuma izsaukums un indes efekti COM_003 .....	125

5.3.6. COM_008 – glābšanas uzbrukums.....	127
5.4. Smilšukastes modulis (sandbox.rpy) .....	131
5.4.1. Izmaiņas SBX_001 .....	131
5.4.2. Jaunas inventāra lietas SBX_005 .....	132
5.5. Skatu modulis (screens.rpy) .....	133
5.5.1. Izmaiņas VIS_004 skatā.....	133
5.6. Skripta modulis (script.rpy).....	134
5.6.1. Izmaiņas SCR_001 .....	134
5.7. Teksta modulis.....	135
5.7.1. Monstru sasnieguma skaitīšana TXT_005 .....	135
5.8. Galvenais modulis (loc.rpy) .....	136
5.8.1. Sasnieguma iegūšana LOCM_001 .....	136
5.8.2. Mrisjas patikšana LOCM_005 .....	138
5.8.3. Uzdevuma quest20 pabeigšanas implementēšana .....	139
5.9. Centra modulis (loc_center.rpy).....	140
5.9.1. Zobens uzlabojumi LOCC_002.....	140
5.9.2. Jauns uzdevums LOCC_001 .....	140
5.9.3. Kļūdu labojumi centra modulī.....	142
5.10. Priekšpilsētas modulis (loc_okr.rpy) .....	142
5.10.1. Uzdevuma quest14 implementācija – LOCK_001 .....	142
5.10.2. Indes nopirkšana no Ilgas – LOCK_002 .....	144
5.10.3. AINU iespējamību izmaiņas – LOCK_003.....	146
5.11. Ārpilsētas modulis (loc_outside.rpy).....	148
5.11.1. Fūrijas uzbrukums – quest11 .....	148
5.12. Lokālā datu krātuve – vari.rpy.....	148
5.12.1. Mainīgo pievienošana.....	148
5.12.2. Indes lietošanas funkcija.....	148
6. TESTĒŠANAS DOKUMENTĀCIJA.....	149
6.1. Testēšanas apraksts.....	149
6.2. Testēšanas žurnāls .....	149
7. PROJEKTA PĀRVALDĪBA .....	155
7.1. Projekta organizācija .....	155
7.2. Kvalitātes nodrošināšana .....	155
7.3. Konfigurāciju nodrošināšana.....	155

7.4.	Darbietilpības novērtējums.....	156
7.4.1.	Darbietilpības novērtējums dienu skaitā .....	156
7.4.2.	Darbietilpības novērtējums pēc koda rindiņu skaita.....	157
8.	SECINĀJUMI .....	160
9.	IZMANTOTĀ LITERATŪRA UN AVOTI .....	161
10.	PIELIKUMI.....	163

# VĀRDNĪCA

Ren'Py – spēļu dzinējs, kas ir balstīts uz Python; plaši pielietots vizuālā stāsta žanra spēlēm.

Python – augsta līmeņa programmēšanas valoda – objektorientēta, interpretējama. Plaši pielietota daudzās IT nozarēs.

.rpy – Paplašinājums Ren'Py dzinēja spēļu koda failiem.

# 1. IEVADS

Šī kvalifikācijas darba mērķis ir analizēt un veikt labojumus Ren'Py dzinēja spēles kodā.

Lai būtu vairāk funkcionāla koda, ko analizēt, darbam tika izvēlēta spēle, kas satur funkcijas, kas ir reti sastopamas citās žanra spēlēs – šajā spēlē ir naudas sistēma, cīņu sistēma, un inventāra sistēma, kas palielina spēles funkcionālo sarežģītību.

Kāpēc analizēt kodu? Pirmkārt, koda analīze ir svarīga un visaptveroša daļa no jebkura kvalificēta programmētāja darba. Tā kā šajā sfērā nav iespējams zināt it visu, ir svarīgi iepazīties ar dažādiem algoritmiem, projektējumiem, metodēm un pieejām, lai paplašinātu savas zināšanas. Otrkārt, ir vairākkārt pierādīts, ka analizēt cita rakstītu kodu ir sarežģītāk, nekā rakstīt savu.

Tā kā bieži vien programmētāja darbs sastāv vairāk no koda uzturēšanas, nevis jauna koda veidošanas,<sup>[1][2]</sup> šis tiek uzskatīts par diezgan plaši pieņemtu faktu programminženierijas jomā, bet viedokļi atšķiras, cik tieši sarežģītāks šis uzdevums ir. Pētījumi liecina, ka vidēji programmētājs pavada stipri vairāk laika uzturot un analizējot cita kodu, nekā rakstot savu – skaitļi atšķiras no trīsreiz vairāk līdz pat 10 reizes vairāk.<sup>[3]</sup> Ar citu pētījumu veikšanu un formulu izveidošanu, tika secināts, ka šī metrika ir robežās no 7 līdz pat 200 reizes vairāk.<sup>[4]</sup> Protams, tas ir atkarīgs no programmatūras sarežģītības – kods ar komplicētiem, cits citu izsaucošiem algoritmiem un sarežģītu sistēmas struktūru ir daudz grūtāk izprotams projekta jaunpienācējam nekā vienkāršākas sistēmas kods. Darba ietvaros tiks pieņemta metrika 3:1 – tas ir, trīs sevis rakstītas rindiņas atbilst vienai izlasītai rindiņai – tālākai informācijai un analīzei skatīt dokumenta sadaļu par darbietilpības novērtējumu.

Koda komentēšana ir laba programmēšanas stila pamats, bet ne visi pie tā pieturas. Ja ir nepieciešams strādāt ar šāda veida kodu, piemēram, darbā, kur programmētāja uzdevums ir uzturēt cita rakstītu kodu, ir vajadzīgs iemācīties, kā izprast šo kodu bez pieejamiem komentāriem un dokumentācijas – un reversi konstruēt precīzu dokumentāciju, lai atvieglotu darbu gan sev, gan kolēģiem.

## 1.1. Nolūks

Šī dokumenta nolūks ir izpētīt un aprakstīt esošu publicētas sistēmas – Ren’Py spēles – kodu un tajā veiktās izmaiņas, un izveidot sistēmas dokumentāciju. Dokuments ir paredzēts sistēmas lietotājiem, kuriem būtu nepieciešama dokumentācija, lai veiktu spēles uzturēšanu un tālākas izmaiņas, ko šajā sfērā sauc par spēles modificēšanu.

## 1.2. Darbības sfēra

Kvalifikācijas darbā ietvertajai koda izpētei un mainīšanai izvēlēta spēle ir “*Call of the Void*” jeb “*Зов Пустоты*” oriģinālajā krievu tulkojumā, kura darba rakstīšanas laikā sastāv no divām sērijām un kuras autori ir komanda “*Novel*”.<sup>[5]</sup> Spēles pirmā sērija tika publicēta 2019.gadā, un otrā sērija 2022. gadā. Pašlaik top arī trešā un pēdējā sērija, kas tiks publicēta 2024.gadā. Darba ietvaros tiks apskatīta tikai pirmā sērija.

Spēle pieder vizuālā stāsta žanram – šo žanru definē spēles gaita, kas lielākoties sastāv no teksta un attēliem, un tipiski satur maz citu funkciju. Balstoties uz šo faktu, šo spēļu gaita ir lielākoties neatkarīga no lietotāja ievades, izņemot noteiktus punktus stāsta laikā, kur izvēle var mainīt stāsta gaitu.

Sistēma ir izveidota Ren’Py valodā. Ren’Py ir spēļu dzinējs, kas ir balstīts Python valodā, kas ir izveidots tieši vizuālajiem stāstiem. Šajā dzinējā rakstīta spēle tika izvēlēta, jo lielākā daļa no šīm spēlēm ir brīvprogrammatūra, un to kods ir viegli pieejams un maināms, izmantojot par brīvu pieejamus rīkus.

## 1.3. Saistība ar citiem dokumentiem

Dokumenta izveidē tika ievēroti standarti, kas aprakstīti dokumentos LV 68:1996 “Programmatūras prasību specifikācijas ceļvedis”<sup>[6]</sup> un LVS 72:1996 “Ieteicamā prakse programmatūras projektējuma aprakstīšanai”.<sup>[7]</sup>

## 1.4. Pārskats

Dokuments ir sadalīts desmit daļās:

1. Ievads – satur informāciju par darba motivāciju, dokumenta nolūku, darbības sfēru un saistību ar citiem dokumentiem.
2. Vispārējais apraksts – satur informāciju par esošā stāvokļa aprakstu, pasūtītāju, darba perspektīvu, darījumprasībām, sistēmas lietotājiem, vispārējiem ierobežojumiem un pieņēmumiem un atkarībām.
3. Programmatūras prasību specifikācija (PPS) – satur funkcionālo un nefunkcionālo prasību aprakstus.
4. Programmatūras projektējuma apraksts (PPA) – satur PPS nodaļā aprakstīto funkcionālo prasību un lokālās datu krātuves projektējumu.
5. Programmatūras kodā veiktās izmaiņas – apraksta visas programmatūras kodā veiktās izmaiņas, balstoties uz PPS un PPA aprakstītajām funkcionālajām prasībām.
6. Testēšanas dokumentācija – apraksta testēšanas procesu un satur testēšanas žurnālu.
7. Projekta pārvaldība – satur informāciju par projekta organizāciju, kvalitātes nodrošināšanu, konfigurāciju pārvaldību un darbietilpības novērtējumu.
8. Secinājumi par darbu.
9. Izmantotie informācijas avoti.
10. Pielikumi – satur koda fragmentus un vienu ekrānuzņēmumu no spēles.

## 2. VISPĀRĒJAIS APRAKSTS

### 2.1. Esošā stāvokļa apraksts

Saistībā ar darba tēmu var aprakstīt divus esošos stāvokļus.

Viens ir kvalitatīvas dokumentācijas esamība. Pieredze rāda, ka lielos, nostiprinātos uzņēmumos tiek stabili ievērota laba programmizstrādes prakse – detalizēta dokumentācija, komentēts kods utt. –, bet daudzos jaunuzņēmumos, kuros tipiski trūkst programmētāju un nav pietiekami liela komanda, lai efektīvi sadalītu uzdevumu pa darbiniekiem, šī prakse bieži ir pirmais upuris produktu izstrādes procesā. Tipiski, jo mazāka komanda un plašāki viņu pienākumi, jo mazāka iespēja, ka programmatūra būs precīzi dokumentēta. Tomēr komandas jaunpienācējiem joprojām ir jāspēj izprast sarakstīto kodu, lai to uzturētu vai papildinātu. Citu koda saprašana ir viena no svarīgākajām un nepieciešamajām programmētāja spējām.

Otrs ir spēļu koda modificēšana kā hobijs cilvēkiem, kas interesējas par datorspēlēm. Šī darba ietvaros tiks veikts process, ko sfērā sauc par “game modding” jeb spēļu modificēšanu, kas iekļauj netriviālu funkciju mainīšanu, lai uzlabotu vai citādāk mainītu pieredzi spēles lietotājiem. Šīs modifikācijas, jeb sarunvalodā sauktas par “modiem” (no angļu val. *modification*, saīsināts – *mod*) ir tipiski patstāvīgi vai mazās komandās izstrādāti un pieejami kā brīvprogrammatūra. Šis process ir legāls, ja tas nesniedz modifikāciju autoram finansiālu ieguvumu vai nepārkāpj autortiesības.<sup>[8]</sup> Pirmais punkts ir pašsaprotams, bet otrais ir atkarīgs no spēles publicētājiem. Spēles kā “Minecraft” ir slavenas ar saviem plaši zināmajiem un izplatītajiem modifikāciju forumiem, un šajā gadījumā spēles izstrādātāji “Mojang” iedrošina spēles modificēšanu – atbilstoši izveidojot funkcijas spēles oficiālajā versijā, kas pārvalda modifikācijas,<sup>[9]</sup> un ir pat pieņēmuši šo modifikāciju autorus darbā uzņēmumā.<sup>[10]</sup> Citi izstrādātāji, piemēram, “Nintendo”, neatļauj savu spēļu modifikāciju caur saviem stingri definētajiem galalietotāja licences līgumiem un pat iesaistīšanos tiesā.<sup>[11]</sup>

Ren’Py dzinēja spēles ir pieejamas kā atvērtā koda sistēmas – tas nozīmē, ka šīm spēlēm pēc definīcijas nav ierobežotu autortiesību licences. Darbā apskatītajā spēlē “Call of the Void” aprakstā ir šāds teksts “This program contains free software under a number of licences, including the MIT License and GNU Lesser General Public License.” (tulkojumā – “Šī

programma satur brīvprogrammatūru ar noteiktām licencēm, ieskaitot MIT License and GNU Lesser General Public License.”), skat. pielikumu 1.

Ir pieminētas MIT un GNU licences. MIT licences nosacījumi saka, ka “katram cilvēkam, kas ir ieguvis programmatūras kopiju un ar to saistītos failus, ir tiesības lietot, kopēt, rediģēt, apvienot, izplatīt, publicēt, apakšlicencēt un pārdot programmatūras kopijas. Vienīgais nosacījums programmatūras lietošanai ir autortiesību informāciju iekļaušana visās kopijās un būtiskās programmatūras daļās.” (tulkots no angļu val.)<sup>[12]</sup> GNU licences nosacījumi saka, ka “[licence] dod atvērtā koda atļaujas lietotājiem, ieskaitot programmatūras brīvu lejupielādi un palaišanu, tiesības veikt izmaiņas programmatūrā pēc vēlmes, tiesības izplatīt programmatūras kopijas, un tiesības modificēt un izplatīt programmatūras jauno versiju kopijas.” (tulkots no angļu val.)<sup>[13]</sup> Tātad, kamēr oriģinālajam kodam tiek norādītas tā autortiesības, koda modificēšana ir legāla šajā gadījumā.

## **2.2. Pasūtītājs**

Darbs tiek veidots pēc autora un Latvijas Universitātes Datorikas fakultātes pasniedzēja Ilvara Miznika iniciatīvas studiju programmas “Programmēšana un datortīklu administrēšana” kvalifikācijas darba ietvaros, bez konkrēta sistēmas pasūtītāja.

## **2.3. Perspektīva**

Darba plānotais rezultāts ir eksistējošas sistēmas – Ren’Py dzinēja spēles “*Call of the Void*” – un tajā veikto izmaiņu dokumentācija.

Darba ietvaros izveidotā dokumentācija var potenciāli tikt izmantota tālākai izmaiņu un koda uzturēšanas veikšanai.

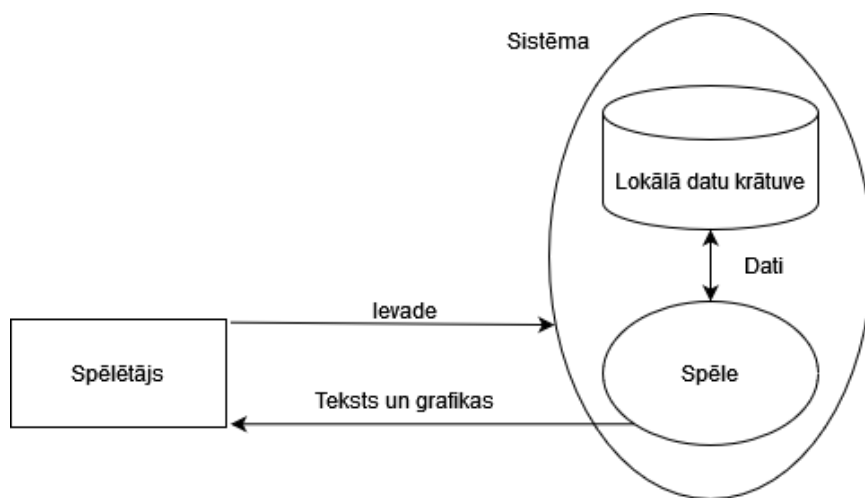
## 2.4. Darījumasprāsības

Sistēmas galvenās funkcijas ir:

- Sarunas un sadarbšanās ar pilsētas iedzīvotājiem
- Cīņas sistēma
- Uzdevumu sistēma (galvenie un neobligāti)
- Tirgošanās sistēma
- Attiecību sistēma

## 2.5. Sistēmas lietotāji

Sistēmas, un no tā sekojot, šīs dokumentācijas lietotāji var tikt dalīti divās teorētiskās lomās – parasti spēlētāji un spēles uzturētāji –, bet starp šīm abām lomām nav atšķirību piekļuves tiesībās, tikai izmantošanas nolūkā.



Attēls 2.1 - 0.līmeņa datu plūsmas diagramma

## 2.6. Vispārējie ierobežojumi

Sistēma tiek modificēta publicētajā Windows operētājsistēmas saderīgajā versijā.

No lietotāja puses nav specifisku ierobežojumu – sistēma ir lietojama bezsaistē, un tās lietošanai nav nepieciešami papildus instalējami rīki.<sup>[14]</sup>

Lai spētu apskatīt un rediģēt sistēmas kodu, ir nepieciešams rīks UnRen;<sup>[17]</sup> vairāk par to aprakstīts programmatūras kodā veikto izmaiņu nodaļā.

## 2.7. Pieņēmumi un atkarības

Sistēmas lietotājam ir nepieciešams dators ar grafisku saskarni, datorpeli vai analogisku ievades ierīci, un modifikācijas nolūkiem nepieciešama teksta ievade (klaviatūra) un koda redaktors ar Ren'Py valodas atbalstu. Autors darba ietvaros izmanto Visual Studio Code – brīvprogrammatūra pieejama no Microsoft – ar iespējotu Ren'Py paplašinājumu, kuras autors ir *LuqueDaniel*.<sup>[15]</sup>

## 3. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

### 3.1. Funkcionālās prasības

#### 3.1.1. Spēles vispārējā gaita

Pēc spēles atvēršanas lietotājam tiek parādīts sākotnējais skats, kur tiek jautāts, vai lietotājs ir pilngadīgs. Ja atbilde ir apstiprinoša, tiek parādīts sākuma skats – no šī skata var sākt jaunu spēli, turpināt esošu saglabātu spēli, atvērt iestatījumus, atvērt atmiņu skatu, atvērt palīdzības skatu, atvērt ārējās saites uz spēles veidotāju Steam lapu vai Discord serveri, vai no spēles iziet.

Pēc jaunas spēles sākšanas vai esošas spēles turpināšanas tiek ielādēti notikumi jeb spēles teksta funkcijas – vairāk par tām paskaidrots zemāk. Spēles gaita norisinās ar peles klikšķiem, kas tin tekstu un attēlu slaidrādi uz priekšu. Ik pa laikam spēlētājam tiek dota iespēja izvēlēties starp divām vai vairāk dialoga iespējām – šīs izvēles parasti ietekmē citas spēles funkcijas, piemēram, palielina reputāciju, galvenā tēla pievilcību citiem tēliem vai vienkārši iespējo spēles tālāku gaitu.

Notikumi tiek iedarbināti, izvēloties noteiktas vietas vispasaules skatos un veicot noteiktas izvēles teksta izvēlnē – piemēram, ieejot krogā, pie bārmeņa var pasūtīt ēdienu, vai arī ik pa laikam, balstoties uz citiem notikumiem spēlē, prasīt jautājumu, kas iedarbina notikumu, kas atbild uz šo jautājumu un veicina uzdevumu gaitu.

Spēlē ir uzdevumi (sfērā tie tiek saukti par kvestiem – no angļu val. *quest*), kas veicina stāsta gaitu. Lai pabeigtu uzdevumu, ir jāiedarbina dažāds skaits notikumu noteiktā secībā. Uzdevuma pabeigšana dod papildus pieredzes punktus; pēc noteikta skaita punktu savākšanas galvenais tēls iegūst jaunu līmeni, un ir dota iespēja uzlabot kādu no tēla atribūtiem. Spēlē ir galvenie uzdevumi un neobligātie uzdevumi – kaut gan ne visi galvenie uzdevumi ir absolūti obligāti stāsta iziešanai, to neizpildīšana var ietekmēt spēles turpmāko gaitu.

Vispasaules skatā ir arī iespēja izvēlēties vietas ārpus stāsta galvenās pilsētas, kur ir iespējamas medības – tas ir, sākt cīņas ar citām būtnēm. No šīm cīņām var iegūt papildus naudu un pieredzes punktus. Naudu var tērēt pilsētā pie vairākiem tirgotājiem – pie bruņotāja, kas pārdod bruņas un citus rīkus, kas uzlabo tēla cīņas spējas; pie alķīmiķa, kurš pārdod dziras, kas

ļauj atgūt zaudētus dzīvības punktus un dzēst nelabvēlīgus efektus; un saunā, kur tēls var nomazgāties – ja tēls nemazgājas, pilsētnieki atsakās mijiedarboties ar viņu.

Spēle tiek pabeigta, kad tiek iedarbināts noteikts notikums galvenajā, lineārajā stāstā – lietotājam tiek dots brīdinājums, ka spēlē vairs nevarēs atgriezties, un pēc notikuma beigām lietotājam tiek paziņots, ka sērija ir pabeigta un stāsta turpinājums ir nākamajā sērijā.

### 3.1.2. Spēles funkcijas

Moduļi spēles kodā ir sadalīti pa atsevišķiem failiem ar .rpy paplašinājumu, kas ir specifisks Ren'Py valodai.

Darba ietvaros spēles moduļi tiks sadalīti divās kategorijās – funkcionālie moduļi un stāsta moduļi.

Funkcionālie moduļi nodrošina sistēmas vispārējo darbību. Šie moduļi ir sekojoši: sasniegumu modulis (ach.rpy), cīņu modulis (combat.rpy), smilšukastes modulis (sandbox.rpy), skatu modulis (screens.rpy), skripta modulis (script.rpy), teksta modulis (apkopots no visām funkcijām, kas pa failiem ir funkcionāli identiskas).

Stāsta moduļi ir funkcionāli homoloģiski, atšķiroties tikai ar teksta saturu un apstrādātajām lokācijām spēles pasaulē. Tie ir: galvenais modulis (loc.rpy), centra modulis (loc\_center.rpy), priekšpilsētas modulis (loc\_okr.rpy – no krievu val. *окраина* jeb *nomale*), ārpuspilsētas modulis (loc\_outside.rpy).

Pēc spēles koda analīzes autors secina, ka funkciju sadalījums pa stāsta moduļiem ir diezgan patvaļīgs, un cita spēles autora rokās varēja būt citāds.

Šie moduļi, it īpaši stāsta moduļi, var sastāvēt no simtiem koda funkcijām – funkcija Ren'Py tiek definēta ar birku *label*, bet, tā kā lielākā daļa no šīm funkcijām ir homoloģiskas – piemēram, tikai rāda tekstu –, tās tiks apkopotas kā viena funkcija to identiskās darbības dēļ.

Stāstu moduļos daudz parādās *label* funkcijas, kas ir nodalītas citāda teksta satura dēļ (šī ir Ren'Py spēļu koda struktūras īpatnība – viss spēles teksts parādās kodā), bet ir funkcionāli identiskas. Šīs funkcijas tiks sadalītas sekojošās kategorijās un tālāk netiks atšķirtas, ja tas nav absolūti nepieciešams citas funkcijas aprakstīšanai. Iekavās tiek pievienoti nosaukumi, kā šīs funkcijas tiks darbā tālāk sauktas:

- Teksta funkcijas, kas tikai rāda tekstu – spēlētāja ievade netiek prasīta un mainīgie netiek rediģēti (Displeja funkcija)
- Teksta funkcijas, kurām nepieciešama vai nu spēlētāja ievade, vai ievade atbilstoši iepriekš definētiem mainīgajiem (Ievades funkcija)
- Teksta funkcijas, kas rediģē vienu vai vairākus spēles mainīgos (Funkcija ar rediģēšanu)
- Teksta funkcijas, kurām gan nepieciešama spēlētāja vai automātiska ievade, gan tiek rediģēti mainīgie (Ievades funkcija ar rediģēšanu)
- Teksta funkcijas, pēc kurām tiek izsaukta cīņa ar COM\_001 (Cīņas izsaukums)
- Teksta funkcijas, kas ir atkarīgas no nejauši iedalīta mainīgā (Funkcija ar nejaušu mainīgo)

Visās šajās teksta funkcijās funkcijas beigās tiek izsaukta vai nu cita teksta funkcija (manuāli ar izsaukumu, vai automātiski, balstoties uz kodu – Ren'Py ir secīgi kompilējoša un izsaucoša valoda), vai kāda no funkcionālo moduļu funkcijām, lielākoties SBX\_001 un SBX\_002, dažos gadījumos COM\_001, vai atgriežas funkcijā, kas doto izsauca. Šis funkciju grupējums tiks apkopots kā teksta modulis.

Tālāk sekojošajās tabulās ir aprakstītas moduļu funkcijas, balstoties uz to vispārējo funkcionalitāti un ar darba autora izveidotiem identifikatoriem.

### 3.1.3. Funkcionālie moduļi

Tabula 3.1 - funkcionālo moduļu specifkācija

Modulis	Funkcija	Identifikators
Sasniegumu modulis	Definēt sasniegumu nosaukumus un tekstu	ACH_001
	Iegūt sasniegumu	ACH_002

Modulis	Funkcija	Identifikators
	Parādīt sasniegumus ( <i>screen</i> birka)	ACH_003
Cīņu modulis	Uzsākt cīņu	COM_001
	Automātiski atrisināt cīņu	COM_002
	Uzbrukums (spēlētājs)	COM_003
	Uzbrukums (pretinieks)	COM_004
	Cīņas beigas	COM_005
	Uzbrukt pretiniekam	COM_006
	Atvairīt pretinieka uzbrukumu	COM_007
Smilšukastes modulis	Tīt laiku uz priekšu	SBX_001
	Iet uz noteiktu vietu virspasaulē	SBX_002
	Ārpilsētas izvēlne	SBX_003
	Parādīt profilu ( <i>screen</i> birka)	SBX_004
	Cīņas inventārs ( <i>screen</i> birka)	SBX_005
	Palielināt līmeni	SBX_006
	Spēles beigas	SBX_007
	Istabas izvēlne	SBX_008
	Mainīt uzdevuma vērtību	SBX_009
Skatu modulis	Parādīt izvēlnes	VIS_001
	Parādīt teksta vēsturi	VIS_002

Modulis	Funkcija	Identifikators
	Pārvaldīt saglabāto spēļu sarakstu	VIS_003
	Parādīt uzdevumu sarakstu	VIS_004
Skripta modulis	Jautāt par spēlētāja vecumu	SCR_001
	Uzsākt jaunu spēli	SCR_002
Teksta modulis	Displeja funkcija	TXT_001
	Ievades funkcija	TXT_002
	Funkcija ar rediģēšanu	TXT_003
	Ievades funkcija ar rediģēšanu	TXT_004
	Cīņas izsaukums	TXT_005
	Funkcija ar nejaušu mainīgo	TXT_006

### 3.1.4. Stāstu moduļi

Tabula 3.2 – stāsta moduļu specifikācija

Modulis	Funkcija	Identifikators
Galvenais modulis	Rīta funkcija	LOCM_001
	Nakts funkcija	LOCM_002
	Atpūsties istabā	LOCM_003
	Trenēties istabā	LOCM_004
	Apmeklēt krogu	LOCM_005

Modulis	Funkcija	Identifikators
Centra modulis	Tirgoties ar alķīmiķi	LOCC_001
	Tirgoties ar bruņotāju	LOCC_002
	Apmeklēt saunu	LOCC_003
	Apmeklēt festivālu	LOCC_004
	Apmeklēt baznīcu	LOCC_005
	Apmeklēt pili	LOCC_006
	Apmeklēt mēru	LOCC_007
	Eksportēt spēli	LOCC_008
Priekšpilsētas modulis	Apmeklēt bordeli	LOCK_001
	Apmeklēt Ilgas māju	LOCK_002
	Pastaigāties pa pilsētu	LOCK_003
Ārpilsētas modulis	Pastaigāties pa mežu/upi	LOCO_001
	Medīt mežā/upē	LOCO_002
	Apmeklēt kalnu	LOCO_003
	Apmeklēt kabīni	LOCO_004

### 3.1.5. Paziņojumi

Paziņojumu teksts ir tulkots no spēles angļu tulkojuma.

*Tabula 3.3 – paziņojumu tabula*

Identifikators	Funkcijas, kas lieto paziņojumu	Paziņojums

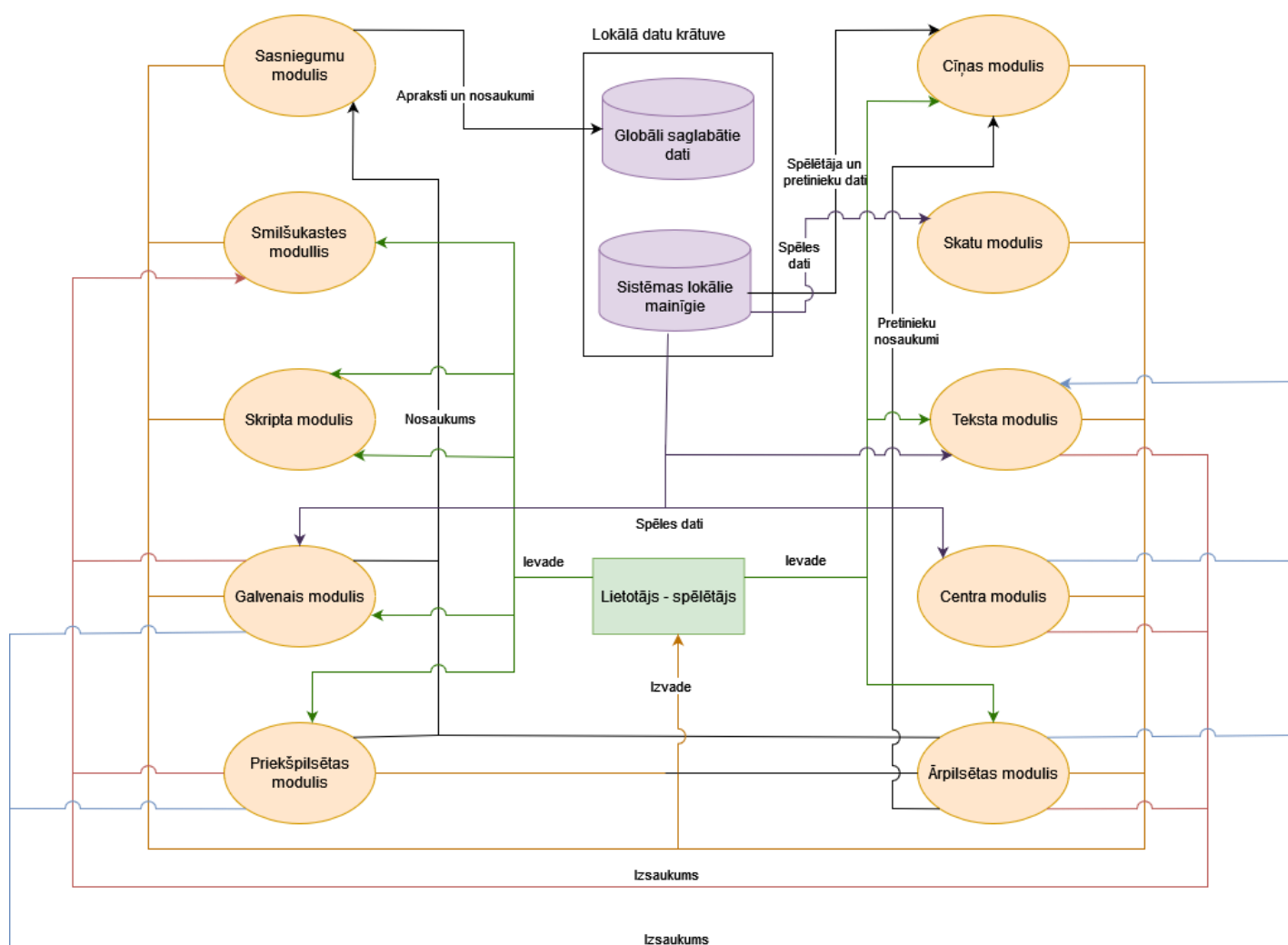
Identifikators	Funkcijas, kas lieto paziņojumu	Paziņojums
M001	ACH_002 ACH_003	“[sasnieguma nosaukums] [sasnieguma teksts]”
M002	ACH_003	”Sasniegums bloķēts.”
M003	COM_005 SBX_007	“Sakāve!”
M004	COM_005	“Forlanam izdevās bēgt no cīņas. Nākamo reizi.”
M005	COM_005	“Uzvara!”
M006	SBX_009	<attēls un skaņa>
M007	VIS_001	“Vai esat pārliecināts, ka vēlaties iziet?”
M008	VIS_002	“Dialoga vēsture ir tukša.”
M009	VIS_003	“Vai esat pārliecināts, ka vēlaties izdzēst šo saglabāto spēli?”
M010	VIS_003	“Vai esat pārliecināts, ka vēlaties pārrakstīt saglabāto spēli?”
M011	VIS_003	“Ielādēšana zaudēs nesaglabāto progresu. Vai vēlaties turpināt?”
M012	SCR_001	“Vai jūs esat sasniedzis 18 gadu vecumu?”
M013	LOCC_001	“Eksportēšana veiksmīga!”
M014	LOCO_003	“Man te vēl nav ko darīt.”
M015	SBX_001	“Krāpties ir slikti!”

### 3.1.6. Pārbaudes

Tabula 3.4 – pārbauzu tabula

Identifikators	Funkcijas, kas lieto pārbaudi	Pārbaudes apraksts	Pārbaudes rezultāts, ja patiesa
T1	COM_006	Pārbauda, vai niknuma punkti <i>pc.rp</i> nepārsniedz maksimālos <i>pc.rpl</i>	Tiek iestatīts <i>pc.rp = pc.rpl</i> .
T2	COM_006	Pārbauda, vai pretinieka <i>hp</i> ir 0	Iestatīts pretinieks <i>enemy[num] = 'none'</i> .
T3	COM_004	Pārbauda, vai spēlētāja <i>pc.hp</i> ir 0	Iestatīts <i>lost = True</i> , izsaukta funkcija COM_005.

### 3.1.7. Funkciju sadalījums pa moduļiem

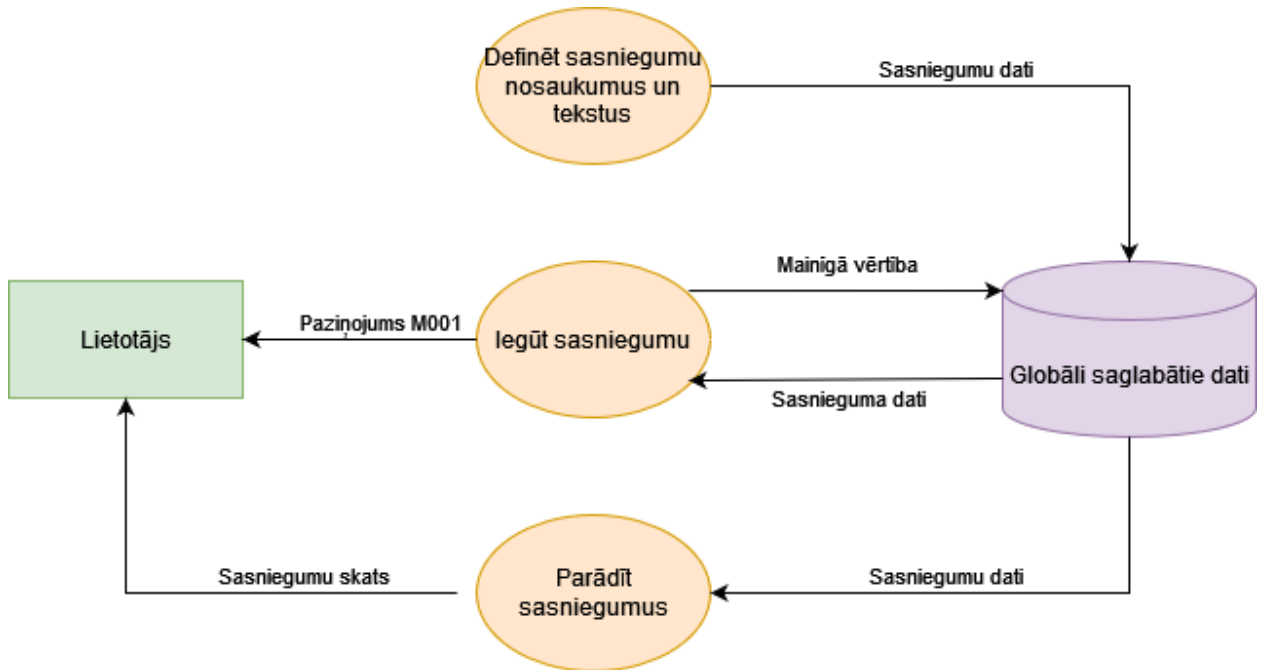


Attēls 3.1 - 1.līmeņa datu plūsmas diagramma

Attēlā 3.1 attēlota sistēmas 1.līmeņa datu plūsmas diagramma ar lietotāju, lokālo datu krātuvi un sistēmas desmit moduļiem.

Katrs modulis atbilst savam .rpy failam, kā aprakstīts nodaļā 3.1.2.

### 3.1.8. Sasniegumu modulis



Attēls 3.2 - sasniegumu moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.2. attēlota sasniegumu moduļa 2.līmeņa datu plūsmas diagramma ar 3 funkcijām, tālāk sauktas par ACH\_(funkcijas skaitlis).

Tabula 3.5 - Definēt sasniegumu nosaukumus un tekstu

Identifikators: ACH_001
Mērķis
Funkcijas mērķis ir sistēmā definēt sasniegumu nosaukumus un tiem pievienoto tekstu.
Ievaddati
1. Iepriekš definēts saraksts ar sasniegumu nosaukumiem 2. Iepriekš definēts saraksts ar sasniegumu tekstiem
Apstrāde
Ievaddatos dotie saraksti tiek saglabāti sistēmā kā mainīgie.
Izvaddati
–

Tabula 3.6 - Iegūt sasniegumu

Identifikators: ACH_002
Mērķis
Funkcijas mērķis ir reģistrēt iegūtu sasniegumu.
Ievaddati
<ol style="list-style-type: none"> <li>1. Sasnieguma nosaukums</li> <li>2. Saraksti ar sasniegumu nosaukumiem un to tekstiem no ACH_001</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Iegūto sasniegumu sarakstam tiek pievienots sasnieguma nosaukums un tam atbilstošais teksts no ACH_001 definētā saraksta.</li> <li>2. Sasniegums tiek globāli saglabāts kā iegūts.</li> </ol>
Izvaddati
Paziņojums M001, kas satur sasnieguma nosaukumu un tekstu.

Tabula 3.7 - Parādīt sasniegumus

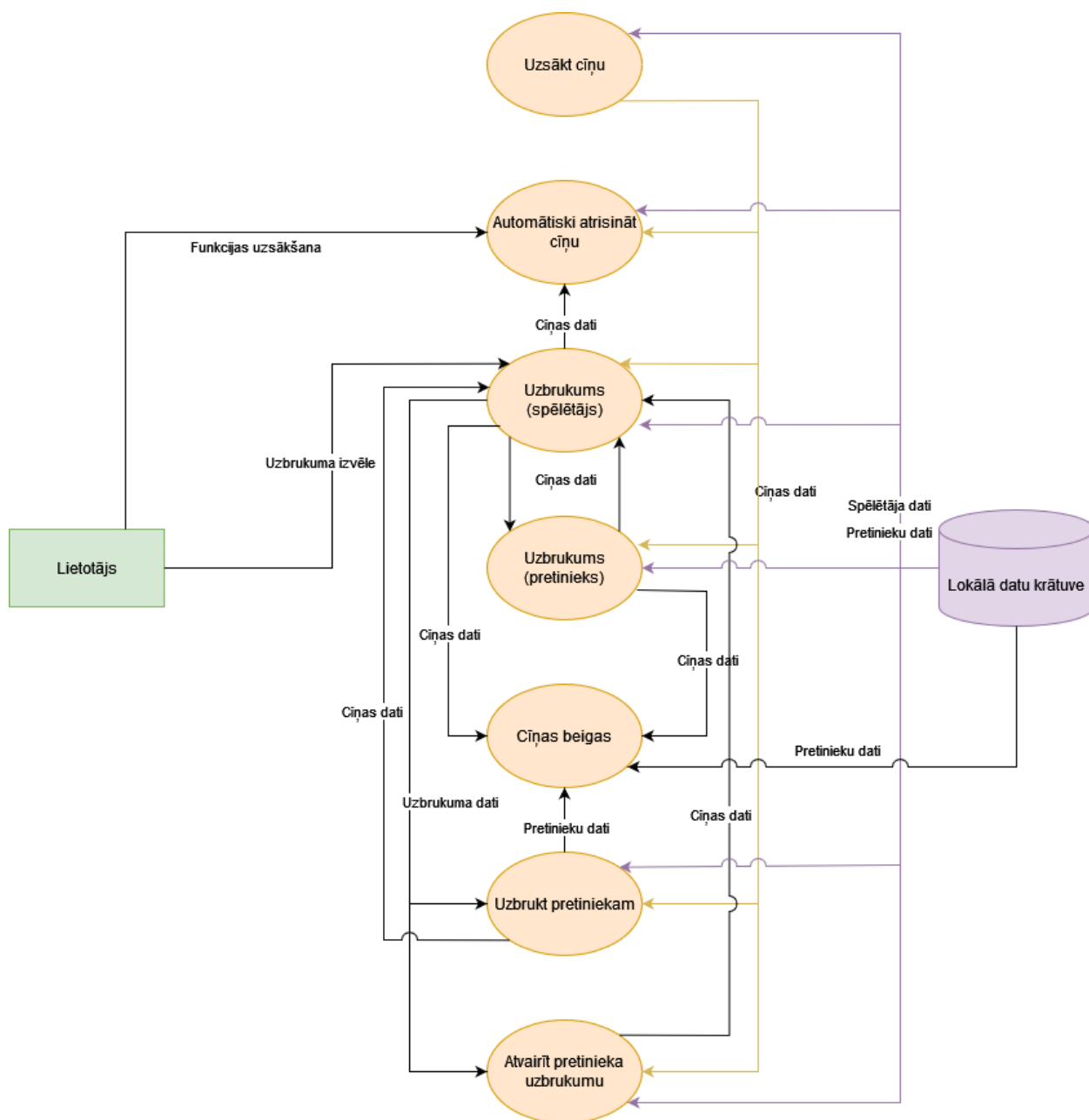
Identifikators: ACH_003
Mērķis
Funkcijas mērķis ir parādīt iegūto un neiegūto sasniegumu/atmiņu sarakstu.
Ievaddati
<ol style="list-style-type: none"> <li>1. Globāli saglabātais iegūto sasniegumu mainīgo saraksts</li> <li>2. Saraksti ar sasniegumu nosaukumiem un to tekstiem no ACH_001</li> <li>3. Globāli saglabātais iegūto atmiņu mainīgo saraksts</li> </ol>
Apstrāde

1. Tiek pārbaudīts iegūto sasniegumu saraksts.
  - a. Ja sasniegums ir atzīmēts kā iegūts, skatā tiek parādīts tā nosaukums un atbilstošais teksts no sarakstiem, kas nāk no ACH\_001, jeb paziņojums M001.
  - b. Ja sasniegums nav atzīmēts kā iegūts, skatā tas tiek parādīts kā paziņojums M002.
2. Priekš atmiņām – katrai vērtībai ir atbilstošs lodziņš skatā. Ja vērtība ir True, tad ir parādīts attēls no atmiņas un to noklikšķinot, tiek izsaukta atbilstošā funkcija TXT\_001. Citādi lodziņā ir attēls ar jautājuma zīmi – atmiņa nav pieejama.

#### Izvaddati

Skatā atšķirīgi parādīti iegūtie un neiegūtie sasniegumi; atmiņu skats un iespējams TXT\_001 izsaukums.

### 3.1.9. Cīņu modulis



Attēls 3.3 - cīņu moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.3. attēlota cīņu moduļa 2.līmeņa datu plūsmas diagramma ar 7 funkcijām, tālāk sauktas par COM\_(funkcijas skaitlis).

Tabula 3.8 - Uzšākt cīņu

Identifikators: COM_001
Mērķis
Funkcijas mērķis ir izsaukt cīņas sistēmu, kurā tiek uzsākta jauna cīņa ar pretinieku.
Ievaddati
<ol style="list-style-type: none"> <li>1. Pretinieku saraksts – tiek padots no izsaucošās funkcijas</li> <li>2. Spēlētāja profila dati (dzīvības punkti, spēka punkti, inventārs) no lokālās datu krātuves</li> <li>3. Pretinieku dati no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek apstrādāts funkcijas ievaddatos padotais pretinieku saraksts – no lokālās datu krātuves tiek iegūti dati par pretiniekiem, kas atbilst sarakstā esošajiem nosaukumiem.</li> <li>2. Tiek nolasīti vajadzīgie spēlētāja dati – dzīvības punkti, spēka punkti un inventārs, kas maina šos atribūtus.</li> <li>3. Tiek izsaukta funkcija COM_003.</li> </ol>
Izvaddati
–

Tabula 3.9 - Automātiski atrisināt cīņu

Identifikators: COM_002
Mērķis
Funkcijas mērķis ir automātiski atrisināt cīņu.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja profila dati (dzīvības punkti, spēka punkti, inventārs) no lokālās datu krātuves</li> <li>2. Pretinieku dati no lokālās datu krātuves</li> <li>3. Iestatītie mainīgie no COM_001</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Katram pretiniekam (no viena līdz trim) pēc kārtas tiek nolasīta pretinieka aizsardzības precizitāte. <ol style="list-style-type: none"> <li>a. Ja pretinieka aizsardzības precizitāte ir lielāka par spēlētāja uzbrukuma precizitāti, tad tiek veikts viegls uzbrukums.</li> <li>b. Ja pretinieka aizsardzības precizitāte ir būtiski mazāka par spēlētāja uzbrukuma precizitāti, tad tiek veikts smags uzbrukums.</li> <li>c. Citādi tiek veikts vidējs uzbrukums.</li> </ol> </li> <li>2. Tā kā pēc katra veiksmīga uzbrukuma tiek uzkrāti niknuma punkti, atkarībā no to daudzuma tiek izsaukts viens no iespējamajiem īpašajiem uzbrukumiem.</li> <li>3. Pirmos divus soļus atkārtoti, kamēr vai nu visiem pretiniekiem dzīvības punktu skaits ir 0, vai spēlētāja dzīvības punktu skaits ir 0. Tiek izsaukta funkcija COM_005.</li> </ol>
Izvaddati
–

Tabula 3.10 - Uzbrukums (spēlētājs)

Identifikators: COM_003
Mērķis
Funkcijas mērķis ir veikt vienu cīņas kārtu no spēlētāja puses.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja profila dati (dzīvības punkti, spēka punkti, inventārs) no lokālās datu krātuves</li> <li>2. Pretinieku dati no lokālās datu krātuves</li> <li>3. Spēlētāja ievade – izvēlēta rīcība</li> <li>4. Iestatītie mainīgie no COM_001</li> </ol>
Apstrāde

<p>1. Tiek apstrādāta spēlētāja izvēle.</p> <p>a. Ja izvēlētā darbība ir viens no parastajiem uzbrukumiem, pretinieka izpēte vai inventāra apskate, darbība tiek veikta ar atbilstošo izvadi/apstrādi.</p> <p>i. Ja izvēlēts viens no parastajiem uzbrukumiem, tiek iestatīti uzbrukuma mainīgie un izsaukta funkcija COM_006.</p> <p>b. Ja izvēlētā darbība ir niknuma uzbrukums, tas tiek veikts tikai tad, ja spēlētājam ir pietiekami daudz niknuma punktu – tiek izsaukta funkcija COM_006.</p> <p>c. Ja izvēlētā darbība ir bēgt no cīņas, rezultāts ir atkarīgs no nejauša mainīgā. Veiksmīgs rezultāts izsauc COM_005.</p> <p>d. Ja izvēlētā darbība ir automātiska cīņa, tiek izsaukta funkcija COM_002.</p> <p>2. Tiek apstrādāti jebkādi paliekoši efekti, kas ir reģistrēti spēlētājam, kas izmaina spēlētāja atribūtus.</p> <p>3. Tiek veikta pārbaude: spēlētāja dzīvības punktu skaits ir 0, pretinieka dzīvības punktu skaits ir 0, 1c.solī ir veiksmīgs rezultāts. Ja kaut viens no šiem nosacījumiem izpildās, tiek izsaukts COM_005. Citādi tiek izsaukts COM_004.</p>
Izvaddati
–

Tabula 3.11 - Uzbrukums (pretinieks)

Identifikators: COM_004
Mērķis
Funkcijas mērķis ir veikt vienu cīņas kārtu no pretinieka puses.
Ievaddati
<p>1. Spēlētāja profila dati (dzīvības punkti, spēka punkti, inventārs) no lokālās datu krātuves</p> <p>2. Pretinieku dati no lokālās datu krātuves</p> <p>3. Iestatītie mainīgie no COM_001</p>
Apstrāde

<ol style="list-style-type: none"> <li>1. Pretinieka uzbrukuma spēks tiek nejauši aprēķināts. Pretinieka uzbrukuma precizitāte tiek aprēķināta no atribūtu datiem. Tiek veikts uzbrukums, kas ir veiksmīgs tikai tad, ja pretinieka uzbrukuma precizitāte ir lielāka par spēlētāja aizsardzības precizitāti.</li> <li>2. Ja tiek veikta uzbrukuma atvairīšana no spēlētāja puses, tiek izsaukta funkcija COM_007.</li> <li>3. Tiek veikta pārbaude: spēlētāja dzīvības punktu skaits ir 0, pretinieka dzīvības punktu skaits ir 0. Ja kaut viens no šiem nosacījumiem izpildās, tiek izsaukts COM_005. Citādi tiek izsaukts COM_003.</li> </ol>
Izvaddati
–

Tabula 3.12 - Cīņas beigas

Identifikators: COM_005
Mērķis
Funkcijas mērķis ir beigt cīņu.
Ievaddati
1. Mainīgais, kas padots izsaukuma brīdī no COM_003 vai COM_004.
Apstrāde
<ol style="list-style-type: none"> <li>1. Ja mainīgais atzīmē, ka cīņa ir beigusies ar spēlētāja nāvi (spēlētāja dzīvības punktu skaits ir 0), tiek izvadīts paziņojums M003.</li> <li>2. Ja mainīgais atzīmē, ka spēlētājs ir bēdzis no cīņas, tiek izvadīts paziņojums M004.</li> <li>3. Ja mainīgais atzīmē, ka cīņa ir beigusies ar uzvaru (visu pretinieku dzīvības punktu skaits ir 0), tiek mainīti spēlētāja atribūti (palielināta pieredze, palielināts naudas daudzums, mainīts uzdevumu progress), un tiek izvadīts paziņojums M005.</li> </ol>
Izvaddati
M003, M004, vai M005, atkarībā no ievaddatos dotā mainīgā.

Tabula 3.13 - Uzbrukt pretiniekam

Identifikators: COM_006
Mērķis
Funkcijas mērķis ir veikt uzbrukumu pretiniekam.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja profila dati (dzīvības punkti, spēka punkti, inventārs) no lokālās datu krātuves</li> <li>2. Pretinieku dati no lokālās datu krātuves</li> <li>3. Uzbrukuma veids – no COM_003</li> <li>4. Iestatītie mainīgie no COM_001</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek aprēķinātas uzbrukumu iespējamības un uzbrukuma spēks. <ol style="list-style-type: none"> <li>a. Uzbrukums ir veiksmīgs tikai tad, ja iespējamība ir pietiekami liela un uzbrukuma spēks nav negatīvs.</li> </ol> </li> <li>2. Ja uzbrukuma veids ir niknuma uzbrukums, uzbrukuma spēks ir lielāks.</li> <li>3. Tiek izvadīti attiecīgie skati un teksts.</li> <li>4. Tiek veiktas pārbaudes T1 un T2 un veiktas attiecīgās darbības.</li> </ol>
Izvaddati
–

Tabula 3.14 - Atvairīt pretinieka uzbrukumu

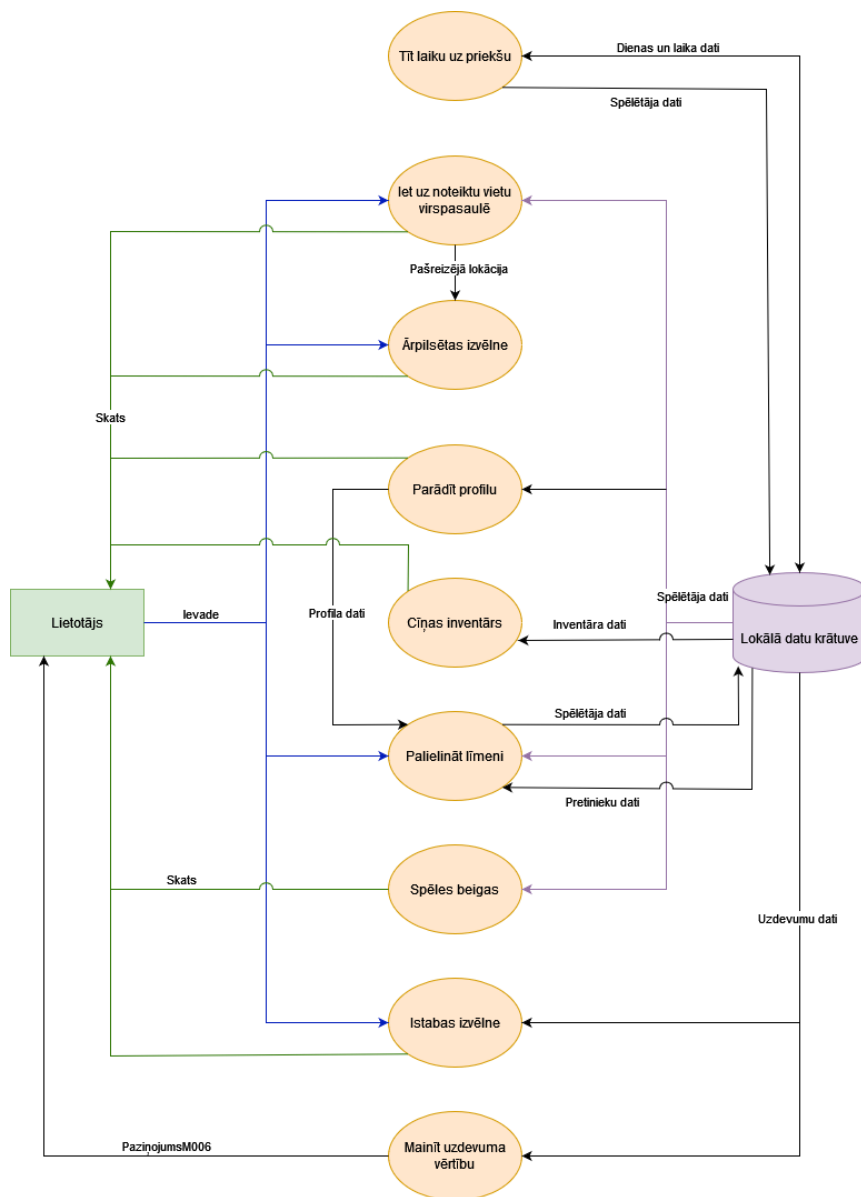
Identifikators: COM_007
Mērķis
Funkcijas mērķis ir atvairīt pretinieka uzbrukumu
Ievaddati
<ol style="list-style-type: none"> <li>1. Iestatīts mainīgais no COM_003</li> <li>2. Pretinieka mainīgais no COM_004</li> </ol>
Apstrāde

1. Atkarībā no nejauša mainīgā tiek aprēķināta atvairīšanas iespējamība.
2. Ja atvairīšana ir veiksmīga, atkarībā no mainīgā ir divas iespējas:
  - a. Pretinieka uzbrukums tiek veiksmīgi bloķēts.
  - b. Pretinieka uzbrukums tiek veiksmīgi bloķēts un tiek izsaukts pretuzbrukums – izsaukta funkcija COM\_006.

Izvaddati

–

### 3.1.10. Smilšukastes modulis



Attēls 3.4 - smilšukastes moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.4. attēlota smilšukastes moduļa 2.līmeņa datu plūsmas diagramma ar 9 funkcijām, tālāk sauktas par SBX\_(funkcijas skaitlis).

Tabula 3.15 - Tīt laiku uz priekšu

Identifikators: SBX_001
Mērķis
Funkcijas mērķis ir mainīt laika un dienas atribūtus.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Diena, dienu skaits un laiks</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Ja spēlētāja datos ir atzīmēta spēlētāja nāve, tiek izsaukta funkcija SBX_007.</li> <li>2. Tiek mainīts laiks par vienu vienību – vienība šajā kontekstā ir ¼ dienas, tātad laika maiņa notiek šādi: rīts -&gt; diena -&gt; vakars -&gt; nakts -&gt; nākamās dienas rīts.</li> <li>3. Tiek mainīta diena, ja laiks mainās no nakts uz rītu.</li> <li>4. Ja ir sasniegti noteikti posmi uzdevumos, tiek izsaukta viena no atbilstošajām teksta moduļa funkcijām.</li> <li>5. Tiek rediģēti spēlētāja mainīgie, kas mainās ar laiku (netīrība, maģiski efekti).</li> <li>6. Tiek izsaukta funkcija SBX_006, ja ir sasniegts noteikts skaits pieredzes punktu.</li> <li>7. Tiek izsaukta funkcija SBX_002.</li> </ol>
Izvaddati
–

Tabula 3.16 - Iet uz noteiktu vietu virspasaulē

Identifikators: SBX_002
Mērķis
Funkcijas mērķis ir ļaut spēlētājam izvēlēties noteiktu lokāciju virspasaulē.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja pašreizējā lokācija</li> <li>3. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Atkarībā no pašreizējās lokācijas tiek izsaukts atbilstošais skats – kartes izvēlne.</li> <li>2. Atkarībā no spēlētāja izvēles tiek izsaukta viena no sekojošajām funkcijām: SBX_001, SBX_003, LOCM_005, LOCC_001 – LOCC_008, LOCK_001 – LOCK_003.</li> </ol>
Izvaddati
.-

Tabula 3.17 - Ārpilsētas izvēlne

Identifikators: SBX_003
Mērķis
Funkcijas mērķis ir ļaut spēlētājam izvēlēties noteiktu lokāciju ārpilsētā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Dienas laiks no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek nejauši aprēķināts mainīgais, kas nosaka, vai notiks nejaušs lāča uzbrukums, ja dienas laiks ir diena. Ja abi nosacījumi izpildās, tiek izsaukta funkcija TXT_005.</li> <li>2. Citādi atkarībā no spēlētāja izvēles tiek izsaukta viena no sekojošajām funkcijām: SBX_001, LOCO_001 – LOCO_004.</li> </ol>
Izvaddati
-

Tabula 3.18 - Parādīt profilu

Identifikators: SBX_004
Mērķis
Funkcijas mērķis ir parādīt spēlētāja profilu.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Tiek nolasīti skatam nepieciešamie spēlētāja atribūti.</li> <li>2. Ja spēlētājam datos ir atzīmēti noteikti bonusi, tiek iegūts šiem bonusiem atbilstošais teksts no lokālās datu krātuves.</li> </ol>
Izvaddati
Skats ar spēlētāja datiem.

Tabula 3.19 - Cīņas inventārs

Identifikators: SBX_005
Mērķis
Funkcijas mērķis ir parādīt spēlētāja inventāru.
Ievaddati
<ol style="list-style-type: none"> <li>1. Inventāra dati no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek nolasīti skatam nepieciešamie inventāra atribūti. Ja katram iespējamajam piederošam objektam daudzums ir lielāks par 0, tam tiek iegūts atbilstošais nosaukums un apraksts no lokālās datu krātuves.</li> <li>2. Uzklikšķinot uz kādas no lietojamajām inventāra lietām, tā tiek izmantota un spēlētāja datiem pievienoti attiecīgie atribūti.</li> </ol>
Izvaddati
Skats ar spēlētāja inventāru.

Tabula 3.20 - Palielināt līmeni

Identifikators: SBX_006
Mērķis
Funkcijas mērķis ir palielināt spēlētāja līmeni.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Pretinieku dati no lokālās datu krātuves</li> <li>3. Spēlētāja ievade – izvēle</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Tiek palielinātas spēlētāja atribūtu vērtības.</li> <li>2. Tiek palielinātas noteiktu pretinieku atribūtu vērtības.</li> <li>3. Tiek izsaukta funkcijas SBX_004 skats.</li> <li>4. Profila skatā tiek dota izvēle palielināt vai nu spēku, vai veiklību. Izvēlētais atribūts tiek palielināts.</li> </ol>
Izvaddati
–

Tabula 3.21 - Spēles beigas

Identifikators: SBX_007
Mērķis
Funkcijas mērķis ir paziņot par spēles beigām stāsta laikā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Ja spēlētājs zaudē cīņā, tiek izvadīts paziņojums M003.</li> <li>2. Citādi parādās papildus teksts par spēles zaudēšanu un izsaukta spēles beigu skats un VIS_001.</li> </ol>
Izvaddati
Spēles beigu skats un izsaukta funkcija VIS_001.

Tabula 3.22 - Istabas izvēlne

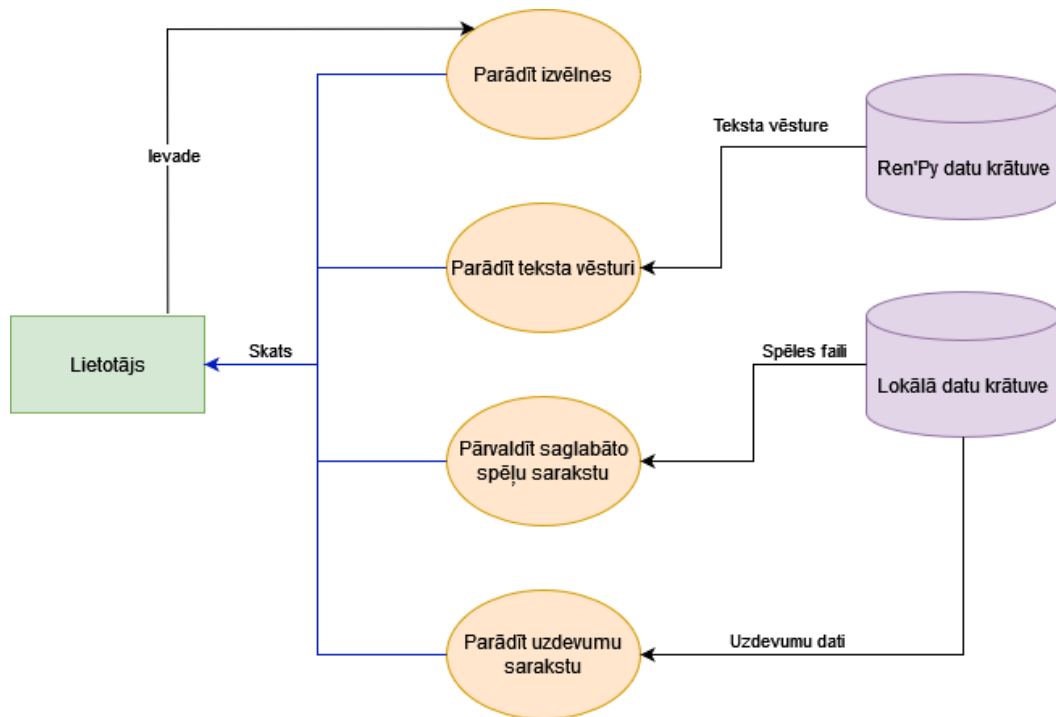
Identifikators: SBX_008
Mērķis
Funkcijas mērķis ir ļaut spēlētājam izvēlēties noteiktu darbību savā istabā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Uzdevumu dati no lokālās datu krātuves</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Atkarībā no spēlētāja izvēles tiek izsaukta viena no sekojošajām funkcijām: SBX_001, SBX_002, LOCM_003 – LOCM_005.</li> <li>2. Ja ir sasniegti noteikti posmi uzdevumos un ir noteikts dienas laiks, atkarībā no spēlētāja izvēles var izsaukt kādu no funkcijām TXT_001 - TXT_004.</li> </ol>
Izvaddati
–

*Tabula 3.23 - Mainīt uzdevuma vērtību*

Identifikators: SBX_009
Mērķis
Funkcijas mērķis ir paziņot par jauna uzdevuma posma sasniegšanu.
Ievaddati
<ol style="list-style-type: none"> <li>1. Uzdevuma nosaukums</li> <li>2. Posma vērtība</li> </ol>
Apstrāde
1. Lokālajā datu krātuvē mainīgajam pie attiecīgā uzdevuma nosaukuma tiek rediģēta vērtība uz ievaddatos doto.
Izvaddati
Paziņojums M006.

### 3.1.11. Skatu modulis



Attēls 3.5 - skatu moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.5. attēlota skatu moduļa 2.līmeņa datu plūsmas diagramma ar 4 funkcijām, tālāk sauktas par VIS\_(funkcijas skaitlis).

Tabula 3.24 - Parādīt izvēlnes

Identifikators: VIS_001
Mērķis
Funkcijas mērķis ir parādīt spēles izvēlnes.
Ievaddati
1. Spēlētāja ievade – izvēle
Apstrāde

<ol style="list-style-type: none"> <li>1. Tiek parādīts spēles galvenais skats.</li> <li>2. Atkarībā no spēlētāja izvēles tiek izsaukta viena no sekojošajām funkcijām: SCR_002, VIS_003, ACH_003, vai atvērti iestatījumi, vai arī lietotāja ierīcē noklusējuma pārlūkprogrammā atvērta hipersaite. <ol style="list-style-type: none"> <li>a. Ja tiek izvēlēta spēles aizvēršana, tiek parādīts paziņojums M007 un, ja atbilde ir apstiprinoša, spēles logs tiek aizvērts.</li> </ol> </li> <li>3. Iestatījumu izvēlnē var mainīt spēles iestatījumus, gūt palīdzību par spēles kontrolēm un redzēt informāciju par spēli – šī ir daļa no Ren'Py noklusējuma koda.<sup>[14]</sup></li> </ol>
Izvaddati
–

Tabula 3.25 - Parādīt teksta vēsturi

Identifikators: VIS_002
Mērķis
Funkcijas mērķis ir parādīt stāsta teksta vēsturi.
Ievaddati
1. Stāsta teksta vēsture – saglabāta Ren'Py dzinēja failos
Apstrāde
<ol style="list-style-type: none"> <li>1. Katrs punkts teksta vēsturē tiek parādīts skatā – sākot no jaunākā.</li> <li>2. Ja teksta vēsture ir tukša, tiek parādīts paziņojums M008.</li> </ol>
Izvaddati
Skats ar stāsta teksta vēsturi vai paziņojums M008.

Tabula 3.26 - Pārvaldīt saglabāto spēļu sarakstu

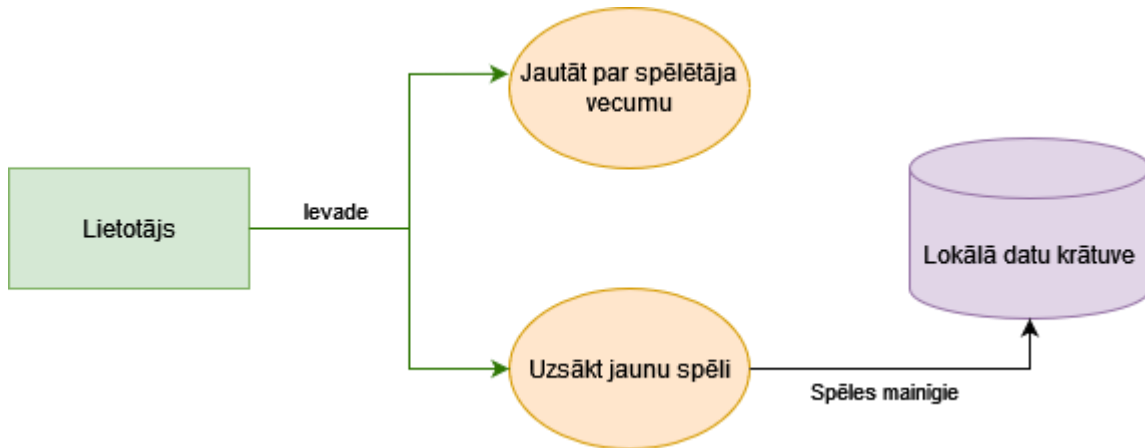
Identifikators: VIS_003
Mērķis
Funkcijas mērķis ir pārvaldīt saglabāto spēļu vēsturi.
Ievaddati
1. Saglabāto spēļu saraksts – failu saraksts spēles mapē.

<b>Apstrāde</b>
<ol style="list-style-type: none"> <li>1. Katrs eksistējošais fails tiek parādīts tam atbilstošajā lodziņā, balstoties uz faila nosaukumu (atbilstoši Ren'Py dokumentācijai failu nosaukumi atbilst noteiktai struktūrai un ir automātiski piešķirti).</li> <li>2. Ja izvēlētam failam tiek nospiests taustiņš “Delete”, tiek parādīts paziņojums M009. Ja atbilde ir apstiprinoša, fails tiek dzēsts.</li> <li>3. Ja lodziņš ir tukšs, to noklikšķinot, tiek saglabāts jauns spēles fails.</li> <li>4. Ja lodziņš nav tukšs, to noklikšķinot: <ol style="list-style-type: none"> <li>a. Ja spēle tiek saglabāta, tiek parādīts paziņojums M010 un pēc apstiprinošas atbildes tiek saglabāts jauns spēles fails, kas pārraksta iepriekšējo.</li> <li>b. Ja spēle tiek ielādēta un spēlētājs pašlaik ir jau ielādējis spēli, tiek parādīts paziņojums M011 un pēc apstiprinošas atbildes ielādēts attiecīgais spēles fails.</li> <li>c. Citādi tiek vienkārši ielādēts attiecīgais spēles fails.</li> </ol> </li> </ol>
<b>Izvaddati</b>
–

*Tabula 3.27 - Parādīt uzdevumu sarakstu*

<b>Identifikators: VIS_004</b>
<b>Mērķis</b>
Funkcijas mērķis ir parādīt uzdevumu sarakstu.
<b>Ievaddati</b>
1. Mainīgo kopa – uzdevumu nosaukumi un tiem atbilstošās posmu vērtības.
<b>Apstrāde</b>
1. Katram uzdevumam tiek veikta pārbaude – ja posma vērtība ir 0, tiek parādīts teksts “Jūs vēl neesat atradis šo uzdevumu.” Citādi tiek parādīts uzdevuma nosaukums un posma vērtībai atbilstošais teksts atbilstošajā uzdevumu lapā.
<b>Izvaddati</b>
–

### 3.1.12. Skripta modulis



Attēls 3.6 - skriptu moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.6. attēlota skriptu moduļa 2.līmeņa datu plūsmas diagramma ar 2 funkcijām, tālāk sauktas par SCR\_(funkcijas skaitlis).

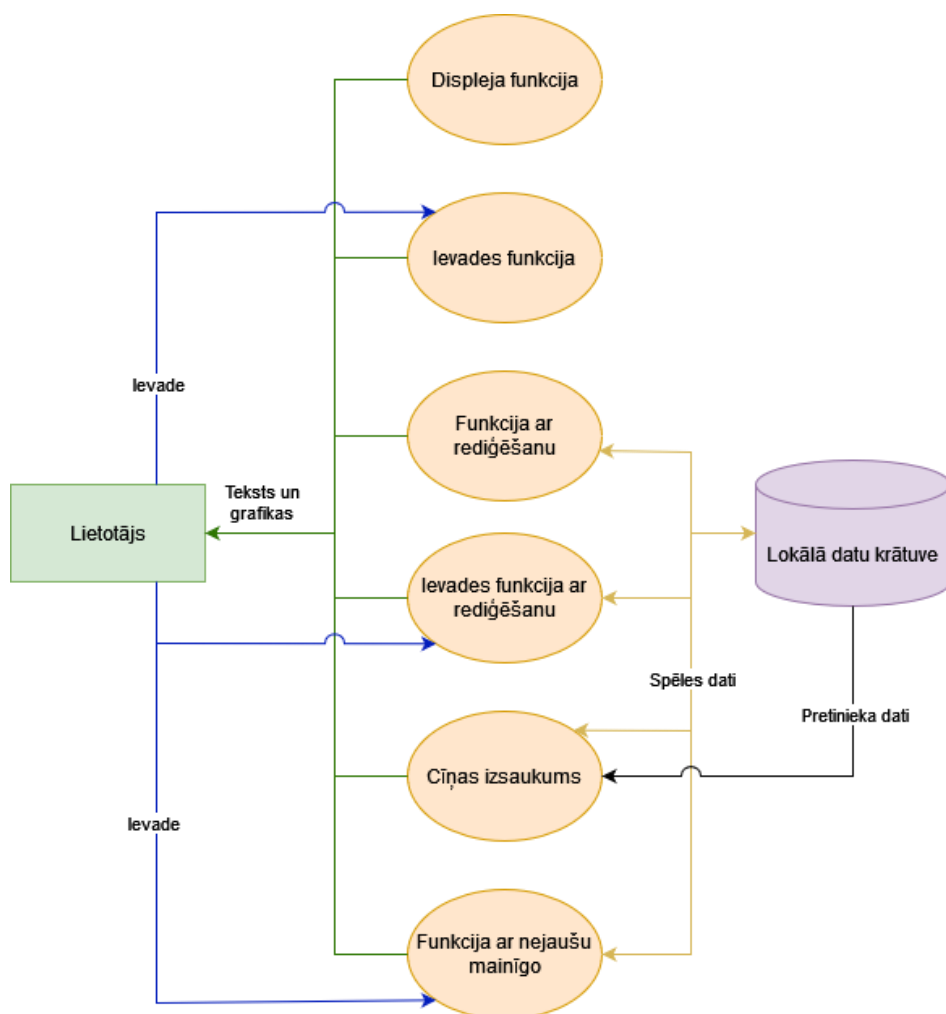
Tabula 3.28 - Jautāt par spēlētāja vecumu

Identifikators: SCR_001
Mērķis
Funkcijas mērķis ir pārbaudīt spēlētāja vecumu.
Ievaddati
1. Spēlētāja ievade – izvēle
Apstrāde
1. Tiek parādīts paziņojums M012. Ja atbilde ir apstiprinoša, tiek izsaukta funkcija VIS_001. Ja atbilde ir neapstiprinoša, spēles logs tiek aizvērts.
Izvaddati
VIS_001 izsaukums vai spēles loga aizvēršana.

Tabula 3.29 - Uzsākt jaunu spēli

Identifikators: SCR_002
Mērķis
Funkcijas mērķis ir uzsākt jaunu spēli.
Ievaddati
1. Spēlētāja ievade – izvēle
Apstrāde
1. Atkarībā no spēlētāja izvēles spēles grūtība tiek iestatīta kā viegla, vidēja vai grūta. 2. Tiek izsaukta atbilstošā funkcija TXT_004.
Izvaddati
–

### 3.1.13. Teksta modulis



Attēls 3.7 - teksta moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.7. attēlota teksta moduļa 2.līmeņa datu plūsmas diagramma ar 6 funkcijām, tālāk sauktas par TXT\_(funkcijas skaitlis).

Tabula 3.30 - Displeja funkcija

Identifikators: TXT_001
Mērķis
Funkcijas mērķis ir parādīt stāsta tekstu.
Ievaddati
1. Ainas nosaukums

Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek izsaukta attiecīgā aina.</li> <li>2. Pēc teksta parādīšanas tiek izsaukta cita teksta moduļa funkcija, SBX_001 vai SBX_002.</li> </ol>
Izvaddati
Stāsta teksts un atbilstošās grafikas.

Tabula 3.31 - Ievades funkcija

Identifikators: TXT_002
Mērķis
Funkcijas mērķis ir parādīt stāsta tekstu, kas mainās, atbilstoši spēlētāja ievadei vai spēlētāja atribūtiem.
Ievaddati
<ol style="list-style-type: none"> <li>1. Ainas nosaukums</li> <li>2. Spēlētāja ievade – izvēle</li> <li>3. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Atkarībā no spēlētāja izvēles vai atribūtiem tiek parādīts atšķirīgs teksts.</li> <li>2. Tiek izsaukta cita teksta moduļa funkcija, SBX_001 vai SBX_002.</li> </ol>
Izvaddati
Stāsta teksts un atbilstošās grafikas.

Tabula 3.32 - Funkcija ar rediģēšanu

Identifikators: TXT_003
Mērķis
Funkcijas mērķis ir parādīt stāsta tekstu un rediģēt noteiktus mainīgos lokālajā datu krātuvē.
Ievaddati
<ol style="list-style-type: none"> <li>1. Ainas nosaukums</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Pēc noteikta teksta parādīšanas tiek rediģēts funkcijā atzīmēts lokālās datu krātuves mainīgais.</li> <li>2. Tiek izsaukta cita teksta moduļa funkcija, SBX_001 vai SBX_009.</li> </ol>
Izvaddati
Stāsta teksts un atbilstošās grafikas.

Tabula 3.33 - Ievades funkcija ar rediģēšanu

Identifikators: TXT_004
Mērķis
Funkcijas mērķis ir parādīt stāsta tekstu un rediģēt noteiktus mainīgos lokālajā datu krātuvē, balstoties uz spēlētāja izvēli vai atribūtiem.
Ievaddati
<ol style="list-style-type: none"> <li>1. Ainas nosaukums</li> <li>2. Spēlētāja ievade – izvēle</li> <li>3. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Atkarībā no spēlētāja izvēles vai atribūtiem tiek parādīts atšķirīgs teksts un rediģēts funkcijā atzīmēts lokālās datu krātuves mainīgais.</li> <li>2. Tiek izsaukta cita teksta moduļa funkcija, SBX_001 vai SBX_009.</li> </ol>
Izvaddati
Stāsta teksts un atbilstošās grafikas.

Tabula 3.34 - Cīņas izsaukums

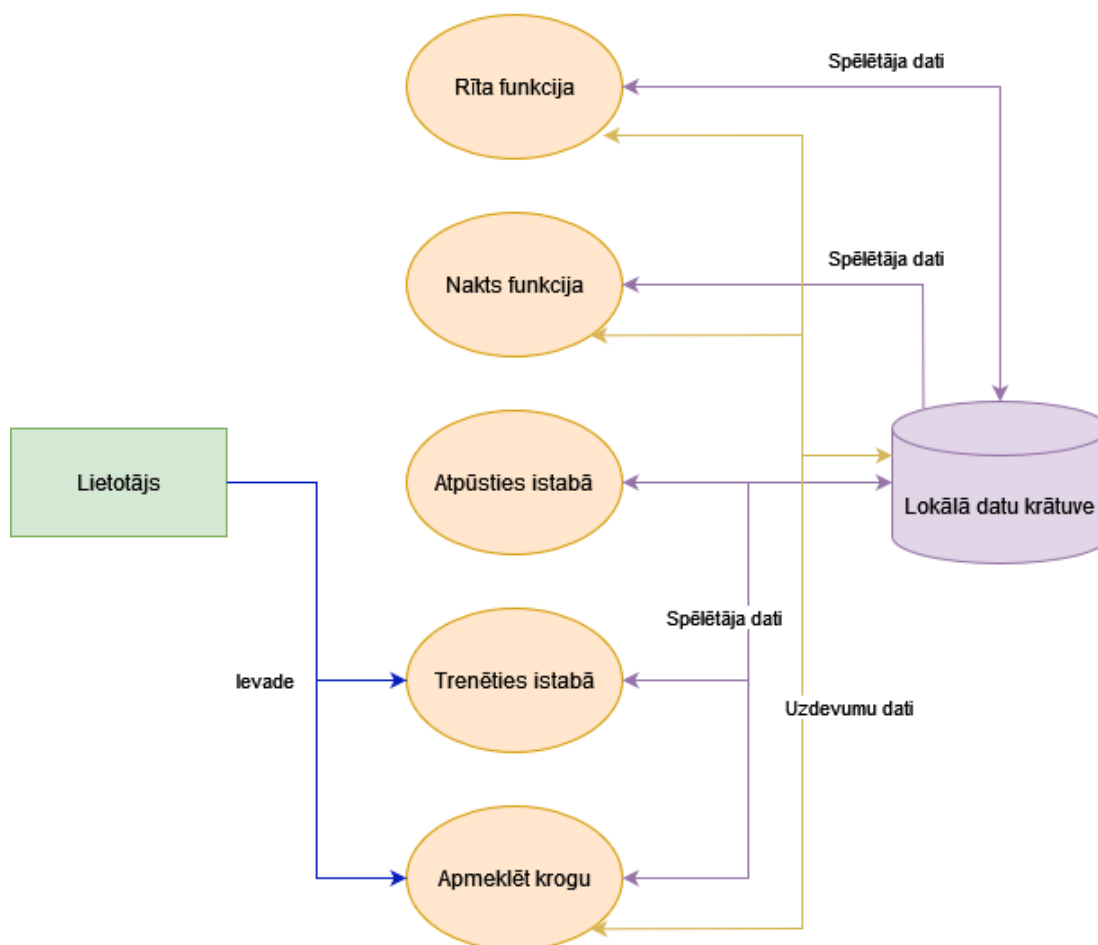
Identifikators: TXT_005
Mērķis
Funkcijas mērķis ir parādīt stāsta tekstu un izsaukt cīņu.
Ievaddati
<ol style="list-style-type: none"> <li>1. Ainas nosaukums</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Pēc noteikta teksta parādīšanas tiek piešķirtas vērtības pretinieku mainīgajiem un izsaukta funkcija COM_001.</li> <li>2. Ja cīņā spēlētājs nav miris, tiek izsaukta cita teksta moduļa funkcija, SBX_001 vai SBX_009.</li> <li>3. Ja cīņā spēlētājs ir miris, tiek izsaukta funkcija SBX_007.</li> </ol>
Izvaddati
Stāsta teksts un atbilstošās grafikas, COM_001 izsaukums.

Tabula 3.35 - Funkcija ar nejaušu mainīgo

Identifikators: TXT_006
Mērķis
Funkcijas mērķis ir parādīt stāsta tekstu, atkarībā no nejauši aprēķināta mainīgā.
Ievaddati
1. Ainas nosaukums
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek nejauši aprēķināta mainīgā vērtība.</li> <li>2. Atkarībā no šī mainīgā tiek parādīts atšķirīgs teksts, pēc kura tiek izsaukta cita teksta moduļa funkcija, SBX_001 vai SBX_002.</li> </ol>
Izvaddati
Stāsta teksts un atbilstošās grafikas.

### 3.1.14. Galvenais modulis



Attēls 3.8 - galvenā moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.8. attēlota galvenā moduļa 2.līmeņa datu plūsmas diagramma ar 5 funkcijām, tālāk sauktas par LOCM\_(funkcijas skaitlis).

Tabula 3.36 - Rīta funkcija

Identifikators: LOCM_001
Mērķis
Funkcijas mērķis ir tikt galā ar darbībām, kas notiek no rīta.
Ievaddati
1. Spēlētāja dati no lokālās datu krātuves

2. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek atiestatīti spēlētāja dzīvības un niknuma punkti.</li> <li>2. Atkarībā no uzdevuma posma tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>3. Ja ir sasniegts noteikts pieredzes punktu skaits, tiek izsaukta funkcija SBX_006.</li> </ol>
Izvaddati
–

Tabula 3.37 - Nakts funkcija

Identifikators: LOCM_002
Mērķis
Funkcijas mērķis ir tikt galā ar darbībām, kas notiek naktī.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Atkarībā no uzdevuma posma tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>2. Ja spēlētājam nav pietiekami daudz naudas, tiek parādīts atbilstošais teksts un izsaukta funkcija SBX_007. Citādi tiek izsaukta funkcija LOCM_001.</li> </ol>
Izvaddati

Tabula 3.38 - Atpūsties istabā

Identifikators: LOCM_003
Mērķis
Funkcijas mērķis ir ļaut spēlētājam atpūsties istabā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Tiek parādīts stāsta teksts un atbilstošās grafikas.</li> <li>2. Tiek palielināts spēlētāja dzīvības punktu skaits.</li> <li>3. Tiek izsaukta funkcija SBX_001.</li> </ol>
Izvaddati
–

Tabula 3.39 - Trenēties istabā

Identifikators: LOCM_004
Mērķis
Funkcijas mērķis ir ļaut spēlētājam trenēties istabā – palielinot spēlētāja atribūtus.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Atkarībā no spēlētāja izvēles tiek palielināts vai nu spēlētāja spēka vai veiklības atribūts, un parādīts atbilstošais teksts.</li> <li>2. Tiek izsaukta funkcija SBX_001.</li> </ol>
Izvaddati
–

Tabula 3.40 - Apmeklēt krogu

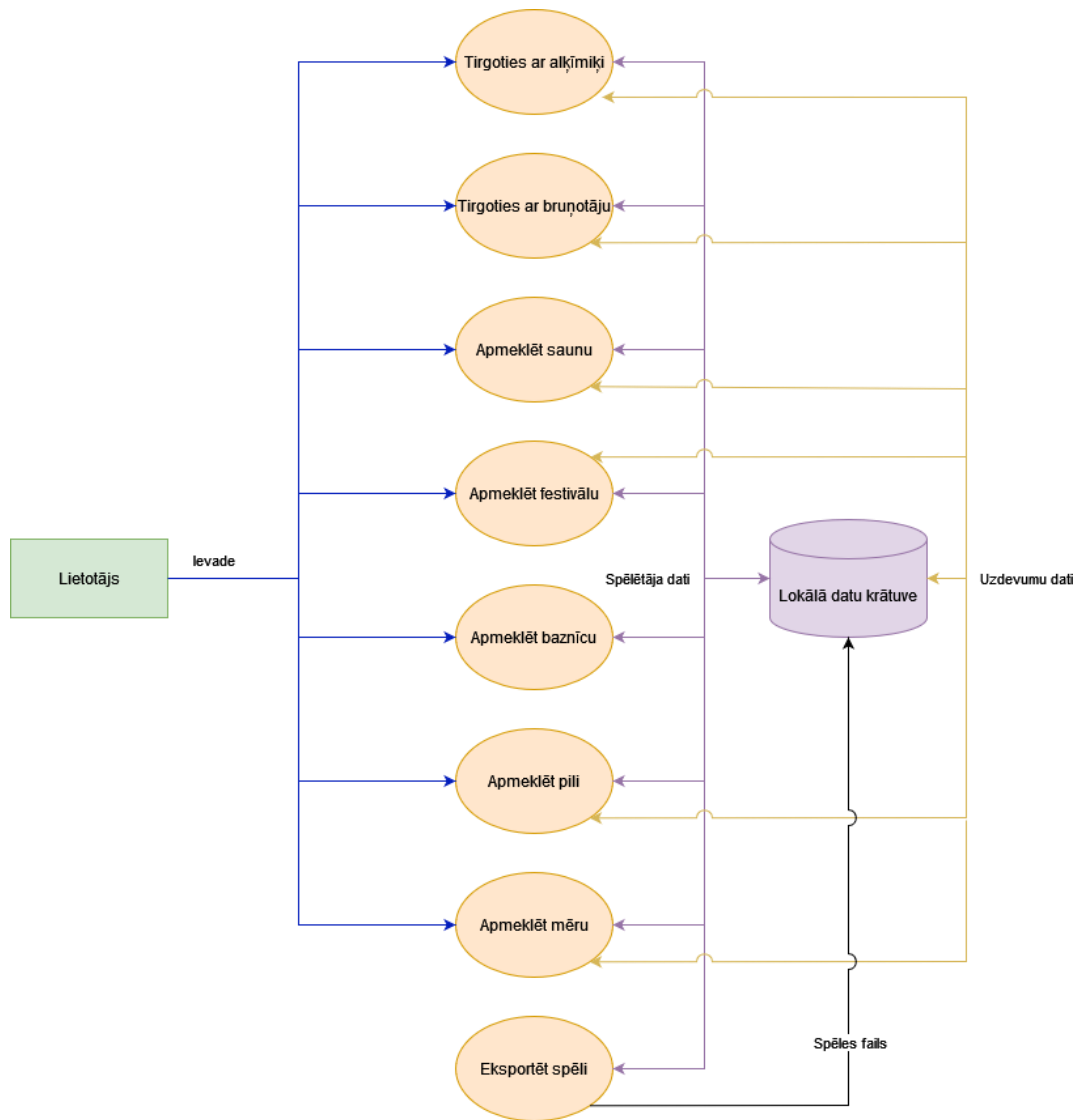
Identifikators: LOCM_005
Mērķis
Funkcijas mērķis ir veikt darbības krogā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde

1. Tiek parādīts atbilstošais teksts. Ja spēlētāja netīrības atribūts ir pārāk liels, tiek izsaukta funkcija SBX\_002. Citādi:
2. Spēlētājam tiek dota izvēle starp darbībām.
  - a. Ja ir sasniegts noteikts posms kādā uzdevumā, tiek dota opcija, kas saistīta ar doto uzdevumu – tiek izsaukta atbilstošā teksta moduļa funkcija.
  - b. Ja tiek izvēlēts “pasūtīt ēdienu”, spēlētāja naudas daudzums tiek samazināts un tiek izsaukta atbilstošās teksta moduļa funkcijas, atkarībā no spēlētāja datiem.
  - c. Pēdējā izvēle izsauc funkciju SBX\_008.

Izvaddati

–

### 3.1.15. Centra modulis



Attēls 3.9 - centra moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.9. attēlota centra moduļa 2.līmeņa datu plūsmas diagramma ar 8 funkcijām, tālāk sauktas par LOCC\_(funkcijas skaitlis).

Tabula 3.41 - Tirgoties ar alķīmiķi

Identifikators: LOCC_001
Mērķis
Funkcijas mērķis ir veikt darbības ar alķīmiķi.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek parādīts atbilstošais teksts.</li> <li>2. Spēlētājam tiek dota izvēle starp darbībām: <ol style="list-style-type: none"> <li>a. Ja ir sasniegts noteikts posms kādā uzdevumā, tiek dota opcija, kas saistīta ar doto uzdevumu – tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>b. Ja tiek izvēlēta opcija pirkt inventāru, tiek parādīts skats ar pieejamajām precēm, kuras spēlētājs var iegādāties.</li> <li>c. Ja tiek izvēlēta opcija pārdot laupījumus, tiek palielināts spēlētāja naudas daudzums.</li> <li>d. Pēdējā izvēle izsauc funkciju SBX_002.</li> </ol> </li> </ol>
Izvaddati
–

Tabula 3.42 - Tirgoties ar bruņotāju

Identifikators: LOCC_002
Mērķis
Funkcijas mērķis ir veikt darbības ar bruņotāju.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Tiek parādīts atbilstošais teksts. Ja spēlētāja netīrības atribūts ir pārāk liels, tiek izsaukta funkcija SBX_002. Citādi:</li> <li>2. Spēlētājam tiek dota izvēle starp darbībām: <ol style="list-style-type: none"> <li>a. Ja ir sasniegts noteikts posms kādā uzdevumā, tiek dota opcija, kas saistīta ar doto uzdevumu – tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>b. Ja tiek izvēlēta opcija pirkt inventāru, tiek parādīts skats ar pieejamajām precēm, kuras spēlētājs var iegādāties.</li> <li>c. Pēdējā izvēle izsauc funkciju SBX_002.</li> </ol> </li> </ol>
Izvaddati
–

*Tabula 3.43 - Apmeklēt saunu*

Identifikators: LOCC_003
Mērķis
Funkcijas mērķis ir veikt darbības saunā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek parādīts atbilstošais teksts.</li> <li>2. Spēlētājam tiek dota izvēle starp darbībām: <ol style="list-style-type: none"> <li>a. Ja tiek izvēlēta parasta mazgāšanās un spēlētājam pietiek naudas, tiek izsaukta atbilstošā funkcija TXT_001 un atiestatīta spēlētāja netīrība.</li> <li>b. Ja tiek izvēlēta mazgāšanās ar masāžu un spēlētājam pietiek naudas, tiek izsaukta atbilstošā funkcija TXT_004 un atiestatīta spēlētāja netīrība.</li> <li>c. Pēdējā izvēle izsauc funkciju SBX_002.</li> </ol> </li> </ol>
Izvaddati
–

Tabula 3.44 - Apmeklēt festivālu

Identifikators: LOCC_004
Mērķis
Funkcijas mērķis ir veikt darbības festivālā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Ja spēlētājam pietiek naudas: <ol style="list-style-type: none"> <li>a. Ja ir sasniegts noteikts posms kādā uzdevumā, tiek dota opcija, kas saistīta ar doto uzdevumu – tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>b. Citādi tiek dota izvēle starp vairākām meitenēm – atkarībā no izvēles tiek izsaukta atbilstošā teksta moduļa funkcija un palielināts apmeklēto meiteņu skaits par 1.</li> <li>c. Kad šis skaits sasniedz 6, tiek izsaukta atbilstošā funkcija TXT_001 un SBX_001.</li> </ol> </li> </ol>
Izvaddati
–

Tabula 3.45 - Apmeklēt baznīcu

Identifikators: LOCC_005
Mērķis
Funkcijas mērķis ir veikt darbības baznīcā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> </ol>
Apstrāde

<ol style="list-style-type: none"> <li>1. Tiek parādīts atbilstošais teksts.</li> <li>2. Spēlētājam tiek dota izvēle starp darbībām: <ol style="list-style-type: none"> <li>a. Ja tiek izvēlēta svētīšana un spēlētājam pietiek naudas, tiek izsaukta atbilstošā funkcija TXT_003 un palielināta spēlētāja maģijas aizsardzība.</li> <li>b. Ja tiek izvēlēta atzīšanās un spēlētājam pietiek naudas, tiek izsaukta atbilstošā funkcija TXT_003 un būtiski palielināta spēlētāja maģijas aizsardzība.</li> <li>c. Pēdējā izvēle izsauc funkciju SBX_002.</li> </ol> </li> </ol>
Izvaddati
–

*Tabula 3.46 - Apmeklēt pili*

Identifikators: LOCC_006
Mērķis
Funkcijas mērķis ir veikt darbības pilī.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek parādīts atbilstošais teksts.</li> <li>2. Spēlētājam tiek dota izvēle: <ol style="list-style-type: none"> <li>a. Ja spēlētājs izvēlas satikt baronu, tiek izsaukta atbilstošā funkcija TXT_001.</li> <li>b. Ja spēlētājs izvēlas trenēties, ja reputācija ir pietiekami augsta, tiek izsaukta atbilstošā funkcija TXT_004, kurā tiek palielināts spēka vai veiklības atribūts, ja spēlētājam pietiek naudas.</li> <li>c. Citādi tiek izsaukta cita atbilstošā funkcija TXT_001.</li> </ol> </li> <li>3. Tiek izsaukta funkcija SBX_001.</li> </ol>
Izvaddati
–

Tabula 3.47 - Apmeklēt mēru

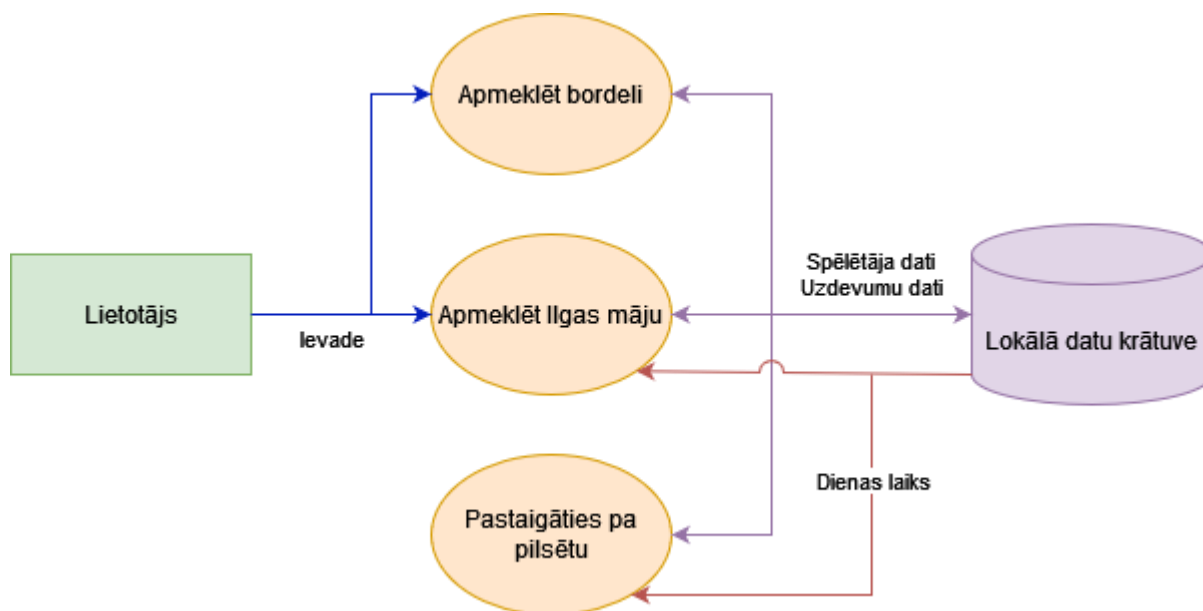
Identifikators: LOCC_007
Mērķis
Funkcijas mērķis ir veikt darbības ar mēru.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Atkarībā no uzdevumā sasniegtā posma, tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>2. Pēc iepriekš minēto uzdevumu izpildes tiek parādīts atbilstošais teksts un dota izvēle: saņemt uzdevumu vai pabeigt uzdevumu; tiek izsaukta atbilstošā funkcija TXT_004.</li> <li>3. Tiek izsaukta funkcija SBX_002.</li> </ol>
Izvaddati
–

Tabula 3.48 - Eksportēt spēli

Identifikators: LOCC_008
Mērķis
Funkcijas mērķis ir eksportēt spēli.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves.</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Mainīgie no spēlētāju datiem tiek kopēti jaunā mainīgo kopā.</li> <li>2. Tiek izveidots jauns fails, kas satur šos mainīgos. Failu var vēlāk izmantot spēles importēšanā 2.sērijā.</li> <li>3. Izvadīts paziņojums M013.</li> </ol>

Izvaddati
Saglabātas spēles fails.

### 3.1.16. Priekšpilsētas modulis



Attēls 3.10 - priekšpilsētas moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.10. attēlota priekšpilsētas moduļa 2.līmeņa datu plūsmas diagramma ar 3 funkcijām, tālāk sauktas par LOCK\_(funkcijas skaitlis).

Tabula 3.49 - Apmeklēt bordeli

Identifikators: LOCK_001
Mērķis
Funkcijas mērķis ir veikt darbības bordeļī.
Ievaddati
1. Spēlētāja ievade – izvēle 2. Spēlētāja dati no lokālās datu krātuves 3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves
Apstrāde

<ol style="list-style-type: none"> <li>1. Tiek parādīts atbilstošais teksts. Ja spēlētāja netīrības atribūts ir pārāk liels, tiek izsaukta funkcija SBX_002. Citādi:</li> <li>2. Spēlētājam tiek dota izvēle starp darbībām. <ol style="list-style-type: none"> <li>a. Ja ir sasniegts noteikts posms kādā uzdevumā, tiek dota opcija, kas saistīta ar doto uzdevumu – tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>b. Ja tiek izvēlēts pavadīt laiku ar kādu meiteni, tiek izsaukta atbilstošā funkcija TXT_004.</li> <li>c. Pēdējā opcija izsauc funkciju SBX_002.</li> </ol> </li> </ol>
Izvaddati
–

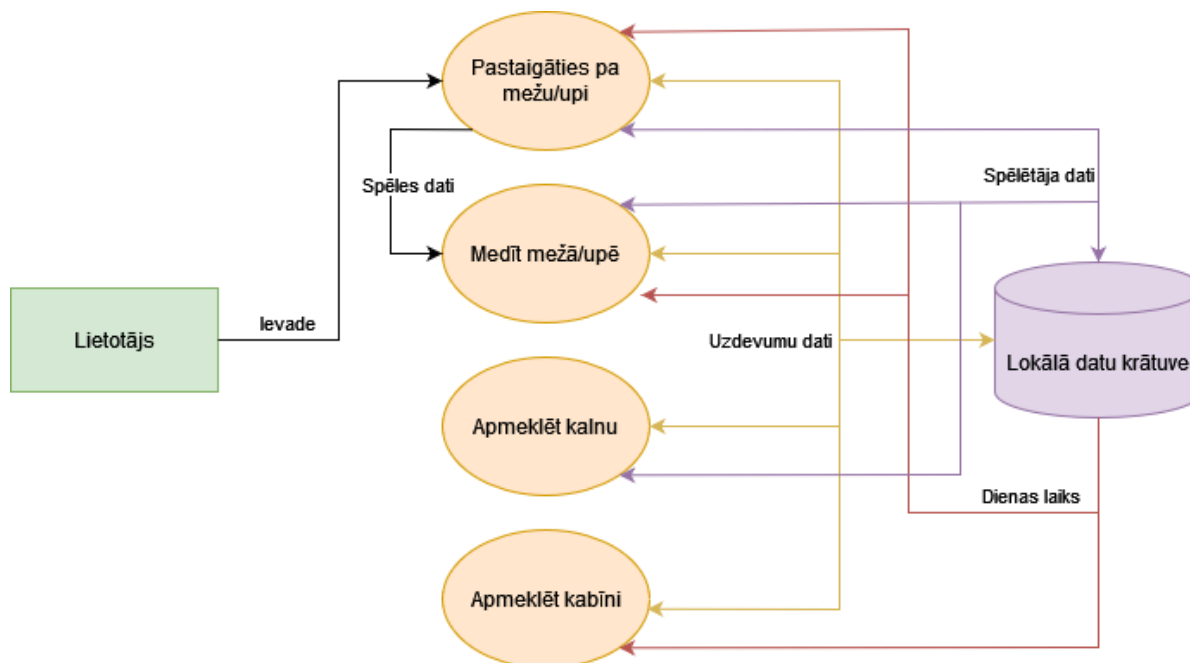
*Tabula 3.50 - Apmeklēt Ilgas māju*

Identifikators: LOCK_002
Mērķis
Funkcijas mērķis ir veikt darbības Ilgas mājā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja ievade – izvēle</li> <li>2. Spēlētāja dati no lokālās datu krātuves</li> <li>3. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> <li>4. Dienas laiks</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek parādīts atbilstošais teksts. Ja spēlētāja netīrības atribūts ir pārāk liels, tiek izsaukta funkcija SBX_002. Citādi:</li> <li>2. Pieejamās darbības ir atkarīgas no dienas un laika. <ol style="list-style-type: none"> <li>a. Ja ir rīts, tad atkarībā no izpildīto uzdevumu posma tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>b. Ja ir diena vai vakars, tad tiek izsaukta atbilstošā funkcija TXT_004.</li> <li>c. Ja ir nakts, tiek izsaukta funkcija SBX_002.</li> </ol> </li> </ol>
Izvaddati
–

Tabula 3.51 - Pastaigāties pa pilsētu

Identifikators: LOCK_003
Mērķis
Funkcijas mērķis ir veikt darbības, kas saistītas ar pastaigāšanos pa pilsētu.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> <li>3. Dienas laiks</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek nejauši aprēķināta mainīgā vērtība.</li> <li>2. Atkarībā no dienas laika un nejauši aprēķinātā mainīgā vērtības tiek izsaukta atbilstošā teksta moduļa funkcija TXT_001 – TXT_004.</li> <li>3. Ja diena ir sestdiena un laiks ir diena, tiek izsaukta funkcija LOCC_004. Citādi tiek izsaukta funkcija SBX_001.</li> </ol>
Izvaddati
–

### 3.1.17. Ārpilsētas modulis



Attēls 3.11 - ārpilsētas moduļa 2.līmeņa datu plūsmas diagramma

Attēlā 3.11. attēlota ārpilsētas moduļa 2.līmeņa datu plūsmas diagramma ar 4 funkcijām, tālāk sauktas par LOCO\_(funkcijas skaitlis).

Tabula 3.52 - Pastaigāties pa mežu/upi

Identifikators: LOCO_001
Mērķis
Funkcijas mērķis ir veikt darbības, kas saistītas ar pastaigāšanos ārpus pilsētas.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> <li>3. Dienas laiks</li> <li>4. Spēlētāja ievade – izvēle</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Ja tiek izvēlēts izpētīt ārpilsētu: <ol style="list-style-type: none"> <li>a. Tiek nejauši aprēķināta mainīgā vērtība.</li> <li>b. Atkarībā no šī mainīgā vērtības: <ol style="list-style-type: none"> <li>i. Vai nu tiek atklāta jauna vieta ārpilsētā, ja vēl nav atklāta;</li> <li>ii. Vai tiek izsaukta funkcija LOCO_002.</li> </ol> </li> </ol> </li> <li>2. Ja tiek izvēlēts peldēties upē: <ol style="list-style-type: none"> <li>a. Tiek nejauši aprēķināta mainīgā vērtība.</li> <li>b. Atkarībā no šī mainīgā vērtības un dienas laika var tikt izsaukta atbilstošā teksta moduļa funkcija.</li> <li>c. Tiek atkal nejauši aprēķināta mainīgā vērtība.</li> <li>d. Atkarībā no šī mainīgā vērtības vai nu tiks samazināta netīrība un palielināts dzīvības punktu skaits, vai izsaukta funkcija LOCO_002.</li> </ol> </li> </ol>
Izvaddati
–

Tabula 3.53 - Medīt mežā/upē

Identifikators: LOCO_002
Mērķis
Funkcijas mērķis ir veikt medības mežā/upē.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> <li>3. Dienas laiks</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Tiek nejauši aprēķināta mainīgā vērtība.</li> <li>2. Ja ir sasniegts noteikts posms kādā uzdevumā, tiek dota opcija, kas saistīta ar doto uzdevumu – tiek izsaukta atbilstošā teksta moduļa funkcija.</li> <li>3. Atkarībā no dienas laika un nejauši aprēķinātā mainīgā vērtības tiek izsaukta funkcija TXT_001, vai piešķirtas vērtības pretinieku mainīgajiem un izsaukta funkcija COM_001.</li> </ol>
Izvaddati
–

Tabula 3.54 - Apmeklēt kalnu

Identifikators: LOCO_003
Mērķis
Funkcijas mērķis ir veikt darbības kalnā.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Ja nav sasniegts nepieciešamais posms noteiktā uzdevumā, tiek izvadīts paziņojums M014 un izsaukta funkcija SBX_003.</li> <li>2. Citādi tiek piešķirta vērtība pretinieka mainīgajam un izsaukta funkcija COM_001, pēc kuras tiek izsauktas teksta moduļa funkcijas.</li> </ol>
Izvaddati
–

Tabula 3.55 - Apmeklēt kabīni

Identifikators: LOCO_004
Mērķis
Funkcijas mērķis ir veikt darbības kabīnē.
Ievaddati
<ol style="list-style-type: none"> <li>1. Uzdevumu saraksts un tiem atbilstošās posmu vērtības no lokālās datu krātuves</li> <li>2. Dienas laiks</li> </ol>
Apstrāde
<ol style="list-style-type: none"> <li>1. Atkarībā no dienas laika un uzdevumos sasniegtajiem posmiem tiek izsaukta atbilstošā funkcija TXT_003.</li> </ol>
Izvaddati
–

## **3.2. Nefunkcionālās prasības**

### **3.2.1. Veiktspējas prasības**

Operētājsistēma: Windows 7/10

Procesora jauda: 1 GHz

RAM: 1GB

Atmiņas ietilpība: 2GB<sup>[16]</sup>

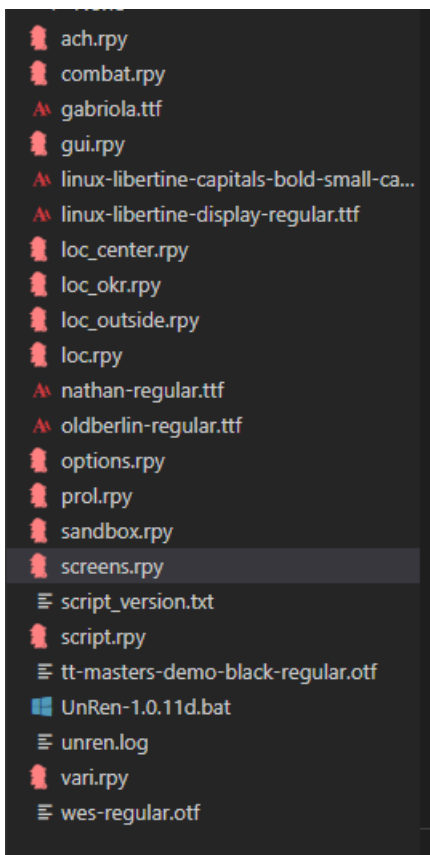
### **3.2.2. Pieejamības prasības**

Sistēma pieejama krievu un angļu valodās.

## 4. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS – KODA IZPĒTE

### 4.1. Lokālās datu krātuves struktūras projektējums

Visi spēles faili, kas nav Ren'Py dzinēja noklusējuma koda faili, tiek glabāti mapē game. Katrs fails ar .rpy paplašinājumu tiek uzskatīts kā spēles modulis, izņemot prol.rpy (kas sastāv tikai no teksta moduļa funkcijām) un vari.rpy, kas ir sistēmas lokālā datu krātuve.



Attēls 4.1 - spēles failu struktūra

Spēles mainīgie tiek glabāti failā vari.rpy; no šiem mainīgajiem ir atkarīgas visu funkciju darbības. Tālākajās tabulās tiks aprakstīti visi failā minētie mainīgie ar to paskaidrojumiem.

Datu tips šeit ir tikai nosacīts – tā kā tiek izmantota Python valoda, mainīgie ir dinamiski rakstīti – tiem nav pie inicializēšanas noteikts tips. Datu tips tiek noteikts no iespējamajām vērtībām.

#### 4.1.1. Funkcionālie mainīgie

Nosaka dažādus spēles aspektus.

Tabula 4.1 – funkcionālo mainīgo apraksts

Nosaukums	Datu tips	Paskaidrojums
lost	bool	Nosaka, vai spēlētājs miris cīņā.
iltrue, il_help	bool	Vai palīdzēja Ilgai spēles prologā – ļoti būtisks, ietekmē visu Ilgas līniju; abi mainīgie identiski.
loc_dec	string	Lokāciju apraksts – parādās pie vietu apmeklēšanas.
theday	int	Nedēļas diena. (nedēļa sākas ar svētdienu)
weekd	string	Nedēļas diena – teksts.
daycount	int	Dienu skaits spēlē.
daytime	string	Dienas laiks – viens no četriem dienas posmiem (rīts, diena, vakars, nakts).
curr_loc	string	Pašreizējā lokācija – nosaka, kura karte tiks izsaukta SBX_002.
bath_chain	int	Cik reizes apmeklēta sauna.
menupage	int	Lapas numurs vairāklapu skatos.
bro_sex_count	int	Cik reizes apmeklētas meitenes bordelī.
kil_money	bool	Vai ir sākta apmaksa par kroga istabiņu.
lo_visit	bool	Vai ir apmeklēta Londa (masiere) saunā.
ag_tr_cur	int	Treniņu progress – veiklība.
ag_tr_max	int	Cik reizes jātrenējas, lai palielinātu veiklību.
st_tr_cur	int	Treniņu progress – spēks.
st_tr_max	int	Cik reizes jātrenējas, lai palielinātu spēku.

Nosaukums	Datu tips	Paskaidrojums
attack_random	int	Nejauši aprēķināts mainīgais – nosaka, vai uzbrukums būs iespējams.
hit_chance	int	Uzbrukuma iespēja.
hit_piercing	int	Uzbrukuma spēks.
hp_taken	int	Cik dzīvības punktu tiek atņemts pretiniekam.
clog	array	Cīņas vēsture; cta izsauc ierakstu veikšanu šajā masīvā.

#### 4.1.2. Attiecību mainīgie

Nosaka attiecības ar citiem tēliem – nepieciešami gan uzdevumiem, gan atbilstošā teksta parādīšanai.

Tabula 4.2 – attiecību mainīgo apraksts

Nosaukums	Datu tips	Paskaidrojums
il_like	int	Ilga (dziedniece)
ki_like	int	Kilena (bārmeņa sieva)
mr_like	int	Mrisja (viesmīle krogā)
lo_like	int	Londa (saunas masiere)
he_like	int	Helga/Sāra (barona brāļameita)
el_like	int	Elana (viena no Tīrajiem, Forlana kolēģe)
ie_like	int	Iereja (mūķene)
ka_like	int	Kagera (Ilgas māte)
ho_like	int	Grāmatu pārdevēja
bur_like	int	Pilsētas mērs

### 4.1.3. Vietu atrašanas karogi

Visi ir bool tipa mainīgie – vērtības var būt vai nu True, vai False.

Tabula 4.3 – vietu atrašanas karogu apraksts

Nosaukums	Paskaidrojums
seen_castle	Vai redzēta barona pils.
seen_apt	Vai redzēts alķīmiķis.
seen_arm	Vai redzēts bruņotājs.
seen_bro	Vai redzēts bordelis.
seen_loc1	Vai redzēta ārpilsēta.
seen_loc2	Vai redzēta kabīne.
seen_loc3	Vai atrasta upe.
seen_loc4	Vai atrasts kalns.

### 4.1.4. Uzdevumu mainīgie

Glabā uzdevumu vērtības – nosaka VIS\_004 skatu un izsaukto funkciju secību.

Tabula 4.4 – uzdevumu mainīgo apraksts

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
m_quest1	int	“Call of the Void” – visas spēles galvenā stāsta līnija.	Galvenais uzdevums
m_quest2	int	Pilsētas izmeklēšana.	Galvenais uzdevums

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
m_quest2a, m_quest2o, m_quest2b, m_quest2m, m_quest2t, m_quest2i	bool	Kuri pilsētas iemītnieki ir izmeklēti.	m_quest2 mainīgie
m_quest3	int	Pilsētas mēra uzdevums – pilsētas kriminālo problēmu risināšana.	Galvenais uzdevums
m_quest4	int	Vai tika nogalināts vīrietis pie mēra mājas.	1.sērijā tālāk neizmantots
m_quest5	int	Forlana laiks cietumā – saistīts ar m_quest3.	1.sērijā tālāk neizmantots
m_quest7	int	“Tīrie” – Elanas un Reidriha uzdevums.	Galvenais uzdevums
quest1	int	Ilgas līnija	–
quest2	int	Fionas līnija	–
quest3	int	Ilgas un Forlana sarunas	Slēpts uzdevums
quest4	int	Noziedznieka noķeršana	1.sērijā neizmantots – quest8 duplikāts
quest5	int	Kilenas līnija	–
quest6	int	Mrisjas līnija	–
quest7	int	Londas līnija	–
quest8	int	Noziedznieka noķeršana	Atkārtojams uzdevums

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
quest9	int	Alķīmiķa uzdevums – iegūt laupījumu	–
quest10	int	Bruņotāja uzdevums – iegūt laupījumu	–
quest11	int	–	Neimplementēts
quest12	int	–	Duplikāts quest25, neimplementēts
quest13	int	Zaģles uzdevums – tikšanās naktī	–
quest14	int	–	Neimplementēts
quest15	int	Upes nimfas (vampīra) uzdevums	–
quest16	int	Kageras līnija	–
quest17	int	Ielas meitas (Higrītas, barona māšas) uzdevums	–
quest18	int	Sāras līnija	–
quest19	int	Upes raganas (Errelajas) uzdevums	–
quest20	int	Sapņi ar Forlana māti	Slēpts uzdevums
quest21	int	Radnas līnija	–
quest22	int	Zaģles uzdevums – noķeršana	–
quest23	int	Meža slepkavas noķeršana	–

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
quest24	int	Dainas uzdevums (amuleta izgatavošana)	–
quest25	int	Trenēšanās ar cīņas meistaru pili	–

#### 4.1.5. Inventāra mainīgie

Objekti ar nosaukumu, daudzumu qty, aprakstu desc, un cenu cost. Izmantots SBX\_005 un LOCC\_001.

Tabula 4.5 – inventāra mainīgo apraksts

Nosaukums	Paskaidrojums
tr_zom	Zombija pirksts – cīņas trofeja
tr_gob	Goblinu ausis – cīņas trofeja
tr_pol	Nakts monstra acis – cīņas trofeja
q_rep	Ķirzakas āda – nepieciešams quest10 jeb bruņotāja uzdevumam
q_bol	Purva monstu žoklis – nepieciešams quest9 jeb alķīmiķa uzdevumam
q_bab	Ilgas vecmāmiņas dzira – lietojama, palielina pc.str
p_h1	Vāja dzīvības dzira – palielina pc.hp par 25
p_h2	Vidēja dzīvības dzira – palielina pc.hp par 50
p_h3	Spēcīga dzīvības dzira – palielina pc.hp par 75
p_r1	Vāja niknuma dzira – palielina pc.rp par 10
p_r2	Vidēja niknuma dzira – palielina pc.rp par 15
p_r3	Stipra niknuma dzira – palielina pc.rp par 20

#### 4.1.6. Tēlu mainīgie

Nosaka gan spēlētāja, gan pretinieku atribūtus. Tiek izmantots cīņas moduļa funkcijās sekojošajā formātā: pc.[atribūts] spēlētāja datiem, [pret.vārds].[atribūts] pretinieku datiem.

Tabula 4.6 – tēlu mainīgo apraksts

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
name	string	Vārds/nosaukums	–
namer	string	Vārds/nosaukums, akuzatīvā	Atšķiras tikai krieviski, angļiski identisks ar name
tohit	int	Uzbrukuma precizitāte	–
ac	int	Aizsardzības precizitāte	–
str	int	Spēks	–
dex	int	Veiklība	–
rp	int	Niknuma punkti	Izmanto tikai spēlētājam
rp1	int	Maksimālie niknuma punkti	Izmanto tikai spēlētājam
mp	int	Uzbrukuma punkti	Izmanto tikai pretiniekiem
mp1	int	Maksimālie uzbrukuma punkti	Izmanto tikai pretiniekiem
hp	int	Dzīvības punkti	–
hp1	int	Maksimālie dzīvības punkti	–
phd	int	Fiziskā aizsardzība	–
mad	int	Maģiskā aizsardzība	–

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
pha	int	Fiziskais uzbrukums	–
maa	int	Maģiskais uzbrukums	–
wash	int	Netīrība	Palielinās par 1 katru dienu, ja neapmeklē saunu
stun	int	Dulluma efekts	Neimplementēts
bleed	int	Asiņošanas efekts	Neimplementēts
loot	int	Laupījums	Spēlētājam: cik laupījums sakrāts  Pretiniekam: cik laupījumu spēlētājs iegūs pie pretinieka uzvaras
bab	int	Nauda	Spēlētājam: cik nauda sakrāta  Pretiniekam: cik naudu spēlētājs iegūs pie pretinieka uzvaras
exp	int	Pieredzes punkti	Spēlētājam: cik pieredzes punkti sakrāti  Pretiniekam: cik pieredzes punktu spēlētājs iegūs pie pretinieka uzvaras
lev	int	Līmenis	–
rep	int	Reputācija	–
sk1	string	Zobens uzlabojumi	Var būt 'rune' vai 'ruk', citādi tukšs
sk2	string	Zobens uzlabojumi	Var būt 'bal', citādi tukšs
pic	.webp attēls	Galvenais attēls	–

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
pic_attack	.webp attēls	Attēls, uzbrūkot	–
pic_hit	.webp attēls	Attēls, saņemot sitienu	–
pic_block	.webp attēls	Attēls, bloķējot uzbrukumu	–

#### 4.1.7. Karodziņi

Nosaka dažādu spēles aspektu statusu.

Tabula 4.7 – karodziņu apraksts

Nosaukums	Datu tips	Paskaidrojums
dop_var1	bool	Upes atrašana – identisks seen_loc3
dop_var2	int	Sapņi ar Kageru
dop_var3	int	Nosaka bruņu stiprumu – jo lielāka vērtība, jo stiprākas bruņas
dop_var4	bool	Vai spēlētājs bēga no cīņas
dop_var5	bool	Vai no cīņas tiks iegūta nauda/pieredze
dop_var6	int	pc.hp bonuss pēc saunas masāžas – pēc laika pazūd
dop_var7	int	pc.ac bonuss – iegūts baznīcā
dop_var8	bool	Kageras uzdevuma testēšana – spēlē citādi neizmantots
dop_var9	bool	Vai tiks izlaists nefunkcionālais teksts LOCM_002
dop_var10	bool	Automātiska cīņa (COM_002 izsaukšana)
dop_var11	int	Sapņi ar Elanu
dop_var12	int	Cik reizes pavadīts laiks ar Mrisju

Nosaukums	Datu tips	Paskaidrojums
dop_var13	bool	Vai redzēta baznīca
dop_var14	bool	Sapnis ar Ilgu
dop_var15	int	Cik reizes redzētas meitenes upē peldoties
dop_var16	bool	Sapnis ar Sāru
dop_var17	bool	Papildus aina ar Ilgu
dop_var18	int	Cik reizes redzētas meitenes pie upes
dop_var19	int	Skaita, cik dienas pagājušas, kopš redzēta ielas meita (quest17)
dop_var20	bool	Vai ir iespēja apmeklēt Arminu bordelī par brīvu
dop_var21	int	Sapņi ar Fionu
dop_var22	bool	Sapnis ar Kilenu
dop_var23	int	Sapņi ar Sāru
dop_var24	int	Cik reizes spēlētājs ir trenējies (cik reizes izsaukta LOCM_004)
dop_var25	bool	Prasīšana Ilgai par amuletu quest24
dop_var26	float	Eksportēšanas mainīgais – nelietots
dop_var27	int	pc.ac bonuss – iegūts baznīcā, lielāks par dop_var27
dop_var28	bool	Vai tiek parādītas augšējo stūru izvēlnes
dop_var29	int	Zobens asums – pēc katras cīņas zūd
dop_var30	int	Moralitāte/pieklājība – jo lielāka vērtība, jo sliktākas morāles

## 4.2. Funkciju projektējums un apraksts

### 4.2.1. Sasniegumu modulis

Tabula 4.8 - Projektējums funkcijai “Definēt sasniegumu nosaukumus un tekstu”

<b>ID: ACH_001</b>
Ievaddati: <ul style="list-style-type: none"><li>• Saraksts ar sasniegumu nosaukumiem – kodā definēts</li><li>• Saraksts ar sasniegumu aprakstiem – kodā definēts</li></ul>
Obligātie lauki: saraksts ar sasniegumu nosaukumiem, saraksts ar sasniegumu aprakstiem
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Tiek izveidots objektu masīvs JSON formātā, kas satur objektus, kuriem definēts nosaukums <i>title</i> un apraksts <i>text</i>. Viss masīva saturs ir definēts kodā. Katrai pieejamai sistēmas valodai ir atsevišķs objektu masīvs.</li><li>• Masīvs tiek saglabāts persistent mainīgajā <i>persistent.achievements_dict</i>, globālajā atmiņā.</li></ul>

Tabula 4.9 - Projektējums funkcijai “Iegūt sasniegumu”

<b>ID: ACH_002</b>
Ievaddati: <ul style="list-style-type: none"><li>• Sasnieguma nosaukums</li><li>• Saraksts ar sasniegumiem un to aprakstiem no ACH_001</li></ul>
Obligātie lauki: Sasnieguma nosaukums, saraksts ar sasniegumiem un to aprakstiem
Citas pārbaudes: <ul style="list-style-type: none"><li>• Ja jau eksistē mainīgais <i>persistent.[nosaukums]</i> ar vērtību <i>True</i> – funkcijas darbība netiek turpināta.</li></ul>

Apstrāde:

- Tiek definēts mainīgais *ach* un tam piešķirta vērtība no objektu masīva, kas tika definēts ACH\_001.
- Piešķirtas vērtības mainīgajiem *a\_title* un *a\_text* no *ach* atribūtiem.
- Izveidots mainīgais *persistent.[nosaukums]* un tam vērtība definēta kā *True* – tiek saglabāts globālajā atmiņā.
- Tiek izsaukts sasnieguma iegūšanas skats: lodziņš, kurā redzams *a\_title* un *a\_text*.

Tabula 4.10 - Projektējums funkcijai “Parādīt sasniegumus”

<b>ID: ACH_003</b>
Ievaddati: <ul style="list-style-type: none"><li>• Globāli saglabātais iegūto sasniegumu saraksts – saraksts ar <i>persistent.[sasn.nosaukums]</i> mainīgajiem, kuriem vērtība ir <i>True</i></li><li>• Saraksts ar sasniegumiem un to aprakstiem no ACH_001</li><li>• Globāli saglabātais iegūto atmiņu saraksts – saraksts ar <i>persistent.[atm.nosaukums]</i> mainīgajiem, kuriem vērtība ir <i>True</i></li></ul>
Obligātie lauki: Globāli saglabātais iegūto sasniegumu saraksts, saraksts ar sasniegumiem un to aprakstiem, globāli saglabātais iegūto sasniegumu saraksts
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>

Apstrāde:

- Gan sasniegumiem, gan atmiņām skati ir funkcionāli identiski:
- Skatā tiek parādīti sasniegumi/atmiņas.
- Sasniegumiem: ja mainīgais neeksistē, lodziņā tiek parādīts teksts, ka sasniegums ir iegūts. Citādi, ja mainīgais eksistē un tā vērtība ir *True*, tiek parādīts attiecīgā sasnieguma teksts un apraksts no saraksta no ACH\_001.
- Atmiņām: ja mainīgais neeksistē vai tā vērtība ir *False*, lodziņā ir attēls ar jautājuma zīmi. Citādi, ja mainīgais eksistē un tā vērtība ir *True*, tiek parādīts attēls no dotās atmiņas.
  - Attēlu noklikšķinot, tiek izsaukta attiecīgā funkcija TXT\_001 ar atbilstošo nosaukumu.

#### 4.2.2. Cīņas modulis

Tabula 4.11 - Projektējums funkcijai “Uzsākt cīņu”

<b>ID: COM_001</b>
Ievaddati: <ul style="list-style-type: none"><li>• Pretinieku saraksts (<i>enemy1, enemy2, enemy3</i>), iegūts no izsaucošās funkcijas</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Pretinieku dati no lokālās datu krātuves</li></ul>
Obligātie lauki: Pretinieks <i>enemy2</i> , spēlētāja dati, pretinieku dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>

Apstrāde:

- Tiek atiestatīti mainīgie *dop\_var4* un *dop\_var10*, laupījums *loot*, pieredzes punkti *exp*, un nauda *bab*.
- Tiek izveidots masīvs *cta* cīņas vēstures saglabāšanai, kas tiek aizpildīts ar “ ”.
- Tiek iestatīti pretinieku dati: katram pretiniekam, kas eksistē, tiek iestatīti dzīvības punkti *enemy[num].hp*, uzbrukuma punkti *enemy[num].mp*, uzbrukuma precizitāte *enemy[num].tohit* no formulas  $\text{spēks} * 0.66 + \text{veiklība} * 0.34$  (*str* un *dex* attiecīgi), un aizsardzības precizitāte *enemy[num].ac*.
- Tiek iestatīti spēlētāja dati: uzbrukuma precizitāte *pc.tohit* no iepriekš minētās formulas, aizsardzības precizitāte *pc.ac* (ar bonusu +1, ja *dop\_var7* vai *dop\_var27* nav 0), *pcatt* un *pctar* (objekti, kas lietoti cīņas funkcijās – satur spēlētāja datus no lokālās krātuves), un bonusu: precizitātes bonuss *pop* un uzbrukuma bonus *dam* (abi iestatīti, atkarībā no zobena uzlabojumiem *pc.sk1* un *pc.sk2*).
- Tiek izsaukta funkcija COM\_003.

Tabula 4.12 - Projektējums funkcijai “Automātiski atrisināt cīņu”

<b>ID: COM_002</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Pretinieku dati no lokālās datu krātuves</li><li>• Iestatītie mainīgie no COM_001</li></ul>
Obligātie lauki: spēlētāja dati, pretinieku dati, mainīgie no COM_001
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>

Apstrāde:

1. Pēc kārtas, sākot ar pirmo (katrā uzbrukuma reizē var uzbrukt tikai vienam pretiniekam), uzbrūk katram pretiniekam, kamēr tā vērtība *enemy[num]* nav “none”:
  - a. Atkarībā no atšķirības starp *enemy[num].ac* un *pc.tohit* tiek piešķirta *player\_action* uzbrukuma veids un iestatīts *player\_target* kā *enemy[num]*.
  - b. Ja *player\_action* ir viegls vai spēcīgs uzbrukums, tiek iestatītas atbilstošās vērtības uzbrukuma iespējamības *mod\_at*, uzbrukuma sitiena *mod\_dam* un uzbrukuma spēka *mod\_str* mainīgajiem un spēlētāja darbība definēta kā parasts uzbrukums.
  - c. Tiek izsaukta COM\_006.
2. Ja visu pretinieku vērtības *enemy[num]* ir “none”, tiek izsaukta funkcija COM\_005.
3. Tiek izsaukta funkcija COM\_004.
4. Funkcija darbību turpina, kamēr netiek izsaukta COM\_005.

Tabula 4.13 - Projektējums funkcijai “Uzbrukums (spēlētājs)”

ID: COM_003
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Pretinieku dati no lokālās datu krātuves</li><li>• Spēlētāja ievade</li><li>• Iestatītie mainīgie no COM_001</li></ul>
Obligātie lauki: spēlētāja dati, pretinieku dati, spēlētāja ievade, mainīgie no COM_001
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>

Apstrāde:

1. Ja visu pretinieku vērtības *enemy[num]* ir “none”, tiek izsaukta funkcija COM\_005.
2. Spēlētājs izvēlas nākamo darbību:
  - a. Ja darbība ir automātiska cīņa jeb *autocombat*, *dop\_var10* vērtība tiek iestatīta kā 1 un tiek izsaukta funkcija COM\_002.
  - b. Ja tiek izvēlēts spēcīgs uzbrukums, viegls uzbrukums vai parasts uzbrukums, tiek iestatītas atbilstošās vērtības uzbrukuma iespējamības *mod\_at*, uzbrukuma sitiena *mod\_dam* un uzbrukuma spēka *mod\_str* mainīgajiem un spēlētāja darbība definēta kā parasts uzbrukums.
  - c. Ja tiek izvēlēta pretinieka izpētīšana:
    - i. Spēlētājs izvēlas, kuru pretinieku pētīs: tiek iestatīts *player\_target*.
    - ii. Atkarībā no atšķirības starp *pc.str* un *player\_target.str*, masīvā *cta* tiek veikts attiecīgais ieraksts. Tas pats ar *pc.dex* un *player\_target.dex*.
    - iii. Spēlētāja darbība tiek iestatīta kā atvairīšana jeb *parry*.
  - d. Ja tiek izvēlēts apskatīt inventāru:
    - i. Tiek paslēpti cīņas skati.
    - ii. Tiek izsaukta saīsināta SBX\_005.
    - iii. Spēlētāja darbība tiek iestatīta kā atvairīšana jeb *parry*.
  - e. Ja tiek veikta atvairīšana, tiek iestatīta *mod\_parry* (karodziņš, vai tiek atvairīts uzbrukums) vērtība uz 1 un masīvā *cta* tiek veikts attiecīgais ieraksts.
  - f. Ja tiek veikta bēgšana, tiek veikts attiecīgais ieraksts masīvā *cta* un:
    - i. Tiek aprēķināts nejaušs mainīgais ar *rand(1,3)*.
    - ii. Ja mainīgais ir lielāks par 1, tiek paslēpti cīņas skati un *dop\_var4* vērtība iestatīta kā 1.
    - iii. Citādi tiek veikts attiecīgais ieraksts masīvā *cta*.
  - g. Ja netiek veikta atvairīšana, spēlētājs izvēlas mērķi – tiek iestatīts *player\_target*, piešķirta attiecīgā vērtība mainīgajam *at\_target*, tiek veikts attiecīgais ieraksts masīvā *cta* un izsaukta funkcija COM\_006.
3. Tiek izsaukta funkcija COM\_004.

Tabula 4.14 - Projektējums funkcijai "Uzbrukums (pretinieks)"

<b>ID: COM_004</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Pretinieku dati no lokālās datu krātuves</li> <li>• Iestatītie mainīgie no COM_001</li> </ul>
Obligātie lauki: spēlētāja dati, pretinieku dati, mainīgie no COM_001
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ol style="list-style-type: none"> <li>1. Katram pretiniekam, kuram vērtība <i>enemy[num]</i> nav "none", tiek iestatīta <i>current_attaker</i> kā pašreizējais pretinieks un tiek izsaukts uzbrukums: <ol style="list-style-type: none"> <li>a. Tiek veikts attiecīgais ieraksts masīvā <i>cta</i> un, ja <i>dop_var10</i> nav 1, parādīts atbilstošais skats.</li> <li>b. Ja no spēlētāja tiek veikta aizsargāšanās jeb <i>player_action</i> ir "parry", tiek izsaukta funkcija COM_007, un: <ol style="list-style-type: none"> <li>i. Ja <i>dop_var5</i> vērtība ir 1, mainīgā vērtība tiek atiestatīta uz 0.</li> <li>ii. Tiek izsaukta funkcija COM_005.</li> </ol> </li> <li>c. Tiek aprēķināta uzbrukuma iespējamība <i>attack_random</i> no formulas <math>\text{rand}(-3, 3) + \text{rand}(-3, 3)</math>.</li> <li>d. Tiek aprēķināta uzbrukuma iespēja no formulas <math>\text{current\_attaker.tohit} - \text{pc.ac} + \text{attack\_random}</math>.</li> <li>e. Ja <i>attack_random</i> ir mazāka/vienāda ar -5 (6% iespēja) vai <i>hit_chance</i> ir mazāka par 0, tiek izsaukts netrāpīšanas skats, veikts ieraksts masīvā <i>cta</i> un izvadīts attiecīgais teksts.</li> <li>f. Tiek aprēķināts <i>hit_piercing</i> atkarībā no tā, vai <i>current_attaker.pha</i> vai <i>current_attaker.maa</i> (maģiskais uzbrukums) ir lielāks: ja <i>pha</i>, tad formula <math>\text{current\_attaker.pha} - \text{pc.phd}</math> (spēlētāja fiziskā aizsardzība) + <math>\text{rand}(-3, 3)</math>; ja <i>maa</i>, tad <math>\text{current\_attaker.maa} - \text{pc.mad}</math> (spēlētāja maģiskā aizsardzība) + <math>\text{rand}(-3, 3)</math>.</li> <li>g. Tiek aprēķināts atņemtais dzīvības punktu skaits <i>hp_taken</i> no formulas</li> </ol> </li> </ol>

*hit\_piercing + current\_attaker.str.*

- h. Ja *hit\_piercing* ir mazāks/vienāds ar 0, tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā *cta*.
  - i. Ja *attack\_random* = 6 (maksimālā vērtība, 2% iespēja), *hp\_taken* tiek reizināts ar 1,5, tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā *cta*.
  - j. Citādi tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā *cta*.
  - k. Tiek samazināts *pc.hp* (spēlētāja dzīvības punkti) par *hp\_taken*.
  - l. Tiek atiestatīts *mod\_parry* (atvairīšanas mainīgais).
2. Tiek veikta pārbaude T3 – ja patiesa, tad tiek izsaukta funkcija COM\_005 ar *lost = True* vērtību.
  3. Atgriežas funkcijā, kas šo funkciju izsauca.

Tabula 4.15 - Projektējums funkcijai “Cīņas beigas”

ID: COM_005
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Iestatītie mainīgie no COM_001, kas rediģēti funkcijās COM_003/COM_004.</li></ul>
Obligātie lauki: spēlētāja dati, mainīgie no COM_001
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Ja <i>lost</i> ir <i>True</i> no funkcijas izsaukuma no COM_004, tiek izvadīts paziņojums M003, tiek izsaukta funkcija VIS_001 un šīs funkcijas darbība pārtraukta.</li><li>• Ja <i>dop_var29</i> vērtība ir lielāka par nulli, tā vērtība tiek samazināta par 1. Ja <i>dop_var29</i> vērtība pēc samazināšanas ir 0, tad izvadīts paziņojums M004.</li><li>• Ja <i>dop_var5</i> vērtība ir 1, tā tiek atiestatīta uz 0 un funkcijas darbība pārtraukta.</li><li>• Citādi, ja turpmāk minētie mainīgie nav 0, tiek mainīts <i>exp</i> uz <i>exp + rand(-20, 20)</i> un <i>loot</i> uz <i>loot + rand(-5,5)</i>, un attiecīgi <i>pc.exp</i> (spēlētāja pieredzes punkti), <i>pc.loot</i> (spēlētāja laupījums) un <i>pc.bab</i> (spēlētāja nauda) tiek pieskaitīti <i>exp</i>, <i>loot</i> un <i>bab</i>, kā arī tiek izvadīts teksts par saņemto atalgojumu.</li></ul>

Tabula 4.16 - Projektējums funkcijai "Uzbrukt pretiniekam"

<b>ID: COM_006</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Pretinieku dati no lokālās datu krātuves</li> <li>• Uzbrukuma veids – no COM_003</li> <li>• Pretinieks <i>at_target</i> – no COM_003</li> <li>• Iestatītie mainīgie no COM_001</li> </ul>
<p>Obligātie lauki: spēlētāja dati, pretinieka dati, uzbrukuma veids, pretinieks, mainīgie no COM_001</p>
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ol style="list-style-type: none"> <li>1. Ja uzbrukuma veids ir parasts uzbrukums: <ol style="list-style-type: none"> <li>a. Tiek aprēķināta uzbrukuma iespējamība <i>attack_random</i> no formulas <math>\text{rand}(-3, 3) + \text{rand}(-3,3)</math>.</li> <li>b. Tiek aprēķināta uzbrukuma iespēja <i>hit_chance</i> no formulas <math>pc.tohit - at\_target.ac (at\_target - \text{pretinieks}) + attack\_random + mod\_at</math>.</li> <li>c. Ja <i>attack_random</i> ir mazāka/vienāda ar -5 (6% iespēja) vai <i>hit_chance</i> ir mazāka par 0, tiek izsaukts netrāpīšanas skats (ja <i>dop_var10</i> nav 1), veikts ieraksts masīvā <i>cta</i> un izvadīts attiecīgais teksts.</li> <li>d. Citādi tiek aprēķināts uzbrukuma spēks <i>hit_piercing</i> no formulas <math>pc.pha</math> (spēlētāja fiziskais uzbrukums) – <math>at\_target.phd</math> (pretinieka fiziskā aizsardzība) + <math>\text{rand}(-3, 3) + mod\_dam</math>.</li> <li>e. Ja <i>hit_piercing</i> ir mazāks/vienāds ar 0, tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā <i>cta</i>.</li> <li>f. Citādi tiek aprēķināta kritiska uzbrukuma iespējamība no <math>\text{rand}(1, 10)</math>.</li> <li>g. Tiek aprēķināts atņemtais dzīvības punktu skaits <i>hp_taken</i> no formulas <math>hit\_piercing + pc.str</math> (spēlētāja spēks) + <math>mod\_str + dam + dop\_var29</math>.</li> <li>h. Ja kritiskā uzbrukuma iespējamības vērtība ir lielāka/vienāda ar 9 (20% iespēja), <i>hp_taken</i> tiek reizināts ar 1,75, tiek izvadīts attiecīgais teksts un</li> </ol> </li> </ol>

<p>veikts ieraksts masīvā <i>cta</i>.</p> <p>i. Citādi tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā <i>cta</i>. Ja <i>mod_parry</i> (karodziņš, vai tiek atvairīts uzbrukums) ir 1, <i>pc.rp</i> (niknuma punkti) tiek palielināts par <i>hp_taken/3</i>. Citādi <i>pc.rp</i> tiek palielināts par <i>hp_taken/2</i>.</p> <p>j. Tiek samazināts <i>at_target.hp</i> (pretinieka dzīvības punkti) par <i>hp_taken</i>.</p> <p>k. Tiek izsauktas pārbaudes T1 un T2 – ja patiesas, attiecīgi tiek atiestatīti <i>pc.rp</i> uz maksimālo iespējamo vērtību <i>pc.rp1</i>; pretinieka <i>enemy[num]</i> vērtība iestatīta uz “none” un <i>loot</i>, <i>exp</i> un <i>bab</i> tiek pieskaitītas attiecīgās vērtības.</p> <p>2. Ja uzbrukuma veids ir niknuma uzbrukums, notiek tas pats, kas 1.punktā, bet:</p> <p>a. 1.d solī <i>hit_piercing</i> tiek aprēķināts no formulas <math>pc.pha + \text{rand}(-3, 3) + 2</math>.</p> <p>b. 1.g solī <i>hp_taken</i> tiek aprēķināts no formulas <math>(hit_piercing + pc.str + dop\_var29)*2 + dam*1.5</math>.</p> <p>c. 1.h solis tiek izpildīts, ja uzbrukuma iespējamības vērtība ir lielāka/vienāda ar 8, un <i>hp_taken</i> reizināts ar 1,5.</p> <p>d. 1.i solis tiek ignorēts.</p>
--

Tabula 4.17 - Projektējums funkcijai “Atvairīt pretinieka uzbrukumu”

<b>ID: COM_007</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Iestatīts mainīgais <i>mod_parry</i> no COM_003</li> <li>• Pretinieks <i>current_attaker</i> no COM_004</li> </ul>
Obligātie lauki: <i>mod_parry</i> , <i>current_attaker</i>
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Tiek aprēķināts nejaušs mainīgais ar <math>\text{rand}(1,3)</math>.</li> <li>• Ja mainīgā vērtība ir 1, atgriežas funkcijā COM_004.</li> <li>• Ja mainīgā vērtība ir 2, tiek izvadīts attiecīgais teksts un masīvā <i>cta</i> veikts ieraksts.</li> <li>• Ja mainīgā vērtība ir 3, tiek izvadīts attiecīgais teksts, masīvā <i>cta</i> veikts ieraksts, iestatīts <math>at\_target = current\_attaker</math> un izsaukta funkcija COM_003.</li> </ul>

### 4.2.3. Smilšukastes modulis

Tabula 4.18 - Projektējums funkcijai "Tīt laiku uz priekšu"

<b>ID: SBX_001</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Diena <i>theday</i>, dienu skaits <i>daycount</i> un laiks <i>daytime</i></li></ul>
Obligātie lauki: spēlētāja dati, diena, dienu skaits, laiks
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ol style="list-style-type: none"><li>1. Ja <i>pc.bab</i> (spēlētāja nauda) ir lielāka/vienāda par 10000 vienībām, tiek izvadīts paziņojums M015 un <i>pc.bab</i> atiestatīts uz 200 vienībām.</li><li>2. Ja <i>lost</i> (spēlētāja dzīvības statuss) ir <i>True</i>, tiek izsaukta funkcija SBX_007.</li><li>3. Atkarībā no <i>daytime</i> vērtības tiek mainīta <i>daytime</i> vērtība:<ol style="list-style-type: none"><li>a. Ja <i>daytime</i> ir '<i>morning</i>' (rīts), <i>daytime</i> tiek mainīts uz '<i>day</i>' (diena) un atkarībā no <i>quest18</i> (Sāras uzdevuma) vērtības tiek izsaukta atbilstošā funkcija TXT_003.</li><li>b. Ja <i>daytime</i> ir '<i>day</i>' (diena), <i>daytime</i> tiek mainīts uz '<i>evening</i>' (vakars).</li><li>c. Ja <i>daytime</i> ir '<i>evening</i>' (vakars), <i>daytime</i> tiek mainīts uz '<i>night</i>' (nakts) un atkarībā no <i>quest1</i> (Ilgas uzdevuma) vērtības un <i>theday</i> vērtības (ja <i>theday</i> ir 7, tas ir, festivāla diena) tiek izsaukta funkcija SBX_009.</li><li>d. Ja <i>daytime</i> ir '<i>night</i>' (nakts), <i>daytime</i> tiek mainīts uz '<i>morning</i>' (rīts), un:<ol style="list-style-type: none"><li>i. Palielināta spēlētāja netīrība <i>pc.wash</i>.</li><li>ii. Atkarībā no noteiktu uzdevumu vērtībām tiek vai nu izsaukta funkcija SBX_009, vai izsaukta funkcija TXT_003 vai TXT_004.</li><li>iii. Tiek pieskaitīts 1 <i>theday</i> un <i>daycount</i> vērtībām.</li><li>iv. Atkarībā no <i>theday</i> vērtības tiek iestatīta atbilstošā <i>weekd</i> (nedēļas dienas teksta) vērtība. Ja <i>theday</i> ir 8 (nedēļa pabeigta), <i>theday</i> tiek</li></ol></li></ol></li></ol>

atiestatīts uz 1.

v. Ja *dop\_var6*, *dop\_var7* vai *dop\_var27* vērtības ir lielākas par 0, to vērtība tiek samazināta par 1. Ja jebkura no šīm vērtībām sasniedz 0, to dotais bonuss (attiecīgi *pc.hp* un *pc.hp1* priekš *dop\_var6* un *pc.mad* pārējiem) tiek dzēsts.

4. Tiek atiestatīti niknuma punkti *pc.rp*.
5. Ja *pc.exp* (piederzes punkti) ir lielāks/vienāds ar 1000, tiek izsaukta funkcija SBX\_006.
6. Ja spēlētāja pašreizējā lokācija *curr\_loc* ir istaba, tiek izsaukta funkcija SBX\_008. Citādi izsauc SBX\_002.

Tabula 4.19 - Projektējums funkcijai “Iet uz noteiktu vietu virspasaulē”

<b>ID: SBX_002</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja pašreizējā lokācija <i>curr_loc</i></li><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li></ul>
Obligātie lauki: <i>curr_loc</i> , spēlētāja ievade
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Tiek izsaukta funkcija SBX_006.</li><li>• Ja <i>curr_loc</i> vērtība ir “<i>centr</i>”: ja <i>theday</i> ir 7 un <i>daytime</i> ir ‘<i>evening</i>’, tiek izsaukta funkcija LOCM_004. Citādi tiek izsaukta centra kartes izvēlne – atkarībā no spēlētāja izvēles tiek izsaukta viena no sekojošajām funkcijām: LOCC_001, LOCC_002 (ja dienas laiks nav nakts), LOCC_003, LOCC_005, LOCC_007 (ja dienas laiks nav vakars vai nakts), LOCK_003, TXT_003 funkcija <i>q1_14</i> (ja Ilgas uzdevumā <i>quest1</i> ir noteikta vērtība un diena nav sestdiena).</li><li>• Ja <i>curr_loc</i> vērtība ir “<i>okraina</i>”, tiek izsaukta priekšpilsētas kartes izvēlne – atkarībā no spēlētāja izvēles tiek izsaukta viena no sekojošajām funkcijām:</li></ul>

LOCK\_001, LOCK\_002, TXT\_004 funkcija *q24\_7* (ja uzdevumā *quest24* ir noteikta vērtība).

- Ja *curr\_loc* vērtība ir “*outside*”, tiek izsaukta funkcija SBX\_003.
- Ja *curr\_loc* vērtība ir “*room*”, tiek izsaukta funkcija SBX\_008.
  - Visās iepriekšminētajās kartēs ir izvēle, kuru izsaucot, tiek atiestatīts *curr\_loc* un atkal izsaukts SBX\_002
- Citādi, ja *curr\_loc* ir atiestatīts, tiek izsaukta pilsētas kartes izvēlne – atkarībā no spēlētāja izvēles tiek izsaukta viena no sekojošajām funkcijām: SBX\_008, LOCC\_006, LOCM\_005, SBX\_002 ar *curr\_loc* iestatītu kā “*centr*”, “*okraina*” vai “*outside*”.
- Visām iepriekš minētajām izvēlnēm ir izvēle izsaukt SBX\_001.

Tabula 4.20 - Projektējums funkcijai “Ārpilsētas izvēlne”

ID: SBX_003
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Spēlētāja ievade</li></ul>
Obligātie lauki: spēlētāja dati, spēlētāja ievade
Citas pārbaudes: <ul style="list-style-type: none"><li>• Ja <i>lost</i> vērtība ir <i>True</i>, tiek izsaukta funkcija SBX_007 un šīs funkcijas darbība pārtraukta.</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Tiek aprēķināts nejaušs mainīgais no <i>rand(1, 10)</i>. Ja ir diena un šis mainīgais ir 1, tiek izsaukta funkcija TXT_005, kur pretinieks <i>enemy2</i> ir <i>bear</i> (jeb lācis).</li><li>• Ja nav iepriekš redzēta ārpilsēta, tas ir, <i>seen_loc1</i> ir <i>False</i>, tā vērtību nomaina uz <i>True</i> un izvada atbilstošo tekstu.</li><li>• Tiek izsaukta ārpilsētas kartes izvēlne – atkarībā no spēlētāja izvēles tiek izsaukta funkcija LOCO_001, LOCO_002, LOCO_003 (ja <i>seen_loc4</i> ir <i>True</i>), LOCO_004 (ja uzdevumam <i>quest21</i> vai <i>quest16</i> ir noteikta vērtība), vai SBX_001.</li></ul>

Tabula 4.21 - Projektējums funkcijai "Parādīt profilu"

<b>ID: SBX_004</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ol style="list-style-type: none"> <li>1. Tiek parādīts profila skats ar galvenā tēla attēlu un sekojošajiem datiem: <ol style="list-style-type: none"> <li>a. <i>pc.hp</i> un <i>pc.hp1</i>, <i>pc.rp</i> un <i>pc.rp1</i>, <i>pc.lev</i>, <i>pc.exp</i>, <i>pc.str</i>, <i>pc.pha</i>, <i>pc.phd</i>, <i>pc.maa</i>, <i>pc.mad</i>, <i>pc.rep</i>.</li> <li>b. Atkarībā no mainīgā <i>dop_var3</i> tiek parādīts atbilstošais teksts bruņu stiprumam.</li> <li>c. Atkarībā no mainīgā <i>dop_var29</i> tiek parādīts atbilstošais teksts zobens asuma statusam.</li> <li>d. Atkarībā no <i>pc.sk1</i> un <i>pc.sk2</i> tiek parādīts atbilstošais zobens uzlabojums.</li> </ol> </li> </ol>

Tabula 4.22 - Projektējums funkcijai "Cīņas inventārs"

<b>ID: SBX_005</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Inventāra dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: inventāra dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>

Apstrāde:

- Katram iespējamajam inventāra objektam, kas definēts kodā pie spēles uzsākšanas: ja tā *qty* ir lielāks par 0, tas skatā tiek parādīts formātā “[*inv*].*name*, [*inv*].*qty*, [*inv*].*desc*”.
- Noteiktām inventāra lietām, tas ir, dzīvības dzirām *p\_h1*, *p\_h2*, *p\_h3*, niknuma dzirām *p\_r1*, *p\_r2*, *p\_r3* un dzirai *p\_bab* ir iespēja tās lietot. Uzklīkšķinot, tiek izsaukta atbilstošā funkcija [*inv*]*\_use*, kas nomaina spēlētāja atribūtus par noteiktu daudzumu – dzīvības dziras palielina *pc.hp*, niknuma dziras *pc.rp*, un dzira *p\_bab* palielina *pc.str*, samazinot dziras daudzumu par 1.
- Ja SBX\_005 tiek izsaukts cīņas laikā, tiek parādītas tikai inventāra lietas minētas 2.punktā.

Tabula 4.23 - Projektējums funkcijai “Palielināt līmeni”

ID: SBX_006
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Pretinieku dati no lokālās datu krātuves</li><li>• Spēlētāja izvēle</li></ul>
Obligātie lauki: spēlētāja dati, pretinieku dati, spēlētāja izvēle
Citas pārbaudes: <ul style="list-style-type: none"><li>• <i>pc.exp</i> jābūt lielākam par 1000.</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Tiek samazināts <i>pc.exp</i> par 1000.</li><li>• Palielinās <i>pc.hp</i> par 10, <i>pc.hp1</i> par 10, <i>pc.rp</i> par 5, <i>pc.lev</i> par 1.</li><li>• Ja <i>pc.lev</i> ir 4, palielinās noteiktu pretinieku <i>hp1</i>, <i>str</i>, <i>dex</i>, <i>loot</i>, <i>exp</i> par 20, 2, 2, 20, 30 attiecīgi, un pretinieku līmenis <i>lev</i> iestatīts kā 4.</li><li>• Tiek izsaukts saīsināts SBX_004 skats, ar papildus opcijām palielināt vai nu <i>pc.str</i>, vai <i>pc.dex</i>. Atkarībā no izvēles noteiktais atribūts tiek palielināts par 1.</li></ul>

Tabula 4.24 - Projektējums funkcijai “Spēles beigas”

<b>ID: SBX_007</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves (<i>lost</i> – spēlētāja nāves karogs)</li> </ul>
Obligātie lauki: <i>lost</i>
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Ja mainīgā <i>lost</i> vērtība ir <i>True</i> – spēlētājs zaudējis cīņā –, tiek izvadīts paziņojums M003.</li> <li>• Tiek izsaukts spēles beigu skats.</li> <li>• Tiek izsaukta funkcija VIS_001.</li> </ul>

Tabula 4.25 - Projektējums funkcijai “Istabas izvēlne”

<b>ID: SBX_008</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja ievade</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja ievade, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Tiek parādīta istabas izvēlne – atkarībā no spēlētāja izvēles var izsaukt vienu no sekojošajām funkcijām: LOCM_005, LOCM_003, LOCM_004 ar parametru <i>str</i> vai <i>dex</i> pēc izvēles, SBX_002 ar <i>curr_loc</i> vērtību “<i>centr</i>”, vai atbilstošā teksta moduļa funkcija, balstoties uz uzdevuma <i>quest1</i> un <i>m_quest7</i> vērtībām.</li> </ul>

Tabula 4.26 - Projektējums funkcijai “Mainīt uzdevuma vērtību”

<b>ID: SBX_009</b>
--------------------

Ievaddati: <ul style="list-style-type: none"> <li>• Uzdevuma nosaukums</li> <li>• Posma vērtība</li> </ul>
Obligātie lauki: uzdevuma nosaukums, posma vērtība
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Uzdevumam ar doto nosaukumu tiek piešķirta dotā vērtība – piemēram, <math>quest0 = 1</math>.</li> <li>• Tiek parādīts un pēc sekundes paslēpts uzdevuma iegūšanas vizuālais paziņojums.</li> <li>• Tiek atskaņota skaņa.</li> </ul>

#### 4.2.4. Skatu modulis

Tabula 4.27 - Projektējums funkcijai “Parādīt izvēlnes”

<b>ID: VIS_001</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja ievade</li> </ul>
Obligātie lauki: spēlētāja ievade
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Tiek parādīts galvenās izvēlnes skats.</li> <li>• Atkarībā no spēlētāja izvēles var tikt izsaukta viena no sekojošajām funkcijām – SCR_002, VIS_003, ACH_003, atvērtas hipersaites lietotāja noklusējuma pārlūkprogrammā, vai atvērts iestatījumu skats, kas ir Ren’Py noklusējuma kods.<sup>[14]</sup></li> <li>• Ja spēlētājs izvēlas aizvērt spēles logu, tiek izvadīts paziņojums M007, un, ja atbilde ir apstiprinoša, spēles logs tiek aizvērts.</li> </ul>

Tabula 4.28 - Projektējums funkcijai “Parādīt teksta vēsturi”

<b>ID: VIS_002</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Stāsta teksta vēsture <code>_history_list</code> no Ren'Py dzinēja failiem</li> </ul>
Obligātie lauki: <code>_history_list</code>
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Katrs ieraksts sarakstā <code>_history_list</code> tiek parādīts skatā, apgrieztā secībā – tā, lai jaunākie ieraksti būtu loga apakšā.</li> <li>• Ja <code>_history_list</code> ir tukšs, tiek izvadīts paziņojums M008.</li> </ul>

Tabula 4.29 - Projektējums funkcijai “Pārvaldīt saglabāto spēļu sarakstu”

<b>ID: VIS_003</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Saglabāto spēļu saraksts – faili no mapes <code>game/saves</code></li> </ul>
Obligātie lauki: saglabāto spēļu saraksts
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>

Apstrāde:

- Katrs eksistējošais fails tiek parādīts savā lodziņā un savā lapā, balsoties uz faila nosakumu. Tā standarti ir definēti Ren'Py dokumentācijā. Piemēram, fails 1-5-LT1.save tiks parādīts 1.lapā, 5.lodziņā.
- Ja, turot kursoru uz noteikta faila, tiek piespiests “Delete” taustiņš, tad tiek izvadīts paziņojums M009 – pie apstiprinošas atbildes attiecīgais fails tiek dzēsts.
- Ja atrodas “save” lapā, nospiežot uz tukša lodziņa, tiek izveidots jauns spēles fails ar attiecīgo formātu. Ja nospiež uz aizņemta lodziņa, tiek izvadīts paziņojums M010 un pie apstiprinošas atbildes tiek izveidots jauns spēles fails.
- Ja atrodas “load” lapā, nospiežot uz aizņemta lodziņa, tiek ielādēta spēle, balstoties uz attiecīgo spēles failu. Ja spēlētājs jau ir ielādējis spēli, tiek izvadīts paziņojums M011 un spēle ielādēta pie apstiprinošas atbildes.

Tabula 4.30 - Projektējums funkcijai “Parādīt uzdevumu sarakstu”

<b>ID: VIS_004</b>
Ievaddati: <ul style="list-style-type: none"><li>• Uzdevumu dati no lokālās datu krātuves</li></ul>
Obligātie lauki: uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Katram uzdevumu mainīgajam:</li><li>• Ja tā vērtība ir 0, tiek parādīts teksts “Jūs vēl neesat atradis šo uzdevumu.”</li><li>• Citādi tiek parādīts atbilstošais uzdevuma nosaukums un teksts.</li></ul>

#### 4.2.5. Skripta modulis

Tabula 4.31 - Projektējums funkcijai “Jautāt par spēlētāja vecumu”

<b>ID: SCR_001</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li></ul>
Obligātie lauki: spēlētāja ievade

<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Tiek parādīts paziņojums M012.</li> <li>• Pie apstipriņošanas atbildes tiek izsaukta funkcija VIS_001.</li> <li>• Citādi spēles logs tiek aizvērts.</li> </ul>

Tabula 4.32 - Projektējums funkcijai “Uzsākt jaunu spēli”

<b>ID: SCR_002</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Spēlētāja ievade</li> </ul>
Obligātie lauki: spēlētāja ievade
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Tiek parādīta izvēlne.</li> <li>• Atkarībā no izvēlētās opcijas <i>dif</i> (spēles grūtība) tiek iestatīta kā “<i>easy</i>”, “<i>norm</i>” vai “<i>hard</i>”.</li> <li>• Tiek izsaukta funkcija, kas iestata spēles mainīgos.</li> <li>• Tiek izsaukta funkcija, kas definē pretinieku atribūtus.</li> <li>• Tiek izsaukta funkcija TXT_004 ar nosaukumu <i>prologue</i>.</li> </ul>

#### 4.2.6. Teksta modulis

Tabula 4.33 - Projektējums funkcijai “Displeja funkcija”

<b>ID: TXT_001</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Funkcijas nosaukums</li> </ul>
Obligātie lauki: funkcijas nosaukums

<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Kodā tiek izsaukta <i>label</i> funkcija ar attiecīgo nosaukumu.</li> <li>• Tiek parādīts atbilstošais teksts un grafikas.</li> <li>• Tiek izsaukta viena no šīm funkcijām: SBX_001, SBX_002 vai teksta moduļa funkcija ar attiecīgo, kodā minēto nosaukumu.</li> </ul>

Tabula 4.34 - Projektējums funkcijai “Ievades funkcija”

<b>ID: TXT_002</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Funkcijas nosaukums</li> <li>• Spēlētāja ievade</li> <li>• Spēlētāja dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: funkcijas nosaukums, spēlētāja ievade, spēlētāja dati
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Kodā tiek izsaukta <i>label</i> funkcija ar attiecīgo nosaukumu.</li> <li>• Tiek parādīts atbilstošais teksts un grafikas. <ul style="list-style-type: none"> <li>▪ Atkarībā no spēlētāja izvēles vai spēlētāja datiem tiek parādīts atšķirīgs teksts un grafikas.</li> </ul> </li> <li>• Tiek izsaukta vai nu funkcija SBX_001, vai teksta moduļa funkcija ar attiecīgo, kodā minēto nosaukumu.</li> </ul>

Tabula 4.35 - Projektējums funkcijai “Funkcija ar rediģēšanu”

<b>ID: TXT_003</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Funkcijas nosaukums</li> </ul>
Obligātie lauki: funkcijas nosaukums

<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Kodā tiek izsaukta <i>label</i> funkcija ar attiecīgo nosaukumu.</li> <li>• Tiek parādīts atbilstošais teksts un grafikas.</li> <li>• Tiek mainīts vismaz viens no spēles mainīgajiem – lielākajā daļā gadījumu uzdevumu mainīgie <i>questx</i>, patikšanas mainīgie <i>xx_like</i>, vai izsaukta funkcija SBX_009 ar attiecīgo <i>questx</i> un posma vērtību.</li> <li>• Tiek izsaukta vai nu funkcija SBX_001, vai teksta moduļa funkcija ar attiecīgo, kodā minēto nosaukumu.</li> </ul>

Tabula 4.36 - Projektējums funkcijai “Ievades funkcija ar rediģēšanu”

<b>ID: TXT_004</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Funkcijas nosaukums</li> <li>• Spēlētāja ievade</li> <li>• Spēlētāja dati no lokālās datu krātuves</li> </ul>
<p>Obligātie lauki: funkcijas nosaukums, spēlētāja ievade, spēlētāja dati</p>
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Kodā tiek izsaukta <i>label</i> funkcija ar attiecīgo nosaukumu.</li> <li>• Tiek parādīts atbilstošais teksts un grafikas. <ul style="list-style-type: none"> <li>▪ Atkarībā no spēlētāja izvēles vai spēlētāja datiem tiek parādīts atšķirīgs teksts un grafikas.</li> </ul> </li> <li>• Tiek mainīts vismaz viens no spēles mainīgajiem – lielākajā daļā gadījumu uzdevumu mainīgie <i>questx</i>, patikšanas mainīgie <i>xx_like</i>, vai izsaukta funkcija SBX_009 ar attiecīgo <i>questx</i> un posma vērtību.</li> <li>• Tiek izsaukta vai nu funkcija SBX_001, vai teksta moduļa funkcija ar attiecīgo, kodā minēto nosaukumu.</li> </ul>

Tabula 4.37 - Projektējums funkcijai "Cīņas izsaukums"

<b>ID: TXT_005</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Funkcijas nosaukums</li> </ul>
Obligātie lauki: funkcijas nosaukums
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Kodā tiek izsaukta <i>label</i> funkcija ar attiecīgo nosaukumu.</li> <li>• Tiek parādīts atbilstošais teksts un grafikas.</li> <li>• Tiek iestatīti pretinieku <i>enemy1</i>, <i>enemy2</i>, <i>enemy3</i> mainīgie un izsaukta funkcija COM_001.</li> <li>• Ja <i>lost</i> ir <i>True</i>, tiek izsaukta funkcija SBX_007.</li> <li>• Citādi var tikt izsaukta funkcija SBX_009, un tiek izsaukta vai nu funkcija SBX_001, vai teksta moduļa funkcija ar attiecīgo, kodā minēto nosaukumu.</li> </ul>

Tabula 4.38 - Projektējums funkcijai "Funkcija ar nejaušu mainīgo"

<b>ID: TXT_006</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Funkcijas nosaukums</li> </ul>
Obligātie lauki:
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Kodā tiek izsaukta <i>label</i> funkcija ar attiecīgo nosaukumu.</li> <li>• Tiek nejauši aprēķināta mainīgā vērtība.</li> <li>• Atkarībā no šī mainīgā vērtības tiek parādīts atšķirīgs teksts un grafikas.</li> <li>• Tiek izsaukta vai nu funkcija SBX_001, vai teksta moduļa funkcija ar attiecīgo, kodā minēto nosaukumu.</li> </ul>

#### 4.2.7. Galvenais modulis

Tabula 4.39 - Projektējums funkcijai "Rīta funkcija"

<b>ID: LOCM_001</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li></ul>
Obligātie lauki: spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>

Apstrāde:

- Tiek atiestatīti *pc.rp* uz 0 un *pc.hp* uz *pc.hp1*.
- Atkarībā no noteiktu uzdevumu vērtībām tiek izsauktas noteiktas tekstu moduļu funkcijas un 3. un 4.punkts izlaists.
- Atkarībā no *quest5* (Kilenas uzd.) vērtības tiek izsaukta noteiktā teksta moduļa funkcija.
- Atkarībā no citu uzdevumu vērtībām tiek izsaukti sapņi – TXT\_001 funkcijas.
- Atkarībā no *quest5* vērtības tiek izsaukta noteikta TXT\_003 funkcija.
- Tiek aprēķināts nejaušs mainīgais. Atkarībā no mainīgā vērtības tiek parādīts atšķirīgs teksts un grafikas.
- Tiek izsaukta funkcija SBX\_006.
- Tiek izsaukta funkcija SBX\_008.

Tabula 4.40 - Projektējums funkcijai “Nakts funkcija”

<b>ID: LOCM_002</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li><li>• Dienu skaits <i>daycount</i></li></ul>
Obligātie lauki: spēlētāja dati, uzdevumu dati, <i>daycount</i>
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>

Apstrāde:

- Ja uzdevumu *quest5* vai *quest18* vērtība ir noteikts posms, tiek izsaukta attiecīgā TXT\_004 funkcija.
- Ja *daycount* ir 8, tiek piedāvāta izvēlne – ja izvēlas izlaist vakara rutīnu nākotnē, *dop\_var9* tiek iestatīts kā 1.
- Tiek nejauši aprēķināts mainīgais. Atkarībā no šī mainīgā vērtības tiek parādīts atsevišķs teksts.
- Ja *pc.bab* ir mazāks par 20, tiek izsaukta atbilstošā TXT\_001 funkcija un SBX\_007. Citādi *pc.bab* samazinās par 20.
- Tiek izsaukta funkcija LOCM\_001.

Tabula 4.41 - Projektējums funkcijai “Atpūsties istabā”

<b>ID: LOCM_003</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja dati no lokālās datu krātuves</li></ul>
Obligātie lauki: spēlētāja dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Tiek parādīts atbilstošais teksts un grafikas.</li><li>• Palielina <i>pc.hp</i> par 30; ja <i>pc.hp</i> pārsniedz <i>pc.hp1</i>, <i>pc.hp</i> tiek iestatīts kā <i>pc.hp1</i>.</li><li>• Tiek izsaukta funkcija SBX_001.</li></ul>

Tabula 4.42 - Projektējums funkcijai “Trenēties istabā”

**ID: LOCM\_004**

Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Mainīgais, kas nosaka palielināmo atribūtu: ‘s’ vai ‘a’</li> </ul>
Obligātie lauki: spēlētāja dati, mainīgais
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Mainīgais nosaka, kuru atribūtu palielinās: ‘s’ palielina spēku, ‘a’ palielina veiklību. Abiem process vienāds.</li> <li>• Tiek pieskaitīta viena trenēšanās reize (<i>st_tr_cur</i> vai <i>ag_tr_cur</i>). Ja trenēšanās reizes sasniedz maksimālo (<i>st_tr_max</i> vai <i>ag_tr_max</i>), <i>tr_cur</i> tiek atiestatīts, <i>tr_max</i> palielinās par 5 un noteiktais atribūts (<i>pc.str</i> vai <i>pc.dex</i>) palielinās par 1.</li> <li>• Tiek palielināts <i>dop_var24</i> par 1. Ja <i>dop_var24</i> ir 10, tiek izsaukta funkcija ACH_002 ar sasnieguma nosaukumu ‘tiger’.</li> <li>• Tiek izsaukta funkcija SBX_008.</li> </ul>

Tabula 4.43 - Projektējums funkcijai “Apmeklēt krogu”

<b>ID: LOCM_005</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• Ja <i>pc.wash</i> &gt; 3, tiek izsaukta funkcija SBX_002 un funkcijas darbība netiek turpināta.</li> </ul>

#### Apstrāde:

- Tiek aprēķināts nejaušs mainīgais. Atkarībā no mainīgā vērtības var tikt izsaukta noteikta TXT\_002 funkcija.
- Citādi parādās izvēlne. Galvenās divas opcijas ir pasūtīt ēdienu un iet uz istabu (izsauc SBX\_008). Citas opcijas parādās, atkarībā no uzdevumu vērtībām, kas izsauc atbilstošās teksta moduļa funkcijas.
- Ja izvēlas pasūtīt ēdienu, *pc.bab* tiek samazināts par 5. Tālākās izvēlnes atkarīgas no *mr\_like*.
  - Ja *mr\_like*  $\leq 3$ , var samaksāt dzeramnaudu – 1 vienību (*pc.bab*-1), kas palielina *mr\_like* par 1, vai tikai pateikt paldies (izsauc TXT\_001).
  - Ja  $3 < mr\_like \leq 6$ , var samaksāt vai nu 1, vai 5 naudas vienības dzeramnaudā; *mr\_like* palielinās tikai tad, ja samaksā 5 vienības.
  - Ja  $7 \leq mr\_like \leq 10$ , var samaksāt vai nu 5, vai 10 naudas vienības dzeramnaudā; *mr\_like* palielinās tikai tad, ja samaksā 10 vienības. Var arī pateikt paldies, kas samazina *mr\_like* par 1.
  - Ja *mr\_like*  $> 10$ , var samaksāt 15 naudas vienības dzeramnaudā, kas palielina *mr\_like* par 1, vai pateikt paldies, kas samazina *mr\_like* par 1.
- Tiek parādīta izvēlne, vai nu tikai atpūsties, vai klausīties klientus.
  - Ja izvēle ir atpūsties, tiek aprēķināts nejaušs mainīgais, un atkarībā no šī mainīgā tiek parādīts atšķirīgs teksts un grafikas.
  - Ja izvēle ir klausīties, tiek aprēķināts nejaušs mainīgais, un atkarībā no šī mainīgā, ja *quest8*, *quest9* vai *quest10* vērtības ir 0, katram uzdevumam ir 33% iespēja tikt uzsāktam – tiek izsaukta funkcija SBX\_009 ar attiecīgo uzdevuma mainīgo un vērtību 1.
- Tiek izsaukta funkcija SBX\_001.

#### 4.2.8. Centra modulis

Tabula 4.44 - Projektējums funkcijai "Tirgoties ar alķīmiķi"

ID: LOCC_001
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Ja <i>seen_apt</i> ir <i>False</i>, tas tiek iestatīts kā <i>True</i> un tiek parādīts papildus teksts.</li><li>• Tiek parādīta izvēlne:<ul style="list-style-type: none"><li>▪ Noteiktas opcijas ir tikai pieejamas, atkarībā no uzdevuma <i>m_quest2</i>, <i>quest9</i> vai <i>quest24</i> vērtībām.</li><li>▪ Tirgošanās: tiek parādīta izvēlne ar lietām, kuras alķīmiķis pārdod. Izvēloties vienu no šīm lietām, tiek samazināts <i>pc.bab</i> par noteikto daudzumu un palielināts dziras daudzuma mainīgais par 1.</li><li>▪ Laupījuma pārdošana: ja <i>loot</i> &gt; 0, var pārdot <i>loot</i> – <i>pc.bab</i> tiek palielināts ar <i>loot*x</i>, kur <i>x</i> ir robežās no 0.9 līdz 1.1. <i>loot</i> tiek atiestatīts.</li><li>▪ Pēdējā izvēle izsauc funkciju SBX_002.</li></ul></li></ul>

Tabula 4.45 - Projektējums funkcijai "Tirgoties ar bruņotāju"

ID: LOCC_002
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li></ul>

Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• Ja <i>pc.wash</i> &gt; 3, tiek izsaukta SBX_002 un funkcijas darbība netiek turpināta.</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Ja <i>seen_arm</i> ir <i>False</i>, tas tiek iestatīts kā <i>True</i> un tiek parādīts papildus teksts.</li> <li>• Tiek parādīta izvēlne: <ul style="list-style-type: none"> <li>▪ Noteiktas opcijas ir tikai pieejamas, atkarībā no uzdevuma <i>m_quest2</i>, <i>m_quest3</i> vai <i>quest10</i> vērtībām.</li> <li>▪ Bruņu tirgošana: tiek parādīta izvēlne ar bruņām, kuras bruņotājs pārdod. Izvēloties vienu no šīm lietām, ja pietiek <i>pc.bab</i> un <i>pc.str</i>, tiek samazināts <i>pc.bab</i> par noteikto daudzumu un iestatītas <i>pc.phd</i> un <i>dop_var3</i> vērtības.</li> <li>▪ Zobens uzlabojumi: pieejams, ja <i>pc.rep</i> &gt;= 5. Katrā pieejamajā izvēlē <i>pc.bab</i> samazinās par noteiktu daudzumu. Zobens asināšana iestata <i>dop_var29</i> vērtību kā 5; pārējās iestata vai nu <i>pc.sk1</i>, vai <i>pc.sk2</i> vērtību, atkarībā no uzlabojuma.</li> <li>▪ Pēdējā izvēle izsauc funkciju SBX_002.</li> </ul> </li> </ul>

Tabula 4.46 - Projektējums funkcijai "Apmeklēt saunu"

<b>ID: LOCC_003</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja ievade</li> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>

Apstrāde:

- Ja *bath\_chain* ir 0, tiek parādīts papildus teksts un izsaukta funkcija SBX\_009 priekš *quest7*.
- Tiek parādīta izvēlne:
  - Jautāt saunas vadītājam par pilsētu – atkarīgs no *m\_quest2* vērtības.
  - Parasta nomazgāšanās – samazinās *pc.bab* par 10, tiek atiestatīts *pc.hp* un *pc.wash*.
  - Mazgāšanās ar masāžu – samazinās *pc.bab* par 25, iestatīta *dop\_var6* vērtība kā 6 un tam sekojot iestatīta jauna *pc.hp1* vērtība, atiestatīti *pc.hp* un *pc.wash*, palielināta *bath\_chain* vērtība par 1. Tālākās darbības ar Londu ir atkarīgas no *quest7* vērtības.
- Beigās tiek izsaukta funkcija SBX\_001.

Tabula 4.47 - Projektējums funkcijai “Apmeklēt festivālu”

ID: LOCC_004
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li><li>• Diena <i>theday</i> un dienas laiks <i>daytime</i></li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• <i>theday</i> ir jābūt 7 un <i>daytime</i> ir jābūt 'evening'.</li></ul>

Apstrāde:

- Ja uzdevumam *quest5* ir noteikta vērtība, tiek izsaukta noteikta TXT\_004 funkcija.
- Ja *pc.bab* < 5, tiek izsaukta funkcija SBX\_002 un šī funkcija netiek turpināta.
- Citādi dota izvēle, samaksāt par festivālu (*pc.bab* – 5) vai nesamaksāt (SBX\_002 izsaukums).
- Ja samaksāja, tiek parādīta izvēlne, ko darīt festivālā: meklēt Kilenu, Mrišju, Londu vai Ilgu (kas izsauc atbilstošās teksta moduļa funkcijas un *holiday\_count* palielinās par 1), paskatīties apkārt (izsaukta atbilstošā TXT\_004 funkcija atkarībā no *quest22* vai *quest24* vērtībām), vai pamest festivālu (izsauc SBX\_001).
- Ja *holiday\_count* >= 6, tiek automātiski izsaukta funkcija SBX\_001.

Tabula 4.48 - Projektējums funkcijai “Apmeklēt baznīcu”

<b>ID: LOCC_005</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>

Apstrāde:

- Ja *dop\_var13* ir 0:
  - *dop\_var13* tiek iestatīts kā 1.
  - Tiek parādīta izvēlne – samaksāt par pakalpojumu vai nesamaksāt. Ja samaksā, tad *pc.bab* samazinās par 35.
  - Jebkurā gadījumā *dop\_var7* tiek iestatīts kā 4 un *pc.mad* palielinās par 1.
- Citādi tiek piedāvāta izvēlne:
  - Svētīšana, ja pietiek naudas un *dop\_var27* ir 0 – *pc.bab* samazinās par 35, *dop\_var7* iestatīts kā 5 un *pc.mad* palielinās par 1.
  - Atzīšanās, ja pietiek naudas un *dop\_var7* ir 0 – *pc.bab* samazinās par 85, *dop\_var27* iestatīts kā 5 un *pc.mad* palielināts par 2.
  - Pēdējā izvēle izsauc funkciju SBX\_002.

Tabula 4.49 - Projektējums funkcijai “Apmeklēt pili”

<b>ID: LOCC_006</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Ja <i>seen_castle</i> ir <i>False</i>, tiek parādīts papildus teksts.</li><li>• Tiek parādīta izvēlne:<ul style="list-style-type: none"><li>▪ Trenēties pilī – pieejams tikai tad, kad <i>pc.rep</i> &gt; 5. Tiek izsaukta funkcija gandrīz identiska LOCM_004, bet tiek palielināti gan spēka, gan veiklības atribūti.</li><li>▪ Satikt baronu vai pili pamest – abos gadījumos tiek izsaukta noteiktā TXT_001 funkcija un SBX_001.</li></ul></li></ul>

Tabula 4.50 - Projektējums funkcijai "Apmeklēt mēru"

<b>ID: LOCC_007</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Spēlētāja ievade</li> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja dati, uzdevumu dati
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• <i>m_quest3</i> nedrīkst būt 0; citādi izsaukta funkcija SBX_002.</li> </ul>
<p>Apstrāde:</p> <ul style="list-style-type: none"> <li>• Ja <i>m_quest3</i> vērtība ir 1 vai 3, tiek izsaukta attiecīgi atbilstošā TXT_003 vai TXT_002 funkcija.</li> <li>• Ja <i>quest22</i> ir 3 vai <i>quest23</i> ir 2, tiek izsaukta atbilstošā TXT_003 funkcija, kurā tiek palielināts <i>pc.bab</i> un izsaukta funkcija SBX_009 ar noteiktajiem parametriem.</li> <li>• Citādi, ja <i>m_quest3</i> <math>\geq 4</math>, tiek dota izvēlne: <ul style="list-style-type: none"> <li>▪ Prasīt par uzdevumu, ja <i>quest8</i> ir 0. Tiek izsaukta TXT_004 funkcija, kurā: Tiek aprēķināts nejaušs mainīgais – 33% iespēja iegūt uzdevumu. Ja piekrīt uzdevumam, tiek izsaukta funkcija SBX_009.</li> <li>▪ Pabeigt uzdevumu, ja <i>quest8</i> ir 2. Tiek atiestatīta <i>quest8</i> vērtība ar funkciju SBX_009, palielināts <i>pc.bab</i> par 80 un <i>pc.rep</i> par 1.</li> </ul> </li> <li>• Tiek izsaukta funkcija SBX_001.</li> </ul>

Tabula 4.51 - Projektējums funkcijai "Eksportēt spēli"

<b>ID: LOCC_008</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• Spēles versijai <i>game_version</i> jābūt '1.0.41' – citādi iegūtais fails būs tālāk nelietojams.</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Formātā <i>eg.[nosaukums]</i> tiek saglabāti sekojošie mainīgie: <ul style="list-style-type: none"> <li>▪ Visi patikšanas mainīgie.</li> <li>▪ Diena <i>theday</i> un tās nosaukums <i>weekd</i>.</li> <li>▪ Uzdevumu mainīgie: <i>m_quest1 – m_quest9, quest1 – quest25</i>.</li> <li>▪ Spēlētāja dati ar formātu <i>pc.[atribūts]</i>.</li> <li>▪ <i>bath_chain, dop_var12, dop_var20, dop_var24, dop_var30</i>.</li> </ul> </li> <li>• Izsaukta Ren'Py iebūvētā spēles faila saglabāšanas funkcija un izvadīts paziņojums M013.</li> </ul>

## 4.2.9. Priekšpilsētas modulis

Tabula 4.52 - Projektējums funkcijai "Apmeklēt bordeli"

ID: LOCK_001
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"><li>• Ja <i>pc.wash</i> &gt; 3, tiek izsaukta funkcija SBX_002 un funkcijas darbība netiek turpināta.</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Ja <i>seen_bro</i> ir <i>False</i>, tas tiek mainīts uz <i>True</i> un tiek parādīts papildus teksts.</li><li>• Ja <i>m_quest3</i> ir 4, tiek izsaukta atbilstošā TXT_004 funkcija.</li><li>• Citādi tiek parādīta izvēlne:<ul style="list-style-type: none"><li>▪ Noteiktas opcijas ir tikai pieejamas, atkarībā no uzdevuma <i>m_quest2</i>, <i>m_quest3</i> vai <i>quest2</i> vērtībām.</li><li>▪ Satikt meiteni – tiek parādīta izvēlne ar meitenēm. Izvēloties meiteni, tiek samazināts <i>pc.bab</i> par noteikto daudzumu un palielināts meiteņu tikšanās reižu skaitītājs. Kad skaitītājs sasniedz 10, tiek izsaukta funkcija ACH_002 ar sasnieguma nosaukumu 'zavs'.</li></ul></li><li>• Tiek izsaukta funkcija SBX_001.</li></ul>

Tabula 4.53 - Projektējums funkcijai "Apmeklēt Ilgas māju"

<b>ID: LOCK_002</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li><li>• Dienas laiks <i>daytime</i></li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati, <i>daytime</i>
Citas pārbaudes: <ul style="list-style-type: none"><li>• Ja <i>pc.wash</i> &gt; 3, tiek izsaukta funkcija SBX_002 un funkcijas darbība netiek turpināta.</li></ul>

Apstrāde:

1. Ja *daytime* ir 'morning':
  - a. Ja *quest1* ir 4 vai 6, tiek izsaukta atbilstošās teksta moduļa funkcijas.
  - b. Ja *quest16* ir 2, 8 vai 9, tiek izsaukta atbilstošā TXT\_001 vai TXT\_004 funkcija.
  - c. Citādi tiek izsaukta funkcija SBX\_002.
2. Ja *daytime* ir 'day' vai 'evening':
  - a. Ja *daytime* ir 'day' un *dop\_var17* ir 0, tiek izsaukta TXT\_002 funkcija un *dop\_var17* iestatīts kā 1.
  - b. Citādi tiek izsaukta izvēlne – izvēlnē pieejamās opcijas ir atkarīgas no *il\_help*, *il\_like* un uzdevumu *m\_quest2*, *quest1*, *quest3*, *quest16* un *quest24* vērtībām.
  - c. Ja opcija nav saistīta ar *m\_quest2* vai *quest24*, tad ir iespēja mainīt *il\_like*.
    - i. Ja opcija ir saistīta ar *quest1* vai *quest3*, palielinās *il\_like* par 1 veiksmīgu atbilžu sniegšanas gadījumā, izsauktas atbilstošās teksta moduļa funkcijas un izsaukta funkcija SBX\_009 ar attiecīgajiem parametriem.
    - ii. Ja opcija ir saistīta ar *quest16*, *il\_like* samazinās par 1 un tiek izsaukta atbilstošā TXT\_004 funkcija.
  - d. Tiek izsaukta funkcija SBX\_001.
3. Ja *daytime* ir 'night', tiek izsaukta funkcija SBX\_002.

Tabula 4.54 - Projektējums funkcijai "Pastaigāties pa pilsētu"

ID: LOCK_003
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li><li>• Dienas laiks <i>daytime</i></li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati, <i>daytime</i>

Citas pārbaudes:

- –

Apstrāde:

1. Tiek aprēķināts nejaušs mainīgais.
2. Ja *daytime* ir ‘*morning*’:
  - a. 35% iespēja izsaukt TXT\_001 funkciju, kur *pc.bab* palielinās par 5 vai 10;
  - b. 25% iespēja izsaukt TXT\_002 funkciju, kur pēc izvēles var izsaukt TXT\_005 funkciju.
3. Ja *daytime* ir ‘*day*’:
  - a. Ja *quest17* ir 2, tiek palielināts *dop\_var19* par 1 – kad *dop\_var19* ir 4, tiek izsaukta TXT\_004 funkcija.
  - b. Ja *quest17* ir 3, tiek izsaukta TXT\_004 funkcija.
  - c. Citādi:
    - i. 25% iespēja izsaukt TXT\_001 funkciju, kur *pc.bab* palielinās par 5 vai 10;
    - ii. 30% iespēja izsaukt TXT\_001 funkciju ar pāri;
    - iii. 20% iespēja izsaukt TXT\_002 funkciju ar Kilenu.
4. Ja *daytime* ir ‘*evening*’:
  - a. 10% iespēja izsaukt TXT\_001 funkciju, kur *pc.bab* palielinās par 5 vai 10;
  - b. 20% iespēja izsaukt TXT\_005 funkciju, kas izsauc SBX\_009 atkarībā no *quest8* vērtības;
  - c. 30% iespēja redzēt TXT\_001 funkciju ar pāri;
  - d. 15% iespēja izsaukt TXT\_002 funkciju, kas tālāk ar 50/50 iespējamību izsauc vai nu TXT\_001, vai TXT\_005 funkciju.
5. Ja *daytime* ir ‘*night*’:
  - a. 50% iespēja izsaukt 4.b. punktā minētu funkciju.
6. Visos citos gadījumos tiek izsaukta funkcija SBX\_001.

#### 4.2.10. Ārpilsētas modulis

Tabula 4.55 - Projektējums funkcijai "Pastaigāties pa mežu/upi"

<b>ID: LOCO_001</b>
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li><li>• Dienas laiks <i>daytime</i></li></ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati, <i>daytime</i>
Citas pārbaudes: <ul style="list-style-type: none"><li>• –</li></ul>
Apstrāde: <ol style="list-style-type: none"><li>1. Ja tiek izvēlēts izmeklēt ārpilsētu:<ol style="list-style-type: none"><li>a. Tiek nejauši aprēķināts mainīgais.</li><li>b. Ja <i>seen_loc2</i> ir <i>False</i>, 33% iespēja, ka <i>seen_loc2</i> tiks iestatīts kā <i>True</i>; tiek izsaukta funkcija SBX_001.</li><li>c. Ja <i>seen_loc4</i> ir <i>False</i>, 33% iespēja, ka <i>seen_loc4</i> tiks iestatīts kā <i>True</i>; tiek izsaukta funkcija SBX_001.</li><li>d. 33% iespēja izsaukt LOCO_002.</li></ol></li><li>2. Ja tiek izvēlēts peldēties upē:<ol style="list-style-type: none"><li>a. Ja <i>dop_var5</i> ir 0, tā vērtību iestata kā 1.</li><li>b. Ja <i>daytime</i> ir 'day' un <i>quest15</i> ir 1, tiek izsaukta atbilstošā TXT_004 funkcija.</li><li>c. Citādi tiek nejauši aprēķināts mainīgais: 40% iespēja, ka tiks izsaukta noteikta teksta moduļa funkcija.</li><li>d. 60% iespēja peldēties upē:<ol style="list-style-type: none"><li>i. Tiek nejauši aprēķināts mainīgais.</li><li>ii. 90% iespēja, ka samazināsies <i>pc.wash</i> par 1 un palielināsies <i>pc.hp</i> par 20 (bet ne vairāk par <i>pc.hp1</i>), un tiks izsaukta funkcija SBX_001; 10% iespēja izsaukt LOCO_002.</li></ol></li></ol></li></ol>

Tabula 4.56 - Projektējums funkcijai “Medīt mežā/upē”

<b>ID: LOCO_002</b>
<p>Ievaddati:</p> <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> <li>• Dienas laiks <i>daytime</i></li> </ul>
<p>Obligātie lauki: spēlētāja dati, uzdevumu dati, <i>daytime</i></p>
<p>Citas pārbaudes:</p> <ul style="list-style-type: none"> <li>• –</li> </ul>
<p>Apstrāde:</p> <ol style="list-style-type: none"> <li>1. Atkarībā no <i>daytime</i>, <i>m_quest3</i> un <i>quest23</i> vērtībām, var tikt izsaukta TXT_005 funkcija, kas saistīta ar šiem uzdevumiem.</li> <li>2. Citādi tiek nejauši aprēķināts mainīgais: <ol style="list-style-type: none"> <li>a. Ja <i>pc.lev</i> <math>\geq 3</math> un <i>daytime</i> ir ‘day’, 50% iespēja izsaukt TXT_005 funkciju ar fūriju.</li> <li>b. Ja <i>daytime</i> ir ‘day’, 50% iespēja izsaukt TXT_005 funkciju ar goblinu.</li> <li>c. Ja <i>daytime</i> nav ‘day’, 75% iespēja atrast pretinieku.</li> <li>d. Ja lokācija ir upe, 70% iespēja atrast pretinieku.</li> </ol> </li> <li>3. Pretinieks tiek iestatīts, atkarībā no nejauši aprēķināta mainīgā. Pēc pretinieka iestatīšanas, tas ir, vērtību piešķiršanas <i>enemy1</i>, <i>enemy2</i> un <i>enemy3</i>, tiek izsaukta funkcija COM_001.</li> <li>4. Tiek izsaukta funkcija SBX_001.</li> </ol>

Tabula 4.57 - Projektējums funkcijai “Apmeklēt kalnu”

<b>ID: LOCO_003</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• <i>quest24</i> vērtībai ir jābūt 5 un uzdevumā iekļautajam mainīgajam <i>q24tr</i> ir jābūt False – citādi tiek izvadīts paziņojums M014.</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Tiek izsaukta atbilstošā TXT_005 funkcija, pēc kuras tiek izsaukta COM_001 un, ja <i>lost</i> ir <i>False</i>, tiek izsauktas atbilstošās teksta moduļa funkcijas.</li> </ul>

Tabula 4.58 - Projektējums funkcijai “Apmeklēt kabīni”

<b>ID: LOCO_004</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Uzdevumu dati no lokālās datu krātuves</li> <li>• Dienas laiks <i>daytime</i></li> </ul>
Obligātie lauki: uzdevumu dati, <i>daytime</i>
Citas pārbaudes: <p>Jāizpildās viena no sekojošajām:</p> <ul style="list-style-type: none"> <li>• <i>quest21</i> vērtība ir 1 un <i>daytime</i> ir ‘<i>day</i>’</li> <li>• <i>quest16</i> vērtība ir 9 vai 11 un <i>daytime</i> ir ‘<i>evening</i>’</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Atkarībā no izpildītajiem nosacījumiem tiek izsaukta atbilstošā TXT_003 funkcija un mainīta atmiņas mainīgo vērtība uz <i>True</i>.</li> </ul>

## 5. PROGRAMMATŪRAS KODĀ VEIKTĀS IZMAIŅAS

Šajā nodaļā ir aprakstītas visas programmatūras kodā veiktās izmaiņas, sadalītas pa moduļiem. Visu izmaiņu kods ir dokumentēts pielikumu nodaļā.

Lai mainītu Ren'Py kodu, spēli sākumā ir nepieciešams dekompilēt. Kompilētas Ren'Py spēles game mape satur *.rpyc* failus, kas nav koda redaktoriem lasāmi, kā arī *res.rpa* failu – šis fails satur visas spēles stāsta grafikas, mūziku un saskarnes grafikas.

Spēli ir iespējams dekompilēt ar brīvprogrammatūru. Darba autors izmantoja rīku *UnRen.bat*, kura veidotājs ir *F95Sam* un kurš ir pieejams par brīvu platformā GitHub.<sup>[17]</sup>

Šo *.bat* failu iekopē game mapē un palaiž. Ir dotas vairākas opcijas; šajā gadījumā pietiek ar trim - *.rpa* pakotņu izvilkšana, kas padara pieejamus grafiku un mūzikas failus, *.rpyc* failu dekompilēšana, kas tos pārvērš koda redaktoriem lasāmajā *.rpy* formātā, un konsoles iespējošana, kurai spēles logā var piekļūt ar īsceļu *Shift+O*.

### 5.1. Izmaiņu saraksts

- Jaunumi:
  - Indes sistēma
    - Pēc katra uzbrukuma pretiniekam tas tiek saindēts – pēc katras kārtas zaudē dzīvības punktus.
    - Jaunas pievienotas inventāra lietas – indes.
    - Indi var nopirkt no Ilgas apmaiņā pret naudu un indes punktiem – indes punkti tiek palielināti katras dienas sākumā; ierobežo indes daudzumu, kuru var nopirkt. Indi var pirkt tikai tad, kad pabeigts 3.uzdevums jeb *quest3 = 5*.
  - Jauns niknuma uzbrukums – maksā visus spēlētāja niknuma punktus jeb *pc.rp*.
    - Katram pretiniekam pēc kārtas tiek uzbrukts vienā gājienā.
    - Tiek pievienoti dzīvības punkti jeb *pc.hp*, balstoties uz pretiniekiem atņemtajiem dzīvības punktiem.
  - Pievienots *quest11* – uzdevums pie alķīmiķa
    - Jānogalina fūrijas divas reizes, iegūstot divus fūrijas kristālus. Atlīdzība – 200 naudas vienības, +80 par katru nākamo kristālu.

- Pievienots *quest14* – uzdevums bordelī
  - Jāapmeklē visas meitenes vismaz vienreiz.
- Jauns sasniegums
  - Jānogalina 100 monstri. Cilvēki un cilvēkveidīgas būtnes neskaitās.
- Jauna lapa uzdevumu skatā funkcijā VIS\_004 – parāda attiecības ar spēles tēliem.
- Uzlabojumi:
  - Uzlabota lietotā formula *pc.exp* un *pc.bab* palielināšanai funkcijā COM\_005.
    - *pc.exp* atkarīgs no spēlētāja līmeņa *pc.lev* – jo mazāks līmenis, jo vairāk *exp* iegūts.
  - Uzlabota formula *attack\_random* un *hit\_chance* aprēķināšanai funkcijā COM\_006 ar precizitātes bonusu *pop*.
  - Implementēti apraksti uzdevumiem *quest3* un *quest20* funkcijas VIS\_004 skatā.
  - Mainītas iespējamības ainām LOCK\_003 – lai ātrāk progresētu uzdevumos.
  - Mainīta Mrišjas attiecību sistēma LOCM\_005 – dzēsta patikšanas samazināšanās.
  - Izlabotas kļūdas zobena uzlabošanā LOCC\_002 un implementēti bonusi cīņas modulī.
  - Funkcijā SCR\_001 implementēta atbildes saglabāšana, lai paziņojums M012 vairs neparādītos.
- Kļūdu labojumi:
  - Dzēsts liekais *quest4* no funkcijas VIS\_004 skata.
  - Labotas loģiskas kļūdas: patikšanu palielināšanās pareizajās vietās; dzēsti lieki *if-else* sazarojumi, kur nekas nemainās; izlabots, kuram mainīgajam tiek piešķirta vērtība.

## 5.2. Sasniegumu modulis (ach.rpy)

### 5.2.1. Jauna sasnieguma apraksts un lodziņš

Implementēts apraksts jaunajam sasniegumam funkcijā ACH\_001, kas parādās funkcijas ACH\_002 paziņojumā M001 – nosaukums ‘*killer*’; sasniegums tiek iegūts, nogalinot 100 monstros (ne cilvēkus vai cilvēkveidīgas būtnes).

Implementēts lodziņš šim sasniegumam funkcijas ACH\_003 skatā.

## 5.3. Cīņas modulis (combat.rpy)

### 5.3.1. Indes mainīgo implementēšana COM\_001

COM\_001 pievienoti mainīgie *enemy1po*, *enemy2po*, *enemy3po* – visu trīs iespējamo pretinieku saindēšanās statuss. Pievienota nozīme mainīgajam *pop* – precizitātes bonuss.

### 5.3.2. Mainītas formulas COM\_005

COM\_005 veiktas izmaiņas *pc.exp* un *pc.bab* formulās:

- *pc.exp* tiek palielināts par  $\exp \cdot (1 + \frac{1}{pc.lev})$ . Tas nozīmē, ka, jo lielāks līmenis, jo mazāk pieredzes tiek iegūts.
- *pc.bab* tiek palielināts par *bab*\*2.

### 5.3.3. Mainītas formulas COM\_006

COM\_006 mainītas formulas *attack\_random* un *hit\_chance* ar *pop*:

- $attack\_random = rand(-3, 3) + rand(-3, 3) + pop$
- $hit\_chance = pc.tohit - at\_target.ac + attack\_random + mod\_at + pop$

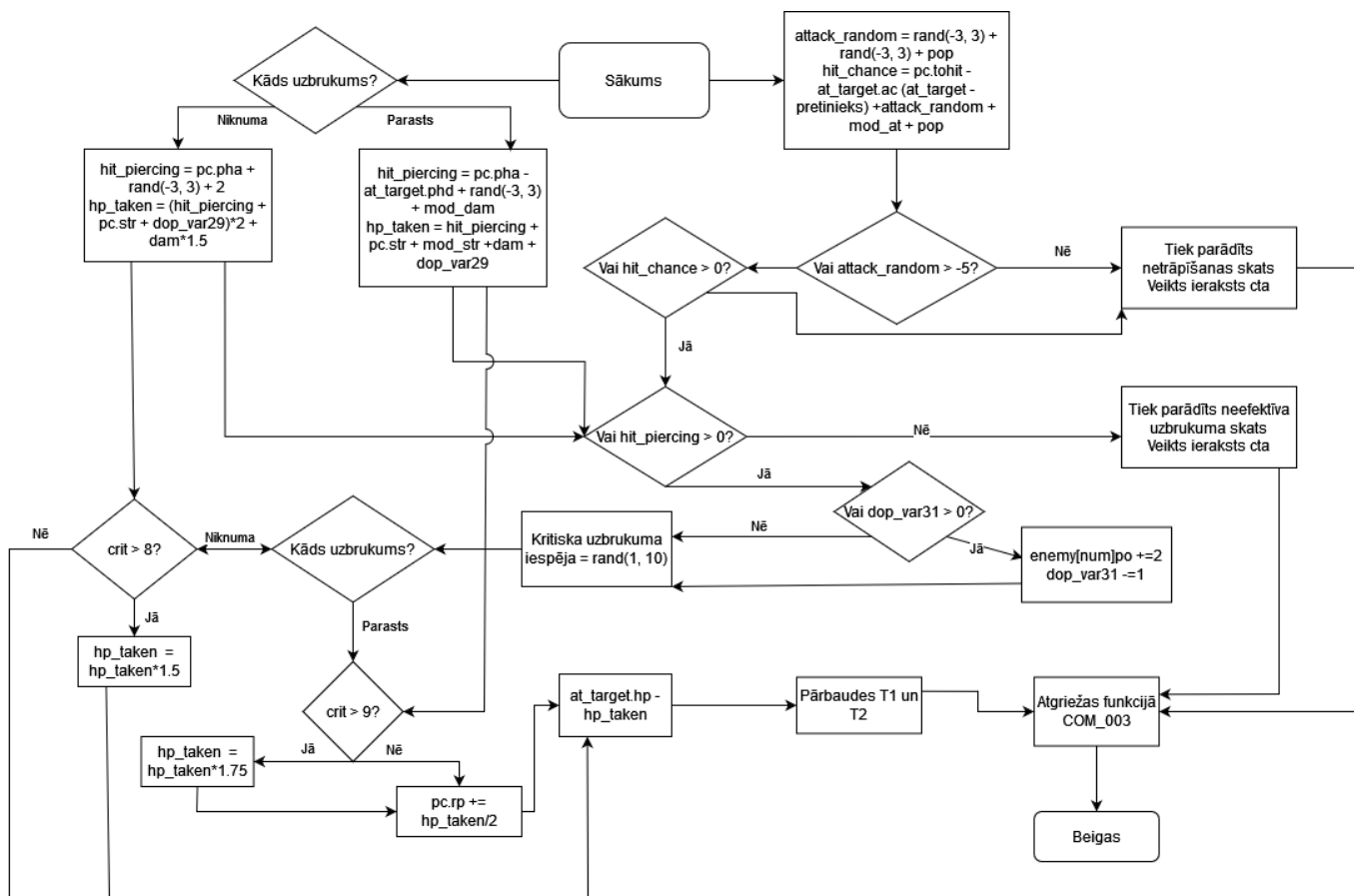
### 5.3.4. Indes efektu izraisīšana COM\_006

Pievienota pretinieka saindēšana, ja ir inde uz zobena (*dop\_var31* > 0). Aprakstā izlaista daļa no projektējuma.

Tabula 5.1 - Atjaunināts daļējs projektējums funkcijai "Uzbrukt pretiniekam"

ID: COM_006
Apstrāde: <ol style="list-style-type: none"><li>1. Ja uzbrukuma veids ir parasts uzbrukums:<ol style="list-style-type: none"><li>a. Tiek aprēķināta uzbrukuma iespējamība <i>attack_random</i> no formulas <math>rand(-3, 3) + rand(-3,3) + pop</math>.</li><li>b. Tiek aprēķināta uzbrukuma iespēja <i>hit_chance</i> no formulas <math>pc.tohit - at\_target.ac</math> (<i>at_target</i> - pretinieks) + <i>attack_random</i> + <i>mod_at</i> + <i>pop</i>.</li><li>c. Ja <i>attack_random</i> ir mazāka/vienāda ar -5 (6% iespēja) vai <i>hit_chance</i> ir mazāka par 0, tiek izsaukts netrāpīšanas skats (ja <i>dop_var10</i> nav 1), veikts ieraksts masīvā <i>cta</i> un izvadīts attiecīgais teksts.</li><li>d. Citādi tiek aprēķināts uzbrukuma spēks <i>hit_piercing</i> no formulas <i>pc.pha</i> (spēlētāja fiziskais uzbrukums) – <i>at_target.phd</i> (pretinieka fiziskā aizsardzība) + <math>rand(-3, 3) + mod\_dam</math>.</li></ol></li></ol>

- e. Ja *hit\_piercing* ir mazāks/vienāds ar 0, tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā *cta*.
  - f. Citādi, ja *dop\_var31* ir lielāks par 0, atkarībā no *at\_target* vērtības attiecīgais *enemy[num]po* palielinās par 2 un *dop\_var31* samazinās par 1.
  - g. Tiek aprēķināta kritiska uzbrukuma iespējamība no  $\text{rand}(1, 10)$ .
  - h. Tiek aprēķināts atņemtais dzīvības punktu skaits *hp\_taken* no formulas  $\text{hit\_piercing} + \text{pc.str}$  (spēlētāja spēks) + *mod\_str* + *dam* + *dop\_var29*.
  - i. Ja kritiskā uzbrukuma iespējamības vērtība ir lielāka/vienāda ar 9 (20% iespēja), *hp\_taken* tiek reizināts ar 1,75, tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā *cta*.
  - j. Citādi tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā *cta*. Ja *mod\_parry* (karodziņš, vai tiek atvairīts uzbrukums) ir 1, *pc.rp* (niknuma punkti) tiek palielināts par  $\text{hp\_taken}/3$ . Citādi *pc.rp* tiek palielināts par  $\text{hp\_taken}/2$ .
  - k. Tiek samazināts *at\_target.hp* (pretinieka dzīvības punkti) par *hp\_taken*.
  - l. Tiek izsauktas pārbaudes T1 un T2 – ja patiesas, attiecīgi tiek atiestatīti *pc.rp* uz maksimālo iespējamo vērtību *pc.rp1*; pretinieka *enemy[num]* vērtība iestatīta uz “none” un *loot*, *exp* un *bab* tiek pieskaitītas attiecīgās vērtības.
2. Ja uzbrukuma veids ir niknuma uzbrukums, notiek tas pats, kas 1.punktā, bet:
    - a. 1.d solī *hit\_piercing* tiek aprēķināts no formulas  $\text{pc.pha} + \text{rand}(-3, 3) + 2$ .
    - b. 1.h solī *hp\_taken* tiek aprēķināts no formulas  $(\text{hit\_piercing} + \text{pc.str} + \text{dop\_var29}) * 2 + \text{dam} * 1.5$ .
    - c. 1.i solis tiek izpildīts, ja uzbrukuma iespējamības vērtība ir lielāka/vienāda ar 8, un *hp\_taken* reizināts ar 1,5.
    - d. 1.j solis tiek ignorēts.



Attēls 5.1 - COM\_006 blokskāma

### 5.3.5. Pievienots glābšanas uzbrukuma izsaukums un indes efekti COM\_003

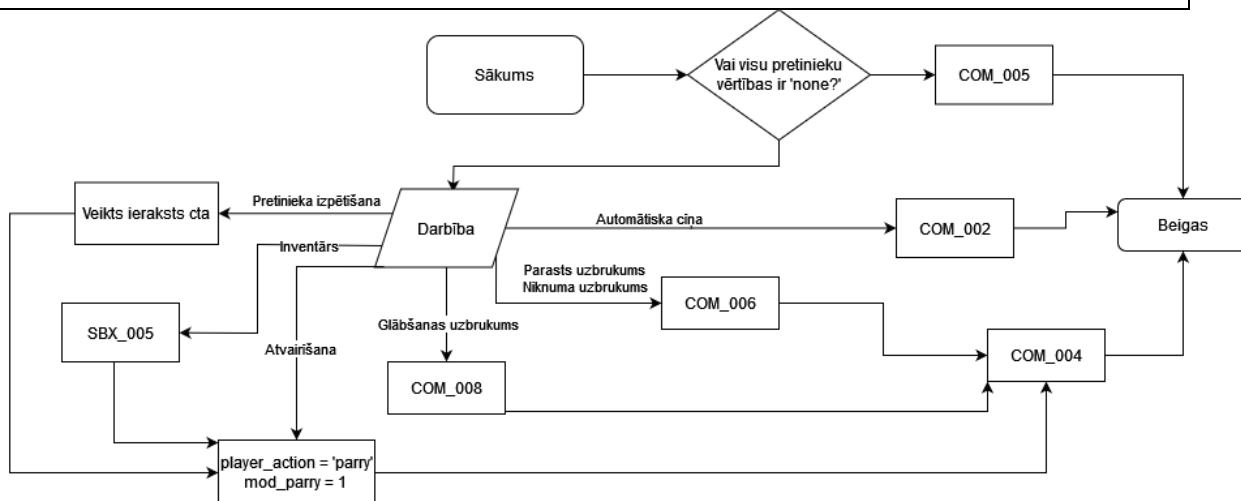
COM\_003 pievienoti papildus soļi (izmaiņas iezīmētas ar dzeltenu krāsu):

Tabula 5.2 - Atjaunināts daļējs projektējums funkcijai "Uzbrukums (spēlētājs)"

ID: COM_003
<p>Apstrāde:</p> <ol style="list-style-type: none"> <li>1. Ja visu pretinieku vērtības <i>enemy[num]</i> ir "none", tiek izsaukta funkcija COM_005.</li> <li>2. Spēlētājs izvēlas nākamo darbību: <ol style="list-style-type: none"> <li>a. Ja darbība ir automātiska cīņa jeb <i>autocombat</i>, <i>dop_var10</i> vērtība tiek iestatīta kā 1 un tiek izsaukta funkcija COM_002.</li> <li>b. Ja tiek izvēlēts spēcīgs uzbrukums, viegls uzbrukums vai parasts uzbrukums, tiek iestatītas atbilstošās vērtības uzbrukuma iespējamības <i>mod_at</i>, uzbrukuma sitiena <i>mod_dam</i> un uzbrukuma spēka <i>mod_str</i></li> </ol> </li> </ol>

- mainīgajiem un spēlētāja darbība definēta kā parasts uzbrukums.
- c. Ja tiek izvēlēta pretinieka izpētīšana:
    - i. Spēlētājs izvēlas, kuru pretinieku pētīs: tiek iestatīts *player\_target*.
    - ii. Atkarībā no atšķirības starp *pc.str* un *player\_target.str*, masīvā *cta* tiek veikts attiecīgais ieraksts. Tas pats ar *pc.dex* un *player\_target.dex*.
    - iii. Spēlētāja darbība tiek iestatīta kā atvairīšana jeb *parry*.
  - d. Ja tiek izvēlēts apskatīt inventāru:
    - i. Tiek paslēpti cīņas skati.
    - ii. Tiek izsaukta saīsināta SBX\_005.
    - iii. Spēlētāja darbība tiek iestatīta kā atvairīšana jeb *parry*.
  - e. Ja tiek veikta atvairīšana, tiek iestatīta *mod\_parry* (karodziņš, vai tiek atvairīts uzbrukums) vērtība uz 1 un masīvā *cta* tiek veikts attiecīgais ieraksts.
  - f. Ja tiek veikta bēgšana, tiek veikts attiecīgais ieraksts masīvā *cta* un:
    - i. Tiek aprēķināts nejaušs mainīgais ar  $\text{rand}(1,3)$ .
    - ii. Ja mainīgais ir lielāks par 1, tiek paslēpti cīņas skati un *dop\_var4* vērtība iestatīta kā 1.
    - iii. Citādi tiek veikts attiecīgais ieraksts masīvā *cta*.
  - g. Ja netiek veikta atvairīšana un netiek veikts glābšanas uzbrukums, spēlētājs izvēlas mērķi – tiek iestatīts *player\_target*, piešķirta attiecīgā vērtība mainīgajam *at\_target*, tiek veikts attiecīgais ieraksts masīvā *cta* un izsaukta funkcija COM\_006.
  - h. Ja tiek veikts glābšanas uzbrukums (pieejams tikai tad, ja  $pc.rp = pc.rp1$ ), tiek atiestatīts *pc.rp* uz 0, iestatīts *target* (mērķa sekotāja mainīgais) uz 2 (jo pirmais pretinieks, kam uzbrūk, ir *enemy2*) un *at\_target* iestatīts kā *enemy2*, un izsaukta funkcija COM\_008.
  - i. Katram pretiniekam, ja *enemy[num]po* ir lielāks par 0 un *enemy[num]* nav 'none':
    - i. Tiek aprēķināts *hp\_taken* no formulas  $\text{rand}(3,8) * \text{enemy}[num]po$ .
    - ii. *enemy[num].hp* tiek samazināts par *hp\_taken*.
    - iii. Tiek samazināts *enemy[num]po* par 1.
    - iv. Tiek veikts attiecīgais ieraksts masīvā *cta*.
  - j. Tiek izsaukta pārbaude T2 un veiktas attiecīgās darbības, ja patiesa.

### 3. Tiek izsaukta funkcija COM\_004.



Attēls 5.2 - COM\_003 blokhēma

#### 5.3.6. COM\_008 – glābšanas uzbrukums

Jauna funkcija: COM\_008 – niknuma uzbrukums visiem pretiniekiem – kodā saukts par “*saving\_attack*” jeb glābšanas uzbrukumu.

Tabula 5.3 – glābšanas uzbrukums

Identifikators: COM_008
Mērķis
Funkcijas mērķis ir veikt niknuma uzbrukumu visiem pretiniekiem, iegūstot dzīvības punktus.
Ievaddati
<ol style="list-style-type: none"> <li>1. Spēlētāja dati no lokālās datu krātuves</li> <li>2. Pretinieku dati no lokālās datu krātuves</li> <li>3. Iestatītie mainīgie no COM_001</li> </ol>
Apstrāde

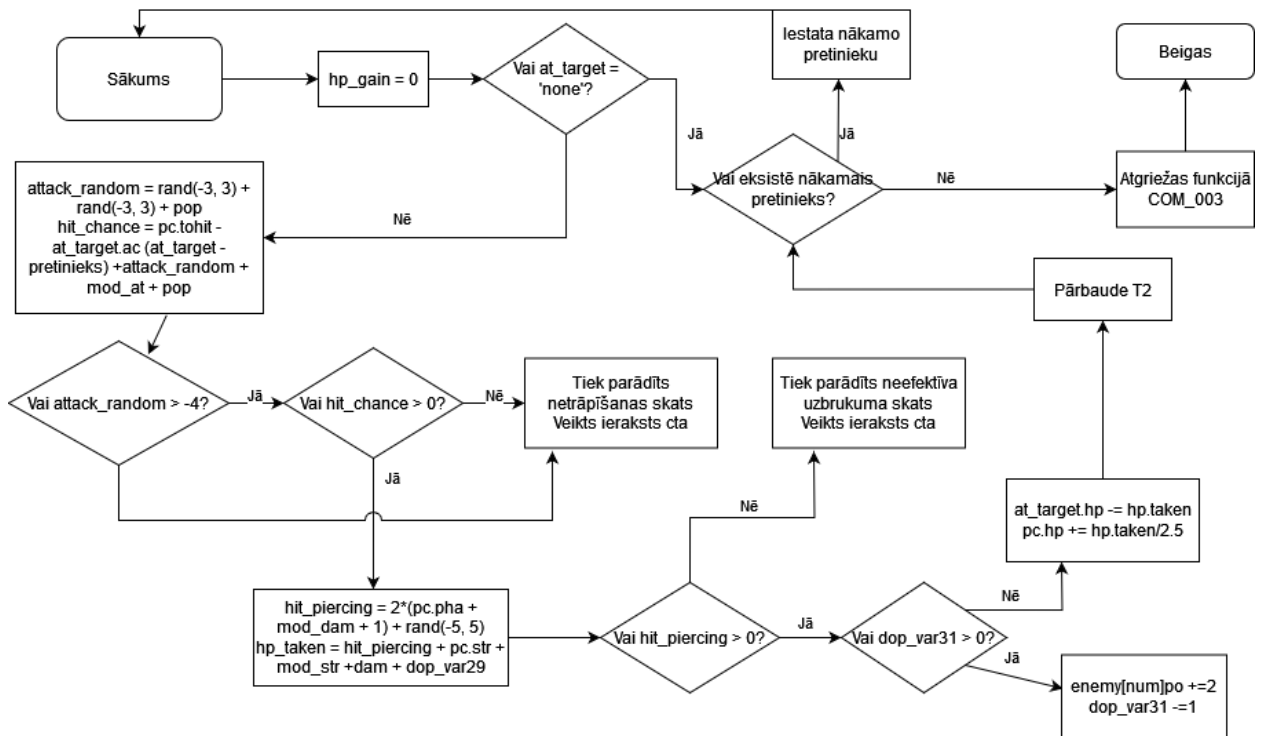
<ol style="list-style-type: none"> <li>1. Katram pretiniekam, ja tas eksistē, tiek aprēķinātas uzbrukumu iespējamības un uzbrukuma spēks. <ol style="list-style-type: none"> <li>a. Uzbrukums ir veiksmīgs tikai tad, ja iespējamība ir pietiekami liela un uzbrukuma spēks nav negatīvs.</li> </ol> </li> <li>2. Ja ir inde uz zobena, pretiniekam tiek palielināts saindēšanās statuss un samazināts indes daudzums uz zobena.</li> <li>3. Tiek veikts uzbrukums – atņemti pretiniekam dzīvības punkti.</li> <li>4. Atkarībā no atņemtajiem dzīvības punktiem tiek spēlētājam pievienoti dzīvības punkti.</li> <li>5. Tiek izvadīti attiecīgie skati un teksts.</li> <li>6. Tiek veikta pārbaude T2 un veiktas attiecīgās darbības.</li> <li>7. Tiek iestatīts nākamais pretinieks, ja eksistē, un atkārtota funkcija, sākot no 1.soļa.</li> </ol>
Izvaddati
–

Tabula 5.4 - Projektējums funkcijai “Glābšanas uzbrukums”

<b>ID: COM_008</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Pretinieku dati no lokālās datu krātuves</li> <li>• Iestatītie mainīgie no COM_001</li> </ul>
Obligātie lauki: spēlētāja dati, pretinieku dati, mainīgie no COM_001
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>

Apstrāde:

- Tiek iestatīts *hp\_gain* – iegūto dzīvības punktu skaits.
- Ja *at\_target* (mērķis) nav ‘none’:
- Tiek aprēķināts *attack\_random* no formulas  $\text{rand}(-3, 3) + \text{rand}(-3,3) + \text{pop}$ .
- Tiek aprēķināts *hit\_chance* no formulas  $\text{pc.tohit} - \text{at\_target.ac}$  (*at\_target* - pretinieks) + *attack\_random* + *mod\_at* + *pop* (precizitātes bonuss).
- Ja *attack\_random* ir mazāka/vienāda ar -4 (~12% iespēja) vai *hit\_chance* ir mazāka par 0, tiek izsaukts netrāpīšanas skats, veikts ieraksts masīvā *cta* un izvadīts attiecīgais teksts.
- Citādi tiek aprēķināts *hit\_piercing* no formulas  $2*(\text{pc.pha} + \text{mod\_dam} + 1) + \text{rand}(-5, 5)$ .
- Ja *hit\_piercing* ir mazāks/vienāds ar 0, tiek izvadīts attiecīgais teksts un veikts ieraksts masīvā *cta*.
- Citādi, ja *dop\_var31* ir lielāks par 0, atkarībā no pretinieka, attiecīgajam pretiniekam tiek palielināts *enemy[num]po* par 2 un samazināts *dop\_var31* par 1.
- Tiek aprēķināts *hp\_taken* no formulas  $\text{hit\_piercing} + \text{pc.str}$  (spēlētāja spēks) + *mod\_str* + *dam* + *dop\_var29*.
- Tiek samazināts *at\_target.hp* (pretinieka dzīvības punkti) par *hp\_taken*.
- Tiek aprēķināts *hp\_gain* no formulas  $\text{hp\_taken}/2.5$ .
- Tiek palielināts *pc.hp* par *hp\_gain*. Ja *pc.hp* pārsniedz *pc.hp1*, tas tiek atiestatīts uz *pc.hp1*.
- Tiek izsaukta pārbaude T2 - ja patiesa, pretinieka *enemy[num]* vērtība iestatīta uz “none” un *loot*, *exp* un *bab* tiek pieskaitītas attiecīgās vērtības.
- Tiek iestatīta nākamā pretinieka *at\_target* vērtība – pretinieku secība ir *enemy2*, *enemy1*, *enemy3* – ja *enemy[num]* nav ‘none’.
- Tiek atkal izsaukta COM\_008.



Attēls 5.3 - COM\_008 bloks shēma

## 5.4. Smilšukastes modulis (sandbox.rpy)

### 5.4.1. Izmaiņas SBX\_001

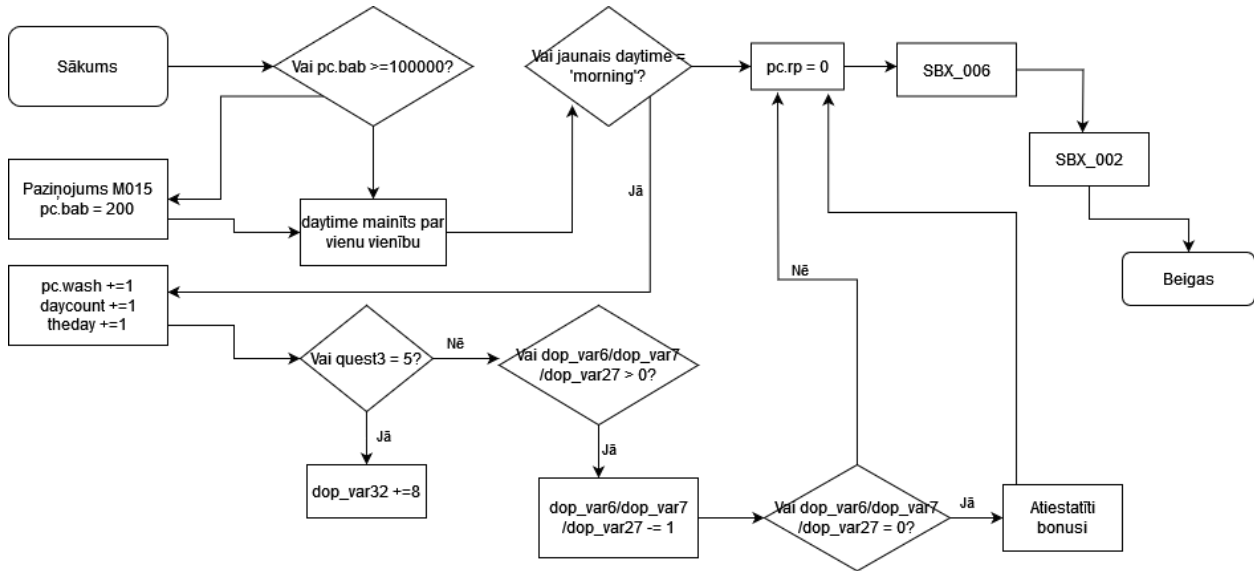
Veiktas sekojošās izmaiņas funkcijā SBX\_001:

Tabula 5.5 - Atjaunināts daļējs projektējums funkcijai "Tīt laiku uz priekšu"

ID: SBX_001
<p>Apstrāde:</p> <ol style="list-style-type: none"><li>1. Ja <i>pc.bab</i> (spēlētāja nauda) ir lielāka/vienāda par 100000 (izmaiņa veikta, jo 10000 ir viegli sasniedzams) vienībām, tiek izvadīts paziņojums M015 un <i>pc.bab</i> atiestatīts uz 200 vienībām.</li><li>2. Ja <i>lost</i> (spēlētāja dzīvības statuss) ir <i>True</i>, tiek izsaukta funkcija SBX_007.</li><li>3. Atkarībā no <i>daytime</i> vērtības tiek mainīta <i>daytime</i> vērtība:<ol style="list-style-type: none"><li>a. Ja <i>daytime</i> ir 'morning' (rīts), <i>daytime</i> tiek mainīts uz 'day' (diena) un atkarībā no <i>quest18</i> (Sāras uzdevuma) vērtības tiek izsaukta atbilstošā funkcija TXT_003.</li><li>b. Ja <i>daytime</i> ir 'day' (diena), <i>daytime</i> tiek mainīts uz 'evening' (vakars).</li><li>c. Ja <i>daytime</i> ir 'evening' (vakars), <i>daytime</i> tiek mainīts uz 'night' (nakts) un atkarībā no <i>quest1</i> (Ilgas uzdevuma) vērtības un <i>theday</i> vērtības (ja <i>theday</i> ir 7, tas ir, festivāla diena) tiek izsaukta funkcija SBX_009.</li><li>d. Ja <i>daytime</i> ir 'night' (nakts), <i>daytime</i> tiek mainīts uz 'morning' (rīts), un:<ol style="list-style-type: none"><li>i. Palielināta spēlētāja netīrība <i>pc.wash</i>.</li><li>ii. Ja <i>quest3</i> <math>\geq 5</math> (iegūta iespēja nopirkt indi no Ilgas), palielināts <i>dop_var32</i> par 8.</li><li>iii. Atkarībā no noteiktu uzdevumu vērtībām tiek vai nu izsaukta funkcija SBX_009, vai izsaukta funkcija TXT_003 vai TXT_004.</li><li>iv. Tiek pieskaitīts 1 <i>theday</i> un <i>daycount</i> vērtībām.</li><li>v. Atkarībā no <i>theday</i> vērtības tiek iestatīta atbilstošā <i>weekd</i> (nedēļas dienas teksta) vērtība. Ja <i>theday</i> ir 8 (nedēļa pabeigta), <i>theday</i> tiek atiestatīts uz 1.</li><li>vi. Ja <i>dop_var6</i>, <i>dop_var7</i> vai <i>dop_var27</i> vērtības ir lielākas par 0, to vērtība tiek samazināta par 1. Ja jebkura no šīm vērtībām sasniedz 0, to dotais bonuss (attiecīgi <i>pc.hp</i> un <i>pc.hp1</i> priekš <i>dop_var6</i> un</li></ol></li></ol></li></ol>

*pc.mad* pārējiem) tiek dzēsts.

4. Tiek atiestatīti niknuma punkti *pc.rp*.
5. Ja *pc.exp* (pieredzes punkti) ir lielāks/vienāds ar 1000, tiek izsaukta funkcija SBX\_006.
6. Ja spēlētāja pašreizējā lokācija *curr\_loc* ir istaba, tiek izsaukta funkcija SBX\_008. Citādi izsauc SBX\_002.



Attēls 5.4 - SBX\_001 blokshēma

### 5.4.2. Jaunas inventāra lietas SBX\_005

Veiktas izmaiņas funkcijā SBX\_005 – pievienotas trīs inventāra lietas:

Tabula 5.6 – jauni pievienotās inventāra lietas

Nosaukums	Paskaidrojums
q_fur	Fūrijas kristāli – nepieciešams <i>quest11</i> jeb alķīmiķa uzdevumam ar fūriju
p_po1	Inde – palielina <i>dop_var31</i> par 4
p_po2	Stipra inde – palielina <i>dop_var31</i> par 7

## 5.5. Skatu modulis (screens.rpy)

### 5.5.1. Izmaiņas VIS\_004 skatā

Izmaiņas funkcijas VIS\_004 skatā:

- Jauna lapa skatā: attiecības ar spēles tēliem. Katram tēlam tiek parādīts *xx\_like*, tas ir, visiem mainīgajiem, kas minēti attiecību mainīgo tabulā, tiek teksta veidā attēlotas vērtības.
- Pievienoti apraksti uzdevumiem *quest3* – Ilgas pagātne – un *quest20* – Forlana bērnības sapņi (iepriekš slēpti uzdevumi), un *quest11* – fūrijas kristālu iegūšana – un *quest14* – meiteņu apmeklēšana bordelī (jaunizveidoti uzdevumi).

Tabula 5.7 – jauni pievienoti uzdevumu mainīgie

Nosaukums	Datu tips	Paskaidrojums	Papildus komentāri
quest11	int	Fūrijas kristāli	Iepriekš neimplementēts – jāiegūst 2 kristāli no fūrijām, naudas balva 200 vienības +80 par katru nākamo kristālu
quest14	int	“Pastaiga pa rožu dārzu”	Iepriekš neimplementēts – jāapmeklē visas meitenes bordelī vismaz vienreiz

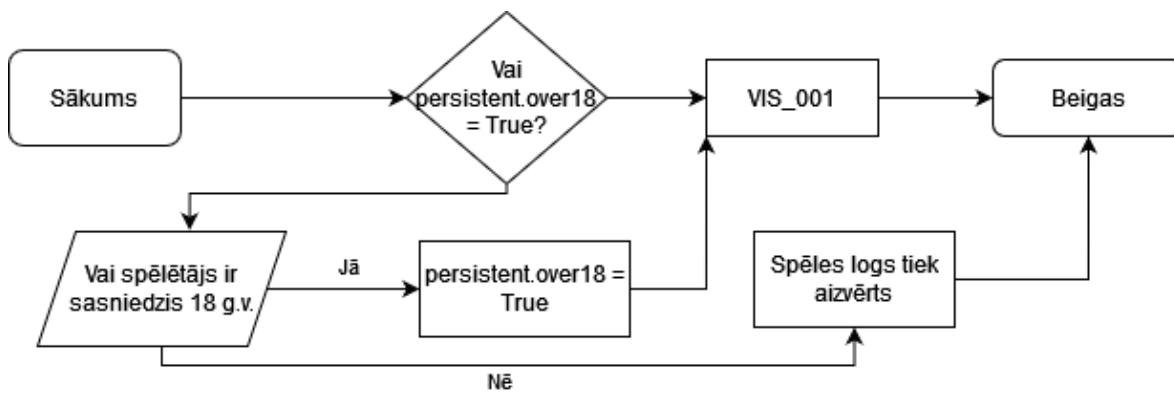
## 5.6. Skripta modulis (script.rpy)

### 5.6.1. Izmaiņas SCR\_001

*persistent* mainīgie ir daļa no globālās atmiņas, kas tiek saglabātas *persistent* failā. Šis fails nav atšifrējams; tam mainīgos var pievienot, deklarējot tos formātā *persistent.[nosaukums]* un piešķirot vērtību. Faila mainīgie var tikt mainīti konsolē un dzēsti, izmantojot komandu *persistent.\_clear()*. Funkcijā, ja spēlētājs apstiprina, ka viņš ir sasniedzis 18 gadu vecumu, tālākās spēles loga atvēršanas reizēs paziņojums M012 netiek parādīts.

Tabula 5.8 - Atjaunināts projektējums funkcijai “Jautāt par spēlētāja vecumu”

ID: SCR_001
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• <i>persistent.over18</i> mainīgais no globālās atmiņas</li></ul>
Obligātie lauki: spēlētāja ievade, <i>persistent.over18</i>
Citas pārbaudes: <ul style="list-style-type: none"><li>• Funkcija tiek izpildīta tikai tad, ja <i>persistent.over18</i> nav True.</li></ul>
Apstrāde: <ul style="list-style-type: none"><li>• Tiek parādīts paziņojums M012.</li><li>• Pie apstiprinošas atbildes tiek izsaukta funkcija VIS_001 un izveidots mainīgais <i>persistent.over18</i> ar vērtību True.</li><li>• Citādi spēles logs tiek aizvērts.</li></ul>



Attēls 5.5 - SCR\_001 blokhēma

## 5.7. Teksta modulis

### 5.7.1. Monstru sasnieguma skaitīšana TXT\_005

Veiktas izmaiņas funkcijā TXT\_005, implementējot skaitītāja monster\_count mainīšanos pēc cīņas.

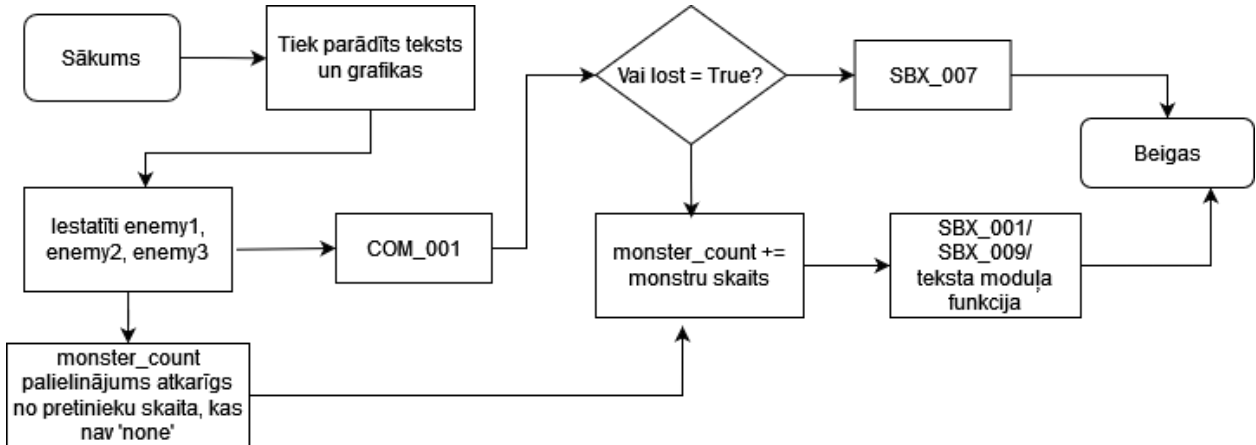
Tā kā neeksistē atsevišķs teksta moduļa .rpy fails, šīs izmaiņas kodā ir implementētas vienā label funkcijā priekšpilsētas modulī un astoņās label funkcijās ārpilsētas modulī.

Tabula 5.9 - Atjaunināts projektējums funkcijai "Cīņas izsaukums"

<b>ID: TXT_005</b>
Ievaddati: <ul style="list-style-type: none"> <li>• Funkcijas nosaukums</li> <li>• Skaitītāja mainīgais monster_count</li> </ul>
Obligātie lauki: funkcijas nosaukums, monster_count
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>

Apstrāde:

- Kodā tiek izsaukta *label* funkcija ar attiecīgo nosaukumu.
- Tiek parādīts atbilstošais teksts un grafikas.
- Tiek iestatīti pretinieku *enemy1*, *enemy2*, *enemy3* mainīgie un izsaukta funkcija COM\_001.
- Ja *lost* ir *True*, tiek izsaukta funkcija SBX\_007.
- Ja *lost* ir *False* un *dop\_var4* ir 0, *monster\_count* tiek palielināts attiecīgi par noteikto nogalināto monsturu skaitu.
- Citādi var tikt izsaukta funkcija SBX\_009, un tiek izsaukta vai nu funkcija SBX\_001, vai teksta moduļa funkcija ar attiecīgo, kodā minēto nosaukumu.



Attēls 5.6 - TXT\_005 blokshēma

## 5.8. Galvenais modulis (loc.rpy)

### 5.8.1. Sasnieguma iegūšana LOCM\_001

Pievienota jaunā sasnieguma iegūšana LOCM\_001.

Tabula 5.10 - Atjaunināts projektējums funkcijai “Rīta funkcija”

**ID: LOCM\_001**

Ievaddati:

- Spēlētāja dati no lokālās datu krātuves
- Uzdevumu dati no lokālās datu krātuves

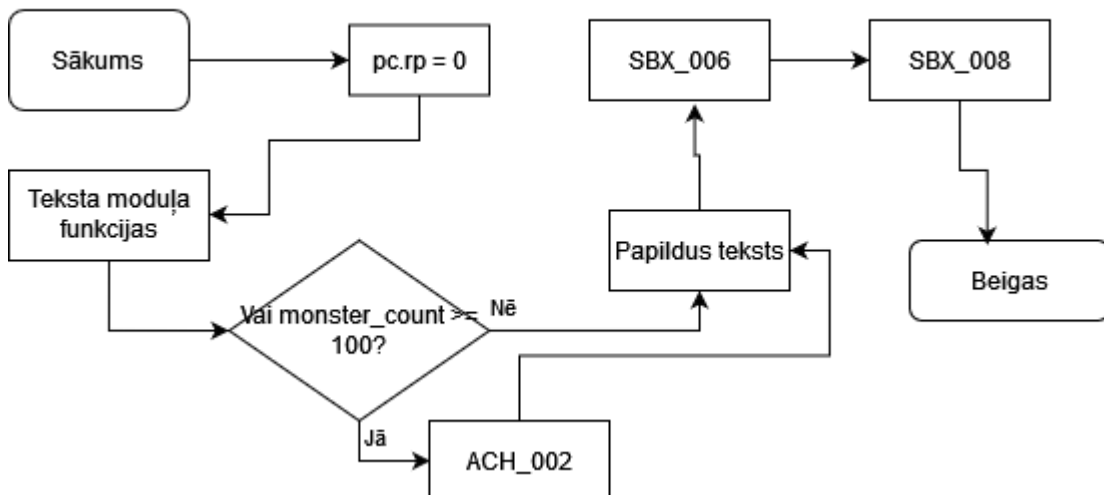
Obligātie lauki: spēlētāja dati, uzdevumu dati

Citas pārbaudes:

- –

Apstrāde:

- Tiek atiestatīti *pc.rp* uz 0 un *pc.hp* uz *pc.hp1*.
- Atkarībā no noteiktu uzdevumu vērtībām tiek izsauktas noteiktas tekstu moduļu funkcijas un 3. un 4.punkts izlaists.
- Atkarībā no *quest5* (Kilenas uzd.) vērtības tiek izsaukta noteiktā teksta moduļa funkcija.
- Atkarībā no citu uzdevumu vērtībām tiek izsaukti sapņi – TXT\_001 funkcijas.
- Ja *monster\_count* ir lielāks/vienāds ar 100, tiek izsaukta funkcija ACH\_002 ar sasnieguma nosaukumu 'killer'.
- Atkarībā no *quest5* vērtības tiek izsaukta noteikta TXT\_003 funkcija.
- Tiek aprēķināts nejaušs mainīgais. Atkarībā no mainīgā vērtības tiek parādīts atšķirīgs teksts un grafikas.
- Tiek izsaukta funkcija SBX\_006.
- Tiek izsaukta funkcija SBX\_008.



Attēls 5.7 - LOCM\_001 blokshēma

### 5.8.2. Mrisjas patikšana LOCM\_005

Veiktas izmaiņas Mrisjas patikšanas mainīgā izmaiņās LOCM\_005.

Tabula 5.11 - Atjaunināts projektējums funkcijai "Apmeklēt krogu"

ID: LOCM_005
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> </ul>
Obligātie lauki: spēlētāja dati, uzdevumu dati
Citas pārbaudes: <ul style="list-style-type: none"> <li>• Ja <math>pc.wash &gt; 3</math>, tiek izsaukta funkcija SBX_002 un funkcijas darbība netiek turpināta.</li> </ul>
Apstrāde: <ul style="list-style-type: none"> <li>• Tiek aprēķināts nejaušs mainīgais. Atkarībā no mainīgā vērtības var tikt izsaukta noteikta TXT_002 funkcija.</li> <li>• Citādi parādās izvēlne. Galvenās divas opcijas ir pasūtīt ēdienu un iet uz istabu (izsauc SBX_008). Citas opcijas parādās, atkarībā no uzdevumu vērtībām, kas</li> </ul>

izsauc atbilstošās teksta moduļa funkcijas.

- Ja izvēlas pasūtīt ēdienu, *pc.bab* tiek samazināts par 5. Tālākās izvēlnes atkarīgas no *mr\_like*.
  - Ja *mr\_like*  $\leq 3$ , var samaksāt dzeramnaudu – 1 vienību (*pc.bab*-1), kas palielina *mr\_like* par 1, vai tikai pateikt paldies (izsauc TXT\_001).
  - Ja  $3 < mr\_like \leq 6$ , var samaksāt vai nu 1, vai 5 naudas vienības dzeramnaudā; *mr\_like* palielinās tikai tad, ja samaksā 5 vienības.
  - Ja  $7 \leq mr\_like \leq 10$ , var samaksāt vai nu 5, vai 10 naudas vienības dzeramnaudā; *mr\_like* palielinās tikai tad, ja samaksā 10 vienības. Var arī tikai pateikt paldies (dzēsta patikšanas samazināšanās).
  - Ja *mr\_like*  $> 10$ , var samaksāt 15 naudas vienības dzeramnaudā, kas palielina *mr\_like* par 1, vai tikai pateikt paldies (dzēsta patikšanas samazināšanās).
- Tiek parādīta izvēlne, vai nu tikai atpūsties, vai klausīties klientus.
  - Ja izvēle ir atpūsties, tiek aprēķināts nejaušs mainīgais, un atkarībā no šī mainīgā tiek parādīts atšķirīgs teksts un grafikas.
  - Ja izvēle ir klausīties, tiek aprēķināts nejaušs mainīgais, un atkarībā no šī mainīgā, ja *quest8*, *quest9* vai *quest10* vērtības ir 0, katram uzdevumam ir 33% iespēja tikt uzsāktam – tiek izsaukta funkcija SBX\_009 ar attiecīgo uzdevuma mainīgo un vērtību 1.
- Tiek izsaukta funkcija SBX\_001.

### 5.8.3. Uzdevuma quest20 pabeigšanas implementēšana

Implementēta *quest20* pabeigšana: noteiktā TXT\_006 funkcijā (*quest20\_dreams*), ja *quest20* = 1 un visi ar *quest20* saistītie mainīgie ir *True*, tiek izsaukta funkcija SBX\_009 ar *quest20* un posma vērtību 2.

## 5.9. Centra modulis (loc\_center.rpy)

### 5.9.1. Zobens uzlabojumi LOCC\_002

Izlabotas kļūdas zobens uzlabojumu iegūšanā:

- Pievienota iespēja iegādāties uzlabojumu ‘*rune*’.
- Labota iepriekš minētā uzlabojuma cena.
- Vērtība ‘*rune*’ piešķirta pareizajam mainīgajam – *pc.sk1*.

### 5.9.2. Jauns uzdevums LOCC\_001

Tiek pievienota iespēja iegūt un pabeigt no jauna izveidoto uzdevumu *quest11*. Šim tika implementētas divas jaunas ainas.

Tabula 5.12 - Atjaunināts projektējums funkcijai “*Tirgoties ar alķīmiķi*”

ID: LOCC_001
Ievaddati: <ul style="list-style-type: none"><li>• Spēlētāja ievade</li><li>• Spēlētāja dati no lokālās datu krātuves</li><li>• Uzdevumu dati no lokālās datu krātuves</li></ul>

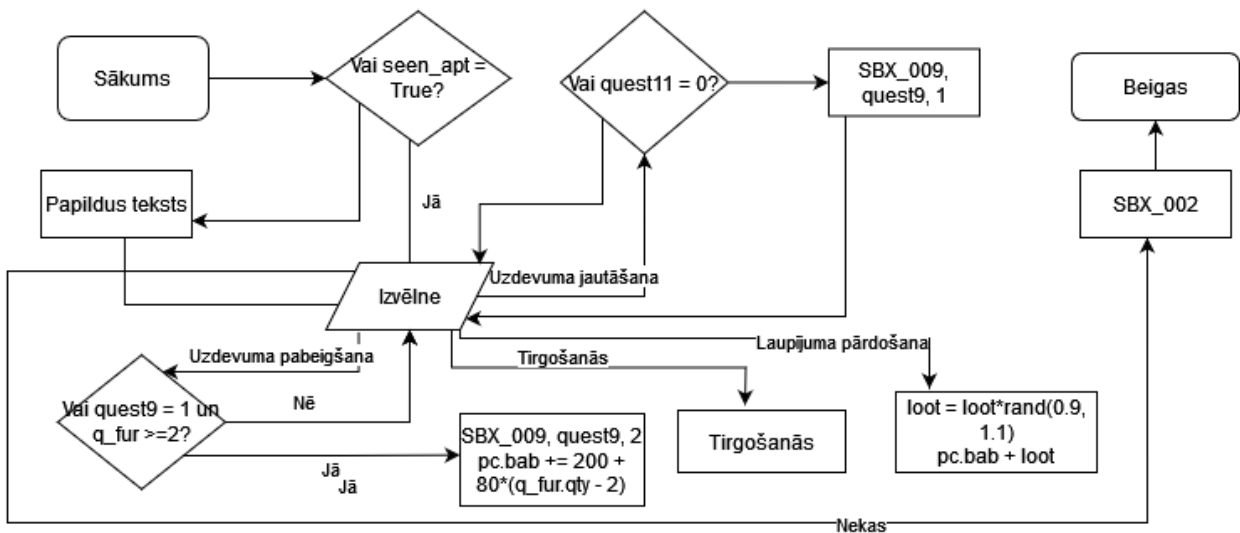
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati

Citas pārbaudes:

- –

Apstrāde:

- Ja *seen\_apt* ir *False*, tas tiek iestatīts kā *True* un tiek parādīts papildus teksts.
- Tiek parādīta izvēlne:
  - Noteiktas opcijas ir tikai pieejamas, atkarībā no uzdevuma *m\_quest2*, *quest9*, *quest11* vai *quest24* vērtībām.
    - Ja *quest11* vērtība ir 0, var alķīmiķim jautāt par uzdevuma iegūšanu. Ja piekrīt, tiek izsaukta funkcija *SBX\_009* ar uzdevumu *quest9* un vērtību 1.
    - Ja *quest11* vērtība ir 1 un *q\_fur.qty* ir vismaz 2, var pabeigt *quest11 – pc.bab* palielinās par  $200 + 80 * (q\_fur.qty - 2)$ .
  - Tirgošanās: tiek parādīta izvēlne ar lietām, kuras alķīmiķis pārdod. Izvēloties vienu no šīm lietām, tiek samazināts *pc.bab* par noteikto daudzumu un palielināts dziras daudzuma mainīgais par 1.
  - Laupījuma pārdošana: ja *loot* > 0, var pārdot *loot – pc.bab* tiek palielināts ar *loot\*x*, kur *x* ir robežās no 0.9 līdz 1.1. *loot* tiek atiestatīts.
  - Pēdējā izvēle izsauc funkciju *SBX\_002*.



Attēls 5.8 - LOCC\_001 blokshēma

### 5.9.3. Kļūdu labojumi centra modulī

Veikti kļūdu labojumi TXT\_004 funkcijā *m\_quest5\_1* un TXT\_006 funkcijā *holiday\_kil\_dance\_grope*.

- *bur\_like* un *pc.rep* palielinās tikai tad, ja Forlans klusi piekrīt mēra uzdevumam, neprasot par to atlīdzību. Iepriekš šie mainīgie palielinājās, ja Forlans prasa par savu darbu atlīdzību, kas loģiski stāsta kontekstā nav pareizi.
- *ki\_like* pēc darbības palielinās tikai tad, ja *ki\_like* ir vairāk par 5 – iepriekš bija otrādi, kas nebija loģiski.

## 5.10. Priekšpilsētas modulis (loc\_okr.rpy)

### 5.10.1. Uzdevuma quest14 implementācija – LOCK\_001

Tiek implementēts quest14 – uzdevums, kurā jāapmeklē visas meitenes bordelī.

Tabula 5.13 - Atjaunināts projektējums funkcijai “Apmeklēt bordeli”

**ID: LOCK\_001**

Ievaddati:

- Spēlētāja ievade
- Spēlētāja dati no lokālās datu krātuves
- Uzdevumu dati no lokālās datu krātuves

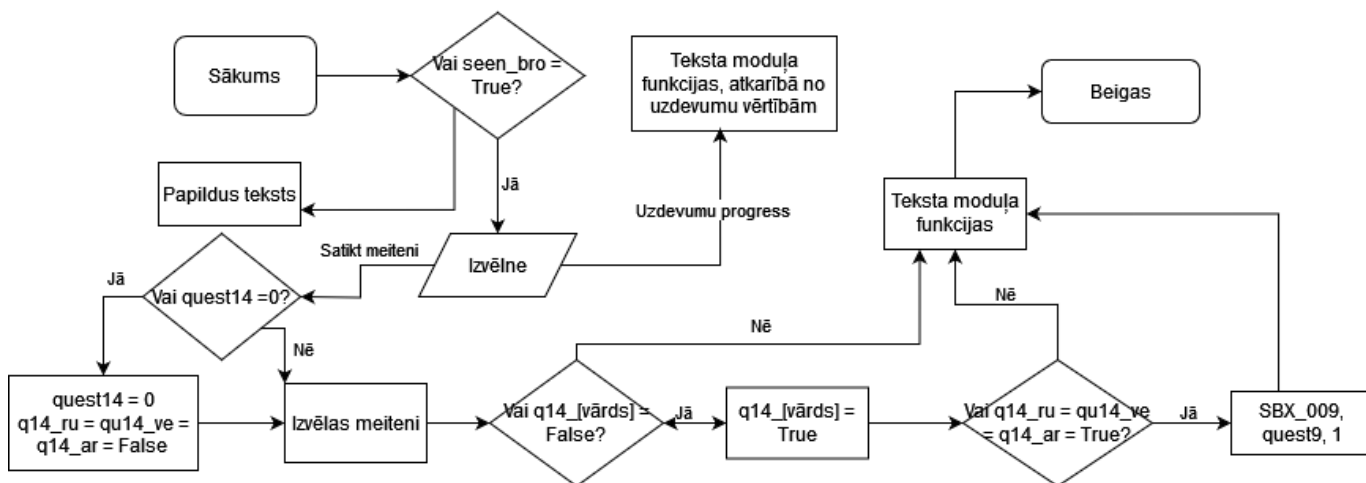
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati

Citas pārbaudes:

- Ja *pc.wash* > 3, tiek izsaukta funkcija SBX\_002 un funkcijas darbība netiek turpināta.

Apstrāde:

- Ja *seen\_bro* ir *False*, tas tiek mainīts uz *True* un tiek parādīts papildus teksts.
- Ja *m\_quest3* ir 4, tiek izsaukta atbilstošā TXT\_004 funkcija.
- Citādi tiek parādīta izvēlne:
  - Noteiktas opcijas ir tikai pieejamas, atkarībā no uzdevuma *m\_quest2*, *m\_quest3* vai *quest2* vērtībām.
  - Satikt meiteni – tiek parādīta izvēlne ar meitenēm. Ja *quest14* ir 0, tiek izsaukta funkcija SBX\_009. Izvēloties meiteni, ja *quest14* ir 1, attiecīgajai meitenei tiek atzīmēts, ka viņa ir apmeklēta (ja jau nav bijusi apmeklēta), tas ir, tiek iestatīta *q14 [vārds]* vērtība kā *True*, tiek samazināts *pc.bab* par noteikto daudzumu un palielināts meiteņu tikšanās reižu skaitītājs. Kad skaitītājs sasniedz 10, tiek izsaukta funkcija ACH\_002 ar sasnieguma nosaukumu 'zavs'.
  - Ja visu meiteņu *q14 [vārds]* ir *True* un *quest14* vērtība ir 1 (uzdevums vēl nav pabeigts), tiek izsaukta funkcija SBX\_009 ar uzdevumu *quest20* un vērtību 2.
- Tiek izsaukta funkcija SBX\_001.



Attēls 5.9 - LOCK\_001 blokshēma

### 5.10.2. Indes nopirkšana no Ilgas – LOCK\_002

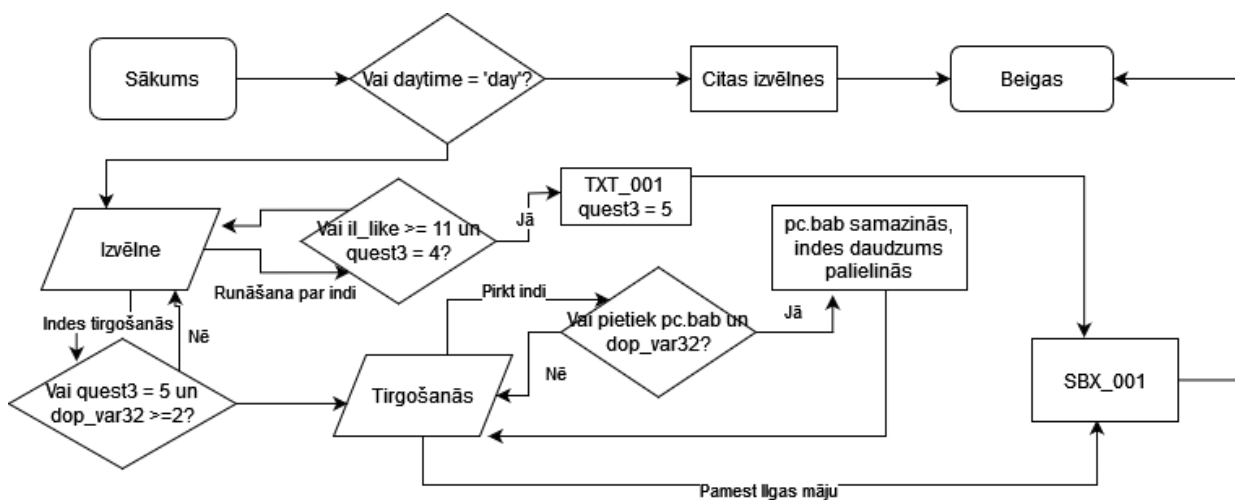
Veiktas būtiskas izmaiņas LOCK\_002. Indes iegādei no Ilgas tika implementētas trīs jaunas ainas, papildus LOCK\_002 izmaiņām.

Tabula 5.14 - Atjaunināts projektējums funkcijai “Apmeklēt Ilgas māju”

ID: LOCK_002
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja ievade</li> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> <li>• Dienas laiks <i>daytime</i></li> </ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati, <i>daytime</i>
Citas pārbaudes: <ul style="list-style-type: none"> <li>• Ja <i>pc.wash</i> &gt; 3, tiek izsaukta funkcija SBX_002 un funkcijas darbība netiek turpināta.</li> </ul>

Apstrāde:

1. Ja *daytime* ir 'morning':
  - a. Ja *quest1* ir 4 vai 6, tiek izsaukta atbilstošās teksta moduļa funkcijas.
  - b. Ja *quest16* ir 2, 8 vai 9, tiek izsaukta atbilstošā TXT\_001 vai TXT\_004 funkcija.
  - c. Citādi tiek izsaukta funkcija SBX\_002.
2. Ja *daytime* ir 'day' vai 'evening':
  - a. Ja *daytime* ir 'day' un *dop\_var17* ir 0, tiek izsaukta TXT\_002 funkcija un *dop\_var17* iestatīts kā 1.
  - b. Citādi tiek izsaukta izvēlne – izvēlnē pieejamās opcijas ir atkarīgas no *il\_help*, *il\_like* un uzdevumu *m\_quest2*, *quest1*, *quest3*, *quest16* un *quest24* vērtībām.
  - c. Ja opcija nav saistīta ar *m\_quest2* vai *quest24*, tad ir iespēja mainīt *il\_like*.
    - i. Ja opcija ir saistīta ar *quest1* vai *quest3*, palielinās *il\_like* par 1 veiksmīgu atbilžu sniegšanas gadījumā, izsauktas atbilstošās teksta moduļa funkcijas un izsaukta funkcija SBX\_009 ar attiecīgajiem parametriem.
    - ii. Ja opcija ir saistīta ar *quest16*, *il\_like* samazinās par 1 un tiek izsaukta atbilstošā TXT\_004 funkcija.
  - d. Ja opcija ir saistīta ar *quest3* (iespējo indes nopirkšanu):
    - i. Pirmā opcija ir pieejama, ja *il\_like*  $\geq$  11 un *quest3* vērtība ir 4. Tiek izsaukta TXT\_004 funkcija, kur atkarībā no dotās atbildes *il\_like* +/-1, un kur *quest3* vērtība iestatīta kā 5.
    - ii. Otrā opcija ir pieejama, ja *quest3* ir 5 un *dop\_var32* ir lielāks/vienāds ar 2. Tiek parādīta izvēlne, kuru indi iegādāties, kā arī izvēle beigt tirgoties, kas izsauc SBX\_001. Var iegādāties *p\_po1* par 40 naudas vienībām un 2 *dop\_var32* punktiem, vai *p\_po2* par 60 naudas vienībām un 3 *dop\_var32* punktiem. Attiecīgi palielinās lietas daudzums jeb *qty* par 1.
  - e. Tiek izsaukta funkcija SBX\_001.
3. Ja *daytime* ir 'night', tiek izsaukta funkcija SBX\_002.



Attēls 5.10 - LOCK\_002 daļēja blokshēma

### 5.10.3. AINU iespējamību izmaiņas – LOCK\_003

Tiek mainītas ainu iespējamības LOCK\_003, lielākoties izdzēšot iespējamību neko neredzēt (SBX\_001 izsaukums) un palielinot to ainu iespējamības, kas palielina uzdevumu vērtības.

Tabula 5.15 - Atjaunināts projektējums funkcijai “Pastaigāties pa pilsētu”

ID: LOCK_003
Ievaddati: <ul style="list-style-type: none"> <li>• Spēlētāja ievade</li> <li>• Spēlētāja dati no lokālās datu krātuves</li> <li>• Uzdevumu dati no lokālās datu krātuves</li> <li>• Dienas laiks <i>daytime</i></li> </ul>
Obligātie lauki: spēlētāja ievade, spēlētāja dati, uzdevumu dati, <i>daytime</i>
Citas pārbaudes: <ul style="list-style-type: none"> <li>• –</li> </ul>

Apstrāde:

1. Tiek aprēķināts nejaušs mainīgais.
2. Ja *daytime* ir 'morning':
  - a. 65% iespēja izsaukt TXT\_001 funkciju, kur *pc.bab* palielinās par 5 vai 10;
  - b. 35% iespēja izsaukt TXT\_002 funkciju, kur pēc izvēles var izsaukt TXT\_005 funkciju.
3. Ja *daytime* ir 'day':
  - a. Ja *quest17* ir 2, tiek palielināts *dop\_var19* par 1 – kad *dop\_var19* ir 4, tiek izsaukta TXT\_004 funkcija.
  - b. Ja *quest17* ir 3, tiek izsaukta TXT\_004 funkcija.
  - c. Citādi:
    - i. 30% iespēja izsaukt TXT\_001 funkciju, kur *pc.bab* palielinās par 5 vai 10;
    - ii. 10% iespēja izsaukt TXT\_001 funkciju ar pāri;
    - iii. 60% iespēja izsaukt TXT\_002 funkciju ar Kilenu.
4. Ja *daytime* ir 'evening':
  - a. 10% iespēja izsaukt TXT\_001 funkciju, kur *pc.bab* palielinās par 5 vai 10;
  - b. 50% iespēja izsaukt TXT\_005 funkciju, kas izsauc SBX\_009 atkarībā no *quest8* vērtības;
  - c. 5% iespēja redzēt TXT\_001 funkciju ar pāri;
  - d. 35% iespēja izsaukt TXT\_002 funkciju, kas tālāk ar 50/50 iespējamību izsauc vai nu TXT\_001, vai TXT\_005 funkciju.
5. Ja *daytime* ir 'night':
  - a. 80% iespēja izsaukt 4.b. punktā minētu funkciju.
  - b. 20% iespēja izsaukt SBX\_001.

## 5.11. Ārpilsētas modulis (loc\_outside.rpy)

### 5.11.1. Fūrijas uzbrukums – quest11

Implementēta `q_fur.qty` palielināšanās par 1 pēc katras cīņas ar fūriju, ja `lost = False` (nav miris cīņā), `dop_var4 = 0` (nav bēdzis no cīņas) un `quest11 = 1` (uzdevums iesākts).

Pie reizes, tāpat kā visām `COM_001` izsaucošām funkcijām ārpusētas moduļi, implementēta `monster_count` palielināšanās pēc katras sekmīgas cīņas.

## 5.12. Lokālā datu krātuve – vari.rpy

### 5.12.1. Mainīgo pievienošana

Pievienoti sekojošie mainīgie:

Tabula 5.16 – pievienoti funkcionālie mainīgie

Nosaukums	Datu tips	Paskaidrojums
<code>monster_count</code>	int	Skaita, cik monstri nogalināti – nepieciešams jaunajai sistēmai
<code>dop_var31</code>	int	Cik indes devu ir uz zobens
<code>dop_var32</code>	int	Indes punkti – var apmainīt pret indi no Ilgas

### 5.12.2. Indes lietošanas funkcija

Implementētas divas funkcionāli identiskas funkcijas, kas saistītas ar indes lietošanu:

1. Palielina `dop_var31` par attiecīgo daudzumu (4, ja `p_po1`, 7, ja `p_po2`)
2. Samazina `p_po1.qty/p_po2.qty` par 1.

## 6. TESTĒŠANAS DOKUMENTĀCIJA

### 6.1. Testēšanas apraksts

Šajā nodaļā aprakstīta sistēmā veikto izmaiņu vienībtestēšana. Testēšana veikta manuāli gan izmaiņu izstrādes procesā, gan pēc izstrādes pabeigšanas. Testēšana sadalīta pa moduļiem un pa izmaiņām, kas tika aprakstītas iepriekšējā nodaļā. Dažām no izmaiņām bija nepieciešami vairāki vienībtesti, lai pārliecinātos, ka izmaiņas ir funkcionālas.

Operētājsistēma: Windows 10 Home 22H2 19045.3803

Sistēmas oriģinālā versija: 1.0.41.

### 6.2. Testēšanas žurnāls

Tabula 6.1 – izmaiņu testēšanas žurnāls

Modulis	Izmaiņa	Testa apraksts	Sagaidāmais rezultāts	Statuss
Sasniegumu modulis	Jauns sasnieguma apraksts	Sasnieguma nosaukuma un apraksta iestatīšana	Tiek veiksmīgi iestatīts sasnieguma nosaukums un apraksts, atverot spēles logu	Sekmīgs
Cīņu modulis	<i>pc.exp</i> palielināšanās	Pieredzes punktu palielināšanās pēc jaunās formulas ( <i>exp</i> = 200, <i>pc.lev</i> = 1)	<i>pc.exp</i> palielinās par 400 punktiem.	Sekmīgs
	<i>pc.bab</i> palielināšanās	Naudas palielināšanās pēc jaunās formulas ( <i>bab</i> = 20)	<i>pc.bab</i> palielinās par 40 vienībām.	Sekmīgs

Modulis	Izmaiņa	Testa apraksts	Sagaidāmais rezultāts	Statuss
	Glābšanas uzbrukuma izsaukums COM_003	COM_008 izsaukums, ja tas ir iespējams ( $pc.rp = pc.rp1$ )	$pc.rp$ atiestatīts uz 0, pirmais pretinieks – $enemy2$ , ja eksistē (citādi $enemy1/enemy3$ )	Sekmīgs
	Indes efekti COM_003	$enemy[num].hp$ samazināšanās, ja $enemy[num]po > 0$	Pēc spēlētāja darbības veikšanas $enemy[num].hp$ samazinās un tiek veikts ieraksts cīņas vēsturē	Sekmīgs
	Indes efektu izraisīšana COM_006	Pretinieka $enemy[num]$ saindēšanās, ja $dop\_var31 > 0$	$enemy[num]po$ palielinās par 2	Sekmīgs
	Uzbrukšana visiem pretiniekiem COM_008	Tiek uzbrukts visiem pretiniekiem	Samazinās visu pretinieku $hp$ (ja veiksmīgs uzbrukums) un tiek veikts attiecīgais ieraksts cīņas vēsturē	Sekmīgs
	Indes efektu izraisīšana COM_008	Visu pretinieku saindēšanās, ja $dop\_var31 \geq 3$ uzbrukumā ar 3 pretiniekiem	$enemy1po$ , $enemy2po$ , $enemy3po$ palielinās par 2	Sekmīgs
	$pc.hp$ palielināšanās pēc COM_008	Spēlētāja $pc.hp$ palielināšanās pēc sekmīga COM_008 uzbrukuma	$pc.hp$ palielinās par pretiniekiem atņemtajiem dzīvības punktiem, dalītiem ar 2,5	Sekmīgs

Modulis	Izmaiņa	Testa apraksts	Sagaidāmais rezultāts	Statuss
Smilšukastes modulis	Pretkrāpšanās mehānisms SBX_001	Vai ir iespējams pārsniegt 10000 naudas vienības, neizsaucot šo mehānismu ( <i>pc.bab</i> = 10001)	Paziņojums M015 netiek izsaukts	Sekmīgs
	Indes punktu palielināšanās SBX_001	Ja <i>quest3</i> = 5, <i>dop_var32</i> jāpalielinās par 8 pēc katras dienas beigām	<i>dop_var32</i> palielinās par 8	Sekmīgs
	Jaunās inventāra lietas SBX_005	Jauni izveidotās inventāra lietas inventāra skatā, ja to <i>qty</i> > 0	Skatā parādās <i>q_fur</i> , <i>p_po1</i> un <i>p_po2</i> , ja to attiecīgie <i>qty</i> > 0	Sekmīgs
Skatu modulis	Uzdevumu apraksti	Jauni izveidoto aprakstu parādīšanās VIS_004 skatā	Skatā parādās atbilstošais apraksts, balstoties uz <i>quest3</i> , <i>quest11</i> , <i>quest14</i> un <i>quest20</i> vērtībām	Sekmīgs
	Attiecību lapa VIS_004 skatā	Jauni izveidotās attiecību lapas parādīšanās VIS_004 skatā	Skatā parādās jauna lapa, kur ir teksta veidā uzskaitīti visi attiecību mainīgie	Sekmīgs
Skripta modulis	<i>persistent.over18</i> iestatīšanās	<i>persistent.over18</i> iestatīts kā <i>True</i> , ja apstiprina, ka ir sasniegts 18 gadu vecums	Nākamo reizi atverot spēles logu, paziņojums M012 neparādās	Sekmīgs

Modulis	Izmaiņa	Testa apraksts	Sagaidāmais rezultāts	Statuss
Teksta modulis	<i>monster_count</i> uzskaitē TXT_005	<i>monster_count</i> palielināšanās, ja tika izsaukta cīņa ar monstriem (pieņemts, ka viens pretinieks)	Pēc sekmīgas cīņas <i>monster_count</i> palielinās par 1	Sekmīgs
Galvenais modulis	Sasnieguma iegūšana LOCM_001	Ja <i>monster_count</i> $\geq$ 100, tiek izsaukta funkcija ACH_002 ar attiecīgo sasnieguma nosaukumu	Tiek izsaukts ACH_002 ar paziņojumu M001	Sekmīgs
	Mrisjas patikšanas izmaiņas LOCM_005	Ja <i>mr_like</i> $\geq$ 7, pasakot tikai 'paldies', <i>mr_like</i> nesamazinās	Pasakot 'paldies', <i>mr_like</i> nemainās	Sekmīgs
	Uzdevuma quest20 pabeigšana	Uzdevums <i>quest20</i> tiek pabeigts, ja visi uzdevuma sapņi ir redzēti un veiksmīgi pabeigti (sapņu mainīgie ir <i>True</i> )	Tiek izsaukta funkcija SBX_009 ar uzdevumu <i>quest20</i> un vērtību 2	Sekmīgs
Centra modulis	Uzdevums <i>quest11</i> pie alkīmiķa	Ir iespējams iegūt <i>quest11</i> pie alkīmiķa, ja tas nav uzsākts	Tiek izsaukta funkcija SBX_009 ar uzdevumu <i>quest11</i> un vērtību 1	Sekmīgs
		Ja <i>quest11</i> ir iesākts un <i>q_fur.qty</i> $\geq$ 2, uzdevumu ir iespējams pabeigt	<i>pc.bab</i> palielinās par $200+80*(q\_fur.qty - 2)$ , tiek izsaukta funkcija SBX_009 ar uzdevumu <i>quest11</i> un vērtību 2	Sekmīgs

Modulis	Izmaiņa	Testa apraksts	Sagaidāmais rezultāts	Statuss
	Kļūdu labojums pie mēra	<i>pc.rep</i> un <i>bur_like</i> palielinās pie loģiski pareizās izvēles	<i>pc.rep</i> un <i>bur_like</i> palielinās par 1	Sekmīgs
	Kļūdu labojums ar Kilenu	<i>ki_like</i> palielinās, ja <i>ki_like</i> > 5	<i>ki_like</i> palielinās par 1	Sekmīgs
Priekšpilsētas modulis	Uzdevuma <i>quest14</i> implementācija LOCK_001	Ja nav uzsākts <i>quest14</i> , tas tiek uzsākts, pirmo reizi jautājot par meiteņu apmeklēšanu	Tiek izsaukta funkcija SBX_009 ar uzdevumu <i>quest14</i> un vērtību 1	Sekmīgs
		Ja ir uzsākts <i>quest14</i> , apmeklējot katru meiteni pirmo reizi, tas tiek atzīmēts un mainās <i>quest14</i> apraksts	Tiek atzīmēts attiecīgais <i>q14 [vārds]</i> kā True un mainās <i>quest14</i> apraksts VIS_004 skatā	Sekmīgs
	Indes iegāde no Ilgas LOCK_002	Ja <i>il_like</i> >= 11 un <i>quest3</i> = 4, parādās izvēle runāt par indi	Tiek izsaukta attiecīgā TXT_004 funkcija, tiek mainīta <i>quest3</i> vērtība uz 5	Sekmīgs
		Ja <i>quest3</i> = 5 un <i>dop_var32</i> >= 2, parādās iespēja nopirkt indi	Atkarībā no izvēles palielinās <i>p_po1.qty</i> vai <i>p_po2.qty</i> par 1, samazinās <i>pc.bab</i> un <i>dop_var32</i>	Sekmīgs
		Ja nepietiek <i>dop_var32</i> punktu tālākai pirkšanai, opcija netiek parādīta ( <i>dop_var32</i> = 2)	Netiek parādīta opcija nopirkt <i>p_po2</i> (maksā 3 <i>dop_var32</i> punktus)	Sekmīgs

Modulis	Izmaiņa	Testa apraksts	Sagaidāmais rezultāts	Statuss
	Ainu iespējamību izmaiņas LOCK_003	Konsolē tiek mainīta <i>walk_ran</i> vērtība	Atkarībā no <i>walk_ran</i> vērtības tiek izsaukta attiecīgā teksta moduļa funkcija	Sekmīgs
Ārpilsētas modulis	Fūrijas uzbrukums <i>quest11</i>	Ja sekmīga cīņa ar fūriju, jāpalielinās <i>q_fur.qty</i> (ja <i>quest11</i> = 1) un <i>monster_count</i>	<i>q_fur.qty</i> un <i>monster_count</i> palielinās par 1	Sekmīgs
Lokālā datu krātuve	Indes lietošanas funkcijas	Ja SBX_005 skatā izvēlas lietot indi, tiek izsaukta indes lietošanas funkcija	Ja <i>p_po1: dop_var31</i> palielinās par 4, <i>p_po1.qty</i> samazinās par 1 Ja <i>p_po2: dop_var31</i> palielinās par 7, <i>p_po2.qty</i> samazinās par 1	Sekmīgs

## 7. PROJEKTA PĀRVALDĪBA

### 7.1. Projekta organizācija

Kvalifikācijas darbā ietvertais projekts tika veikts patstāvīgi. Izstrāde tika veikta, izmantojot spējās programmatūras izstrādes dzīves cikla modeli – sākumā tika veikta iepazīšanās ar Ren'Py dzinēju, iepazīšanās ar darbā aprakstīto sistēmu, un funkciju dokumentēšana. Balstoties uz šo izpēti, tika projektētas darbā veiktās izmaiņas, kuras pēc tam tika pakāpeniski testētas un papildinātas, līdz darbs sasniedza pieņemamu pabeigtības stāvokli.

### 7.2. Kvalitātes nodrošināšana

Kods – gan pētītais, gan izveidotais – tika detalizēti komentēts, lai izprastu oriģinālo sistēmu un paskaidrotu visas veiktās izmaiņas. Nepārtrauktības nolūkos tika ievērots līdzīgs koda rakstīšanas stils oriģinālajiem autoriem, bet, tā kā oriģinālais kods nebija komentēts, komentārus izveidoja šī darba autors.

Pēc katra izmaiņas soļa implementēšanas, tā kā izmaiņu uzskaitē pieminētās funkcijas varēja būt atkarīgas no vairākām funkcijām vienā modulī vai pat vairākiem moduļiem, sistēmas darbība tika testēta, lai pārliecinātos par funkcionalitātes pareizu darbību un novērstu potenciāli letālu kļūdu veidošanos un izplatīšanos.

### 7.3. Konfigurāciju nodrošināšana

Projekta konfigurācijas, repozitorija un nākamo versiju pārvaldei tika izmantots versiju kontroles rīks Git, kas bija savienots ar repozitoriju tīmekļa vietnē GitHub.

Repozitorijā tika ievietoti sistēmas galvenie faili kopā ar failu tulkojumiem (bez grafikām), vēlāk pievienoti komentāri visam kodam, un inkrementāli pievienotas izmaiņas repozitorijam.

Repozitorija kods tika rediģēts, izmantojot koda redaktoru Visual Studio Code.

## 7.4. Darbietilpības novērtējums

Darbietilpības novērtējumu var apskatīt no diviem skatupunktiem.

1. Darbietilpības novērtējums dienu skaitā;
2. Darbietilpības novērtējums, balstoties uz rindiņu skaitu.

### 7.4.1. Darbietilpības novērtējums dienu skaitā

Tālāk aprakstītājā tabulā tiek veikts darbietilpības novērtējums, balstoties uz to, cik dienas katrs uzdevums prasīja.

Rezultāts tiek iegūts no PERT formulas:

$$R = \frac{\text{optimistiskais} + 4 * \text{reālistiskais} + \text{pesimistiskais}}{6} \text{. [18]}$$

Tabula 7.1 – darbietilpības novērtējums dienās

Uzdevums	Novērtējums (dienās)			
	Optimistiskais	Reālistiskais	Pesimistiskais	Rezultāts
Iepazīšanās ar Ren'Py dzinēju un failu struktūru	2	3	5	3,16
Iepazīšanās ar sistēmu	5	7	10	7,16
Koda piekļuve un repozitorija izveidošana	0,5	1	2	1,08
Koda izpēte un komentāru veikšana	15	20	25	20
Izmaiņu projektēšana	2	3	5	3,16

Uzdevums	Novērtējums (dienās)			
	Optimistiskais	Reālistiskais	Pesimistiskais	Rezultāts
Izmaiņu implementēšana un testēšana	7	10	14	10,16
Dokumentācijas izveide	10	15	25	15,83
Kopā:	41,5	59	86	60,55

Var secināt, ka pēc autora novērtējuma darbs prasīja 60,55 persondienas. Personmēnesī ir aptuveni 20 persondienas, tātad pēc šī novērtējuma darbs prasīja aptuveni 3,02 personmēnešus.

#### 7.4.2. Darbietilpības novērtējums pēc koda rindiņu skaita

Darbietilpību var arī novērtēt, balstoties uz QSM vēsturiskajiem datiem par rindu skaitu Quartile 1 projektos. Šādu projektu mediāna ir 1889 loģiskā programmkoda rindas mēnesī; Quartile 1 Range 50% augšējā robeža ir 466 loģiskā programmkoda rindas mēnesī.<sup>[19]</sup>

Attiecīgi, 3 personmēnešu vērts darbs ir attiecīgi 5667 loģiskās rindas mediānas projektiem un 1398 loģiskās rindas pirmās kvartiles projektiem.

Šīs metrikas attiecas uz pilnīgi oriģinālu programmkodu, ne kodu, kas izstrādāts eksistējošā sistēmā.

Kā tika minēts darba ievadā, lielākā daļa programmētāja darba ir koda uzturēšana, nevis jauna koda rakstīšana.<sup>[1][2]</sup> Pētījumu apkopojums, veikts 2010.gadā, secina, ka programmētājs vidēji pavada no 3 līdz 20 reizēm vairāk laika, uzturot kodu (pētījumā minētie skaitļi ir, ka vidēji pavada 75% līdz pat 90% sava laika un resursus uzturēšanā).<sup>[3]</sup>

Citur minētas sarežģītākas formulas<sup>[4]</sup> : tā kā programmētāja darba laiku var attēlot formulā  $T = T_{koda\ rakstīšana} + T_{domāšana} + T_{koda\ lasīšana}$ , kur T ir laiks pēc patvaļīgas vienības, bet visvieglāk aprēķināt minūtēs.

Šajā pētījumā tiek secināts, ka vidējais programmētājs pavada vidēji 5 minūtes dienā, tīri rakstot kodu – pēc pētījuma autora secinājumiem, šis atbilst 10-37 rindām persondienā.

Domāšanas laiks ir grūtāk pieņemama vērtība; pētījuma autors pieņem, ka šī metrika ir 5-10 reizes vairāk, nekā koda rakstīšanas laiks. Darba autors no pieredzes uzskata, ka šai metrikai būtu jābūt 10-20 reizes vairāk. Iegūst, ka vidēji programmētājs pavada aptuveni 50-100 (vidēji 75) minūtes dienā, veidojot savu programmkodu.

Vienā persondienā ir 8 stundas jeb 480 minūtes, tātad iegūst, ka vidēji programmētājs pavada 380-430 (vidēji 410) minūtes dienā, lasot kodu. No šiem aprēķiniem tiek iegūta aptuvena metrika 75:410, jeb aptuveni 1:5.4 – tas ir, programmētājs pavada 5,4 reizes vairāk laika lasot kodu, nekā rakstot. Protams, te nav minēts, cik rindas tiek lasītas šajās 410 minūtēs, bet ir skaidri redzams, ka koda lasīšana ir laikietilpīgāka darba daļa, nekā koda rakstīšana.

Pētījuma autors piemin, ka viņa aprēķinātā metrika ir robežās no 1:7 līdz pat 1:200 – darba autors uzskata, ka šī metrika ir pārspīlējums, un neiekļauj domāšanas laiku.

Ņemot šo faktu, ka koda lasīšana un analizēšana ir grūtāks darbs, nekā koda rakstīšana, ar piebildi, ka oriģinālajai pētītajai sistēmai nebija ne dokumentācijas, ne komentāru – tātad, darba autoram tas bija jādara pašam –, tiks tālāk pieņemta metrika 1:3 – tas ir, koda analizēšana ir trīs reizes laikietilpīgāks un tādējādi grūtāks uzdevums, nekā koda rakstīšana.

Tālāk tiks apskatīti trīs varianti, kā aplūkot darbā paveikto rakstīšanu un analīzi, un kā apskatīto/mainīto rindu skaits ir salīdzināms ar QSM vēsturiskajiem datiem.

1. Optimistiskais variants: tiek ņemtas vērā visas sistēmas kodā komentētās rindiņas – pat tās, kuras atrodas funkcijās, kas netika mainītas. Ir argumenti par un argumenti pret šī darba uzskatīšanu par darba daļu. Visas funkcionālas koda rindas sistēmā tika analizētas, lai izveidotu dokumentāciju, bet tas nenozīmē, ka to analizētais kods bija nepieciešams izmaiņu veikšanai.
  - a. Šajā gadījumā visā sistēmā analizētais koda rindu skaits ir ~3300, ja pieņem, ka kodā visas funkcionālās rindas ir komentētas, un katrs komentārs atbilst vienai funkcionālai rindai. Šo skaitli, reizinot ar 3, iegūst līdzvērtīgo rindu skaitu ~9900, kas krietni pārsniedz QSM datu 3 personmēnešu vērtā projekta mediānu. Pat, ja reizinājums netiek ņemts vērā, pieņemot, ka koda lasīšana ir tikpat sarežģīta, kā rakstīšana, 3300 joprojām pārsniedz *Quartile 1 Range 50%* augšējo robežu.

2. Pesimistiskais variants: tiek ņemtas vērā tikai darba autora veiktās izmaiņas.
  - a. Šajā gadījumā koda rindu skaits ir 515. Ja tiek pieņemts, ka šo rindu rakstīšana ir trīsreiz grūtāka, nekā oriģināla koda rakstīšana, jo bija jāanalizē nedokumentētas sistēmas darbība un kods, iegūst līdzvērtīgo rindu skaitu 1545, kas pārsniedz *Quartile 1 Range 50%* augšējo robežu.
3. Reālistiskais variants: tiek ņemtas vērā darba autora analizētās rindas funkcijās, kas tika izmainītas, kā arī darba autora veiktais izmaiņu kods. Šī metode ir pirmo divu metožu apvienojums.
  - a. Šajā gadījumā analizēto rindu skaits tikai izmainītajās funkcijās ir 1193 rindas, un izmaiņu koda rindu skaits ir 515. Šeit līdzvērtīgo rindu skaitu aprēķinās pēc formulas  $\text{analizētais} * 3 + \text{rakstītais}$  – visprecīzāk ņemot vērā no pētījumiem iegūtās metrikas, ka koda lasīšana ir trīsreiz grūtāks uzdevums, nekā rakstīšana. Iegūst līdzvērtīgo rindu skaitu  $1193 * 3 + 515 = 4094$  rindas, kas krietni pārsniedz *Quartile 1 Range 50%* augšējo robežu un tuvojas QSM datu mediānai.

Visos apskatītajos variantos pēc koda rindām darbietilpība pārsniedz 3 personmēnešus.

## 8. SECINĀJUMI

Kvalifikācijas darba ietvaros tika izpētīta un dokumentēta komandas “*Novel*” Ren’Py dzinēja spēle “*Call of the Void*”, kā arī veiktas un dokumentētas sistēmas izmaiņas.

Darbā aprakstītā ideja – nedokumentētas sistēmas dokumentēšana, analīze un labošana – bija līdz šim neapskatīta tēma, tāpēc nācās radošāk izpausties ar darba veikšanu un tā struktūru, piemēram, pievienot nodaļu tieši par programmatūras kodā veikto izmaiņu uzskaiti.

Darba laikā ne tikai tika dziļi iepazīts Ren’Py dzinējs, bet arī paplašinātas zināšanas Python programmēšanas valodā – kurā darba autors pirms darba tapšanas nebija pārliecināts. Pēc autora domām, šīs zināšanas būs ļoti vērtīgas nākotnē, jo Python ir ļoti izplatīts IT jomā.

Darba ietvaros tika arī gūta pieredze ar spēles sistēmas izstrādi, ko autors varētu tālāk potenciāli izmantot darbā vai neatkarīgā izstrādē, vai arī tikai kā hobiju.

Tālāks darbs ar šo sistēmu un nākamajām spēles sērijām nav plānots, bet darba autors to uzskata par nākotnē apskatāmu tēmu.

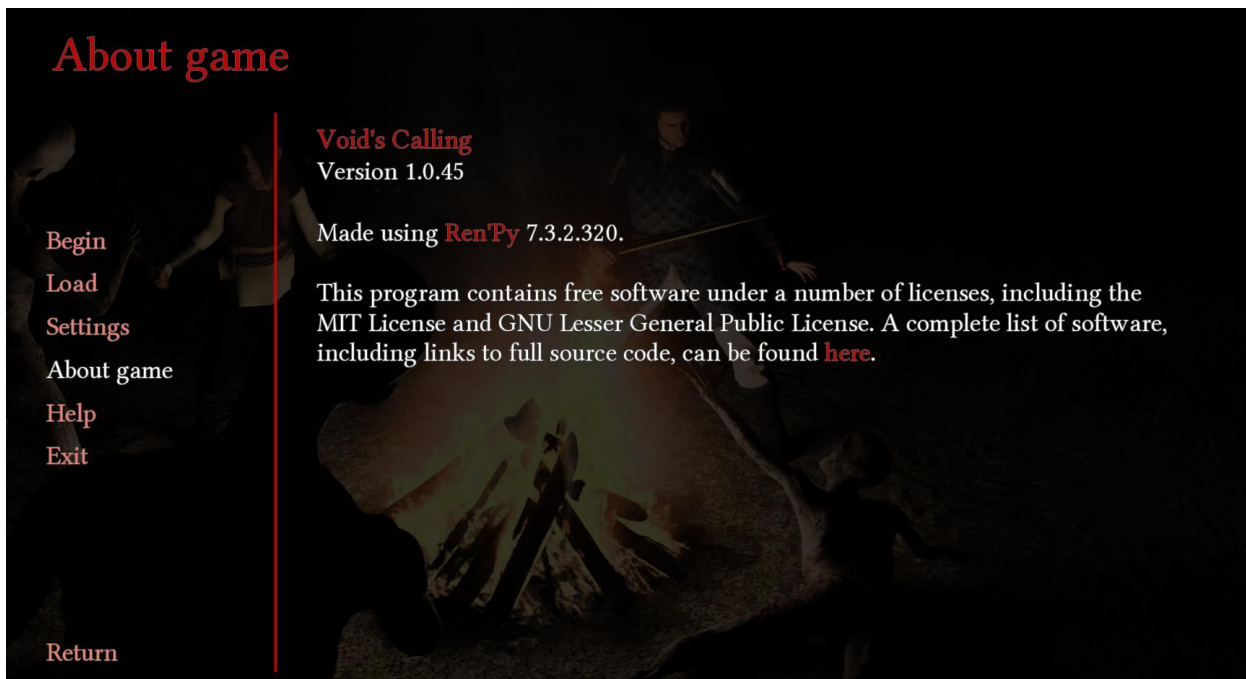
## 9. IZMANTOTĀ LITERATŪRA UN AVOTI

- [1] [https://odo.lv/magic\\_cauldron\\_ar01s03](https://odo.lv/magic_cauldron_ar01s03) Ražošanas mīti – Valdis Vītoliņš
- [2] [https://dspace.lu.lv/dspace/bitstream/handle/7/4708/22822-Edgars\\_Diebelis\\_2012.pdf?sequence=1&isAllowed=y](https://dspace.lu.lv/dspace/bitstream/handle/7/4708/22822-Edgars_Diebelis_2012.pdf?sequence=1&isAllowed=y) Programmatūras paštestēšana – Edgars Diebelis, promocijas darbs, 2012.gads; 37.lappuse
- [3] <https://web.archive.org/web/20120313070806/http://users.jyu.fi/~koskinen/smcosts.htm> Software Maintenance Costs – Jussi Koskinen [arhivēts]
- [4] <https://bayrhammer-klaus.medium.com/you-spend-much-more-time-reading-code-than-writing-code-bc953376fe19> You spend much more time reading code than writing code. – Bayrhammer Klaus
- [5] <https://www.patreon.com/novel> “Call of the Void” izstrādātāju lapa – Patreon
- [6] “LVS 68:1996 Programmatūras prasību specifikācijas ceļvedis” – iegūts no universitātes resursiem 18.10.2022.
- [7] “LVS 72:1996 Ieteicamā prakse programmatūras prasību aprakstīšanai” – iegūts no universitātes resursiem 18.10.2022.
- [8] <https://www.fromdev.com/2023/02/what-are-mods-in-video-games-is-it-legal.html> What Are Mods in Video Games? Is it Legal? – Fromdev Publisher
- [9] <https://www.vg247.com/notch-minecraft-mod-used-to-threaten-my-vision-minecraft-creator-speaks> Notch: 'Minecraft mod used to threaten my vision' - Minecraft creator speaks – Dave Cook
- [10] <https://www.theguardian.com/technology/2017/feb/08/meet-the-blockheads-a-rare-glimpse-inside-minecrafts-hq> Meet the blockheads: a rare glimpse inside Minecraft's HQ – Keith Stuart
- [11] <https://www.digitallawjournal.org/jour/article/view/119> Video game modding in the U.S. intellectual property law: Controversial issues and gaps – Ch. A. Lee
- [12] <https://snyk.io/learn/what-is-mit-license/> What is the MIT License? – Snyk

- [13] <https://snyk.io/learn/what-is-gpl-license-gplv3-explained/> GNU General Public License: GPLv3 explained – Snyk
- [14] [https://www.renpy.org/wiki/renpy/Why\\_Ren%27Py%3F](https://www.renpy.org/wiki/renpy/Why_Ren%27Py%3F) Why Ren'Py? – Ren'Py dokumentācija
- [15] <https://github.com/LuqueDaniel/vscode-language-renpy> Ren'Py extension for Visual Studio Code – LuqueDaniel
- [16] [https://store.steampowered.com/app/1212020/Voids\\_Calling\\_ep1/](https://store.steampowered.com/app/1212020/Voids_Calling_ep1/) Call of the Void spēles lapa platformā Steam – Novel
- [17] <https://github.com/F95Sam/UnRen> UnRen - RPA Extractor, RPYC Decompiler, Console/Developer Menu enabler, and more – F95Sam
- [18] <https://invent.ilmkidunya.com/images/Section/12.pdf> Software Engineering: A Practitioner's Approach, Eighth Edition – Roger S. Pressman, Bruce R. Maxim; 736.lappuse
- [19] <https://failiem.lv/u/jh54amv7n#/view/c64z5db9t> Kā pamatot gatava produkta darbietilpību – Juris Borzovs

## 10. PIELIKUMI

Pielikums 1 – ekrānuzņēmums no sistēmas “About game” lapas, kas parāda programmatūras licences



Pielikums 2 – fragments no mainītās COM\_003 funkcijas

```
label fight_new_round: #COM_003

    #<...>
    if player_action != "parry" and player_action != "saving_attack": #ja darbība
nav atvairisana - ir uzbrukums ##
        if dop_var10 == 0: #ja nav automātiska cīņa
            call screen player_target #izsauc uzbrukumu
            $ player_target = _return #iestata mērķi
        if player_target == "enemy1": #atkarībā no pretinieka
            call cta from _call_cta_1
            if preferences.language == "english":
                $ clog[99] = pcatt.name + " attacks " + enemy1.namer #cīņas vēsturē
ieraksta uzbrukumu
            else:
                $ clog[99] = pcatt.name +_(" атакует ") + enemy1.namer
        if player_target == "enemy2": #tas pats ar pārējiem pretiniekiem
            call cta from _call_cta_2
            if preferences.language == "english":
                $ clog[99] = pcatt.name + " attacks " + enemy2.namer
            else:
                $ clog[99] = pcatt.name +_(" атакует ") + enemy2.namer
        if player_target == "enemy3":
            call cta from _call_cta_3
```

```

        if preferences.language == "english":
            $ clog[99] = pcatt.name + " attacks " + enemy3.namer
        else:
            $ clog[99] = pcatt.name +_(" атакует ") + enemy3.namer
elif player_action != "saving_attack": #ja ir aizsargāšanās ##

    call cta from _call_cta_4
    if preferences.language == "english":
        $ clog[99] = pcatt.name + " gets into a defensive stance" #tas tiek
        atzīmēts cīņas vēsturē
    else:
        $ clog[99] = pcatt.name +_(" встает в защитную стойку")

    if player_action == "saving_attack": ##jauns uzbrukums - veic uzbrukumu
    visiem pretiniekiem un dod veselības punktus
        call cta ##
        if preferences.language == "english": ##
            $ clog[99] = pcatt.name + " attacks all enemies!"## cīņas vēsturē
        atzīmēts
        else:##
            $ clog[99] = pcatt.name + " attacks all enemies!"##
        $ pc.rp = 0 ## atiestata niknuma punktus
        $ target = 2 ## kurš pretinieks ir mērķis
        $ at_target = enemy2 ##iestatīts pretinieks
        call saving_attack from _call_saving_attack ##izsauc uzbrukumu

    if enemy1po > 0 and enemy1 != "none": ##katram pretiniekam, kurš eksistē un
    ir dzīvs, ja tam ir indes efekts:
        $ hp_taken = renpy.random.randint(3, 8) * enemy1po ##atņemtā dzīvība ir
        aprēķināta no rand(3,8) * cik daudz efekta punkti pretiniekam ir
        $ enemy1.hp -= hp_taken ##atņem dzīvības punktus
        $ enemy1po -=1 ##noņem vienu efekta punktu
        call cta ##izsauc vēsturi
        $ clog[99] = enemy1.namer + " is poisoned! -" + str(hp_taken) + " HP"
    ##pieraksta vēsturē
    if enemy2po > 0 and enemy2 != "none": ##tas pats pārējiem
        $ hp_taken = renpy.random.randint(3, 8) * enemy2po ##
        $ enemy2.hp -= hp_taken ##
        $ enemy2po -=1 ##
        call cta ##
        $ clog[99] = enemy2.namer + " is poisoned! -" + str(hp_taken) + " HP" ##
    if enemy3po > 0 and enemy3 != "none": ##
        $ hp_taken = renpy.random.randint(3, 8) * enemy3po ##
        $ enemy3.hp -= hp_taken ##
        $ enemy3po -=1 ##
        call cta ##
        $ clog[99] = enemy3.namer + " is poisoned! -" + str(hp_taken) + " HP" ##
    call enemy1_alive_check ##pārbauda, vai pretinieki dzīvi
    call enemy2_alive_check ##

```

```

call enemy3_alive_check ##

jump fight_enemy_round #pretinieku kārta

```

*Pielikums 3 – COM\_005 izmainītās formulas*

```

$ exp = int (round (exp * (1 + 1/pc.lev))) #iegūtā pieredze, +-20 pēc nejaušības
## labota formula pieredzes iegūšanai
    if exp <0: #ja rezultāts <0, iedod 1 pieredzes punktu
        $ exp = 1
    if loot != 0: #ja laupījums eksistē
        $ loot = loot + renpy.random.randint (-5,5) #laupījums +-
5, var pārdot pie alķīmiķa
        "Победа! Заработано [exp] опыта."
    $ pc.exp += exp #pievieno pieredzes punktus
    if bab >0: #ja iegūta nauda
        $ bab = bab*2 ##vairāk naudas
        "В кошельках убитых было [bab] таленов."
    $ pc.bab +=bab #pievieno naudu
    if loot >0: #ja iegūts laupījums
        "Собрана добыча примерно на [loot] таленов! Стоит
попробовать сдать ее аптекарю."
        $ pc.loot +=loot #pievieno laupījumu, var pārdot pie
alkīmiķa

```

*Pielikums 4 – fragments no COM\_006*

```

label sword_attack: #COM_006 - spēlētāja parasts uzbrukums
    $ attack_random = renpy.random.randint (-3,3) + renpy.random.randint (-3,3) +
pop #vai uzbruks - nejaušs mainīgais ##EDIT: + precizitātes bonuss no COM_001

    $ hit_chance = pc.tohit - at_target.ac + attack_random + mod_at + pop
#iespēja uzbrukt aprēķināta no spēlētāja uzbrukuma precizitātes - pretinieka
aizsardzības prec. + nejaušā + bonuss, ja ir
    if attack_random <=-5: # ~6% iespēja netrāpīt, ja nav pop bonusa, tālāki
aprēķini dokumentā ##EDIT: + precizitātes bonuss no COM_001 -
combat_engine
    #<...>
    elif attack_random >=5 or hit_chance >=0: # ~94% iespēja, ja iespēja uzbrukt
ir pozitīva, ja nav pop bonusa, tālāki aprēķini dokumentā
    # <...>
    if hit_piercing >=0: #ja uzbrukuma spēks lielāks/vienāds par nulli
        if dop_var31 > 0: ##ja spēlētājam ir inde uz zobena:
            if at_target == enemy1: ##atkarībā no pretinieka
                $ enemy1po += 2 ##pie sitiena tiek piešķirti indes efekta
punkti
            if at_target == enemy2: ##tas pats pārējiem
                $ enemy2po += 2 ##
            if at_target == enemy3: ##
                $ enemy3po += 2 ##

```

```

        $ dop_var31 -=1 ## tiek samazināts indes daudzums uz zobena
        $ hp_taken = hit_piercing + pc.str + mod_str + dam + dop_var29 #cik
dzīvības punkti noņemti:
                                                    #cik
spēcīgs uzbrukums + spēks + bonuss, ja ir + bonuss no zobena, ja ir + zobens
asums
        #<...>
return

```

*Pielikums 5 – COM\_008 kods*

```

label saving_attack: ##COM_008 - jauns niknuma uzbrukums - uzbrūk visiem un
atjauno veselības punktus
    $ hp_gain = 0 ##mainīgais, kas seko līdzī, cik jāpievieno dzīvības punkti
    if at_target != "none": #ja pretinieks eksistē
        $ attack_random = renpy.random.randint (-3,3) + renpy.random.randint (-
3,3) + pop ##aprēķina attack_random kā iepriekš ar precizitātes bonusu
        $ hit_chance = pc.tohit - at_target.ac + attack_random + mod_at + pop
#aprēķina hit_chance kā iepriekš

        if attack_random <=-4 or hit_chance < 0: ## ~12% iespēja netrāpīt, ja
hit_chance ir pozitīvs
            if at_target == enemy1: ##atkarībā no pretinieka
                call enemy1_miss ##izsauc atbilstošo skatu
            if at_target == enemy2: ##
                call enemy2_miss##
            if at_target == enemy3:##
                call enemy3_miss##
            call cta##
            if preferences.language == "english":##
                $ clog[99] = " miss!" ##pieraksta cīņas vēsturē
            else:##
                $ clog[99] = _(" промах!")##
        else:##
            $ hit_piercing = 2*(pc.pha + mod_dam + 1) + renpy.random.randint (-5,
5) #aprēķina uzbrukuma stiprumu no šādas formulas ar + rand(-5, 5), lai būtu kaut
kāds nejaušības mehānisms
            if at_target == enemy1: ##izsauc atbilstošo skatu atkarībā no
pretinieka
                call enemy1_hit ##
            if at_target == enemy2: ##
                call enemy2_hit ##
            if at_target == enemy3: ##
                call enemy3_hit ##
            if hit_piercing >= 0: ## ja uzbrukuma stiprums ir nenegatīvs
                if dop_var31 > 0: ##ja spēlētājam ir inde uz zobena:
                    if at_target == enemy1: ##atkarībā no pretinieka
                        $ enemy1po += 2 ##pie sitiena tiek piešķirti indes efekta
punkti

```

```

        if at_target == enemy2: ##tas pats pārējiem
            $ enemy2po += 2 ##
        if at_target == enemy3: ##
            $ enemy3po += 2 ##
        $ dop_var31 -=1 ## tiek samazināts indes daudzums uz zobena
        $ hp_taken = hit_piercing + pc.str + mod_str + dam + dop_var29
#aprēķina, cik dzīvības punkti tiks atņemti
        $ at_target.hp -= hp_taken ## atņem tos pretiniekam
        $ hit_text(hp_taken) ## teksta lodziņš
        call cta ##
        if preferences.language == "english": ##
            $ clog[99] = _("      successful hit ") + "-" + str(hp_taken) +
" HP" ##pierakstīts cīņas vēsturē
        else:##
            $ clog[99] = _("      попадание ") + "-" + str(hp_taken) + "
HP"##
        $ hp_gain = int (round (hp_taken/2.5)) ## iegūtie dzīvības punkti
ir atņemtie/2.5 - spēles līdzsvarošanas nolūkos
        if pc.hp+hp_gain > pc.hp1: ##ja tiktu pārsniegts maksimālais
iespējamais dzīvības punktu skaits
            $ pc.hp = pc.hp1 ##tiek atiestatīti dzīvības punkti
        else: ##
            $ pc.hp+=hp_gain ##citādi tos pievieno spēlētāja dzīvības
punktiem

        call enemy1_alive_check ##pārbauda, vai pretinieki dzīvi
        call enemy2_alive_check##
        call enemy3_alive_check##
    else:## ja uzbrukuma stiprums negatīvs
        if preferences.language == "english": ##
            $ a_text = "No\ndamage" ##netiek atņemti dzīvības punkti,
atzīmē cīņas vēsturē
        else:##
            $ a_text = _("Нет\нповреждений")##
        $ attack_text(a_text) ##atzīmēts uzbrukums cīņas vēsturē
        call cta##
        if preferences.language == "english":##
            $ clog[99] = "      no damage " ##atzīmē cīņas vēsturē
        else:##
            $ clog[99] = "      нет повреждений"##
    if target == 2: ##ja mērķis ir 2.pretinieks (vidējais - tas, kurš gandrīz
vienmēr tur būs)
        if enemy1 != "none": ## ja 1.pretinieks eksistē un ir dzīvs
            $ target = 1 ## mērķis ir 1.pretinieks
            $ at_target = enemy1 ## iestata pretinieku
            call saving_attack ##izsauc to pašu funkciju
        elif enemy3 != "none": ## tas pats ar 3.pretinieku, ja 1.nav dzīvs
            $ target = 3 ##
            $ at_target = enemy3 ##
            call saving_attack ##

```

```

elif target == 1: ##ja ir uzbrukts 2.pretiniekam (ja eksistē) un pašreizējais
mērķis ir 1.pretinieks
    if enemy3 != "none": ##ja 3.pretinieks eksistē un ir dzīvs
        $ target = 3 ##mērķis ir 3.pretinieks
        $ at_target = enemy3 ##iestata pretinieku
        call saving_attack ##izsauc to pašu funkciju
    return ##atgriežas COM_003

```

*Pielikums 6 – izmaiņas VIS\_004 skatā*

```

screen questlog(): #VIS_004 - uzdevumu saraksts

#<...>

textbutton _("Основные задания"): #galvenie uzdevumi
    action SetVariable("menupage", 1)
textbutton _("Побочные задания"): #neobligātie uzdevumi
    action SetVariable("menupage", 2)
textbutton _("Завершено"): #pabeigtie uzdevumi
    action SetVariable("menupage", 3)
textbutton _("Relationships"): ##jauna lapa - attiecības ar tēliem
    action SetVariable("menupage", 4) ##iestata lapu kā 4.lapu

viewport id "quest_vp":
    draggable True
    mousewheel True
    xpos 450 ypos 50 xmaximum 1150 ymaximum 800
    has vbox:
        spacing 7
    if menupage == 1: #galvenie uzdevumi
        #<...>

    if menupage == 2: #neobligātie uzdevumi
        #<...>
        if quest3 == 0: #Ilgas pagātne - slēpts uzdevums
            text _("Вы еще не получили этот квест.") color "#000"
            text ""
        if quest3 == 1: ##
            text _("Exchanging feelings") color "#000" ##EDIT - implementēts
3.neobligātais uzdevums ar Ilgu
            text _("Ilga wanted to know more about me and my mission. She
really is a bright girl, but I'm not sure how wise it is to share information
with her.") color "#000" ##
            text "" ##
        if quest3 == 2:##
            text _("Exchanging feelings") color "#000" ##EDIT - implementēts
3.neobligātais uzdevums ar Ilgu ##

```

```

        text _("Perhaps I was wrong. She seems to have taken a liking to
me, as have I. But she still seems reluctant to trust me fully.") color "#000"##
        text ""##
    if quest3 == 3:##
        text _("Exchanging feelings") color "#000" ##EDIT - implementēts
3.neobligātais uzdevums ar Ilgu
        text _("Almighty, what is wrong with me? She's magnetic, drawing
me in like a lamia. And yet... I can't stop. I must ask her one last thing.")
color "#000"##
        text ""##
    ##EDIT: dzēsts liekais quest4
    #<...>
    if quest11 == 0: ##EDIT: implementēts 11.uzdevums - pie alķīmiķa
        text _("Вы еще не получили этот квест.") color "#000"
        text ""
    if quest11 == 1: ##
        text _("Fury crystals.") color "#000"##
        text _("The local apothecary wants two crystals from a fury. The
reward is two hundred talens.")##
        text ""##
    #<...>

    if quest14 == 0: ##EDIT - pievienots uzdevums bordelī
        text _("Вы еще не получили этот квест.") color "#000"##
        text ""##
    if quest14 == 1: ##EDIT - pievienots uzdevums bordelī - iesākts
        text _("Walking through the rose garden") color "#000"##
        text _("Fiona's girls are all beautiful, there's no doubt about
that. I should spend some time... getting to know them.") color "#000"##
        if q14_ru: ##EDIT: sarakstā tiek atzīmēts, ar kurām meitenēm
pavadīts laiks.
            text _("I've spent time with Ruvi.")##
        if q14_ve:##
            text _("I've spent time with Velana.")##
        if q14_ar: ##
            text _("I've spent time with Armina.")##
        text ""##

    #<...>
    if quest20 ==0:##20.slēptais uzdevums - bērnības atmiņas
        text _("Вы еще не получили этот квест.") color "#000" ##
        text "" ##
    if quest20 ==1: ##20.slēptais uzdevums - bērnības atmiņas, iesākts
        text _("Childhood memories") color "#000" ##
        text _("Some things I just can't forget.") color "#000"##
        if q20_2==1: ##ja 2.sapnis redzēts
            text _("Almighty, I was so hungry. I couldn't finish the
song.")##
        if q20_3==1: ##ja 2.sapnis redzēts

```

```

        text _("I know we were struggling to survive, but to call on
demons... I can't forgive that now.")##
        if q20_4==1: ##ja 2.sapnis redzēts
            text _("I knew she didn't believe in the Almighty. She
worshipped other forces. Not that it made a difference.")##
            if q20_5==1: ##ja 2.sapnis redzēts
                text _("I don't think I should have listened to her. I just
wanted to hear the rest of her song.")##
                if q20_6==1: ##ja 2.sapnis redzēts
                    text _("Who was that Wiesel? How many crimes had my mother
committed?")##
                    text ""##
                    #<...>

if menupage == 3: #pabeigtie uzdevumi
    text _("Завершенные задания:") xalign .5 color "#000"
    text ""
    # <...>
    if quest3 >= 4: ##EDIT - implementēts 3.neobligātais uzdevums ar Ilgu
- pabeigts (lielāks/vienāds, jo ja quest3==5, tad ir pieejams nopirkt indi no
Ilgas)
        text _("Exchanging feelings") color "#000"##
        text _("I don't know how to feel. I should be happy that I've no
competition for her affection, but her story was a lot sadder than I expected. At
least she doesn't have to worry about me. I know I told her that the family life
isn't for me, but... I'd be lying if I said I didn't want to have something more
with her.") color "#000"##
        text ""##
        #<...>
        if quest11 == 2: ##EDIT: izveidots 11.uzdevums.
            text _("Fury crystals.") color "#000"##
            text _("Выполнено.") color "#000"##
            text ""
            #<...>
            if quest14 == 2: ##EDIT - pievienots uzdevums bordelī - pabeigts
                text _("Walking through the rose garden") color "#000"##
                text _("They are all beautiful and skilled indeed, just like
Fiona said. But I wonder if I could get something more if I spent more time with
Fiona.") color "#000"##
                text ""
                #<...>
                if quest20 ==2: ##20.slēptais uzdevums - bērnības atmiņas, pabeigts
                    text _("Childhood memories") color "#000" ##
                    text _("She tried to be a mother. But she was a terrible person.
I hope to never meet her again.") color "#000"##
                    text "" ##
                    #<...>

# izveidota jauna lapa uzdevumu skatā, kas parāda attiecības -----
-----

```

```

    if menupage == 4: ##jauna lapa - attiecības ar citiem tēliem
        text _("Relationships with the townfolk:") xalign .5 color "#000"
##lapas nosaukums, centrēts (daļēji)
        text "" ##
        text _("Affinity with Ilga: " + str(il_like) + " points") color
"#000" ## katram tēlam parādīts, cik liela patikšana
        text "" ##
        text _("Affinity with Kilena: " + str(ki_like) + " points") color
"#000" ##
        text "" ##
        text _("Affinity with Mrisya: " + str(mr_like) + " points") color
"#000" ##
        text "" ##
        text _("Affinity with Londa: " + str(lo_like) + " points") color
"#000" ##
        text ""##
        text _("Affinity with Fiona: " + str(fi_like) + " points") color
"#000"##
        text ""##
        text _("Affinity with Sara: " + str(he_like) + " points") color
"#000"##
        text "" ##
        text _("Affinity with Elana: " + str(el_like) + " points") color
"#000" ##
        text ""##
        text _("Affinity with Iereya: " + str(ie_like) + " points") color
"#000" ##
        text "" ##
        text _("Affinity with Kagera: " + str(ka_like) + " points") color
"#000" ##
        text "" ##
        text _("Affinity with the book merchant: " + str(ho_like) + "
points") color "#000" ##
        text "" ##
        text _("Progress with the burgomaster: " + str(bur_like) + " points")
color "#000" ##
        text "" ##

#<...>

```

### *Pielikums 7 – SCR\_001 izmaiņas*

```

label splashscreen: #SCR_001 - jautāšana par vecumu
$ start_point = "start" #iestata sākuma punktu
call ach_name from _call_ach_name #definē sasniegumu nosaukumus
$ renpy.movie_cutscene("images/video/intro.webm")
stop movie
scene black

```

```

$ dop_var28 = 1 #atslēgtas augšējo stūru izvēlnes
if not persistent.over18: ##EDIT: ja nav iepriekš inicializēts
persistent.over18 mainīgais
    centered "Даная игра содержит материалы, не предназначенные для лиц, не
достигших 18 лет. Для продолжения вам необходимо подтвердить свой возраст. "
    menu:
        "Вам уже есть 18 лет?"
        "Да": #apstiprinoša atbilde
            $ persistent.over18 = True ##EDIT: deklarē ar vērtību True.
Paskaidrojums - persistent mainīgie nav definēti kodā; tā kā tiek izmantota
Python valoda (Ren'Py ir balstīts uz Python),
                    #saglabājas Python īpatnības -
mainīgie tiek deklarēti pie inicializācijas. persistent.[name] mainīgie tiek
saglabāti globālajā datu krātuvē: persistent failā.
                    #Šajā repozitorijā persistent fails
nav iekļauts, jo tas nav atšifrējams ar standarta teksta/koda rediģētājiem, tikai
tā mainīgie rediģējami kodā vai konsolē.
                    #persistent faili ir Ren'Py īpatnība.
Visās tālākajās spēles atvēršanas reizē šī izvēlne netiks parādīta.
        return #galvenā izvēlne
        "Нет": #noraidoša
            $ renpy.quit() #spēle tiek aizvērta
return

```

*Pielikums 8 – implementētas indes lietošanas funkcijas*

```

init -100 python:
#<...>
def p_po1_use(): ##inde
    global dop_var31 ##vajag global birku, lai mainītu krātuves mainīgo
    dop_var31 +=4 ##palielina, cik daudz indes ir uz zobena
    p_po1.qty -=1 ##samazina dziru skaitu

def p_po2_use(): ##stipra inde
    global dop_var31 ##vajag global birku, lai mainītu krātuves mainīgo
    dop_var31 +=7 ##palielina, cik daudz indes ir uz zobena
    p_po2.qty -=1 ##samazina dziru skaitu

```

*Pielikums 9 – LOCM\_001; ACH\_002 izsaukums*

```

label new_morning: #LOCM_001
$ dop_var28 = 0 #parāda augšējo stūru izvēlnes
if not persistent.killer: ##ja nav iegūts jaunais sasniegums
    if monster_count >=100: ##ja nogalināto monsturu skaits ir lielāks/vienāds
ar 100
        $ GetAchievements ("killer") ##tiek iegūts sasniegums - parādīts
lodziņš

```

*Pielikums 10 – quest11 implementēšana pie alķīmiķa*

```
label apt_trade: #LOCC_001
  menu:
    "Спросить про ингредиенты для оберега." if 3<= quest24 <=7 and not
q24apt: #Dainas uzdevums - amulets
    if not q24apt1:
        $ q24apt1 = True #alķīmiķim prasīts par amuletu
        jump quest24_3_apt
    jump quest24_3_apt2 #ja atkārtoti prasīts, tad tiek pieņemts, ka
Forlans maksās par amuleta sastāvdaļu
    "Спросить о необычных событиях в городе" if (m_quest2==1 and not
m_quest2a): #galvenais uzdevums - izmeklēšana
        jump apt_rec
    "Торговать":
        jump apt_trade_main #tirgošanās
    "Продать лут" if pc.loot >0: #pārdot laupījumu
        jump apt_loot
    "Сдать задание" if quest9 == 1 and q_bol.qty >=5: #pabeigt alķīmiķa
uzdevumu
        jump apt_quest9
    "Ask about job" if quest11 == 0: ##
        jump apt_quest11 ##EDIT: izveidots uzdevums
    "Finish task - fury" if quest11 == 1 and q_fur.qty >=2:##
        jump apt_quest11_fin ##EDIT: pabeigts izveidotais uzdevums
    "Уйти":
        jump apt_leave #SBX_002

label apt_quest11: #TXT_004 - alķīmiķa fūrijas uzdevums, ##EDIT: darba autora
izveidots
    fo "- Valenturn, do you have a job for me?" ##
    apt "- As a matter of fact, I do."## ##
    scene apt_4 with dissolve
    apt "- Bring me two crystals from the head of a fury. I want to add them to a
new potion I'm working on." ##
    fo "- Furies are dangerous on the best of days. Why should I help with your
experiments?"##
    apt "- I'll pay you two hundred talens for them, and an extra eighty talens
for every crystal after that."##
    menu:##
        "Agree": #piekrist
            fo "- Deal."##
            apt "- Thank you. If you go looking for them, they usually come out
during daytime."##
            $ showquest("quest11", 1) #uzdevums atzīmēts sarakstā kā iesākts ##
##
            scene apt_2 with dissolve
        "Refuse": ##
```

```

        fo "- Sorry, I don't have the time for this." ## ##
        scene apt_2 with dissolve
        apt "- That's a shame. Well, feel free to ask again whenever you
do."##
        apt "- Anything else?"##
        jump apt_trade #atpakaļ uz tirgošanas ##

label apt_quest11_fin: #TXT_003 - alķīmiķa fūrijas uzdevums ##EDIT: darba autora
izveidots
        fo "- Valenturn, I've accomplished your task and I'm here to collect my
reward." ## ##
        scene apt_5 with dissolve
        apt "- Well, noble Sir, let's see then." ##
        $ q_fur.qty -=2 #2 sakrātas uzdevuma lietas nepieciešamas pabeigšanai ##
        $ pc.bab += 200 #iegūst naudu par uzd pabeigšanu ##
        apt "- Here's your two hundred talens." ##
        if q_fur.qty > 0: ##
            apt "- I'll give you eighty talens additionally for each crystal you
managed to bring, in addition to the requested two crystals, as promised." ##
            $ pc.bab += q_fur.qty*80 #ja vairāk par 2 lietām, katru nākamo pārdod par
80 naudas vienībām ##
            $ q_fur.qty = 0 #un atiestata lietu skaitu ##
            $ renpy.pause(delay=1.0, hard=True) ##ainas pauze ##
            scene apt_4 with dissolve
            $ showquest("quest11", 2) #uzdevums atzīmēts sarakstā kā pabeigts ##
            apt "- Anything else?" ##
            jump apt_trade #turpina tirgoties ##

```

#### *Pielikums 11 – quest14 implementēšana bordelī*

```

label bro_girl_choice: #TXT_004 - izvēlēties meiteni --saistīts ar LOCK_001
    $ loc_dec = ""
    if quest14 == 0: ##EDIT - ja nav uzsākts darba autora izveidotais uzdevums
        $ showquest ("quest14", 1) ##EDIT - atzīmēts kā uzsākts
        $ q14_ru = q14_ve = q14_ar = False ##EDIT - seko līdzī, ar kurām meitenēm
pavadīts laiks
        scene bord_9 with dissolve
        "Повинуясь поданному хозяйкой сигналу, в комнату вошли девушки. Форлан с
предвкушением разглядывал их, выбирая, на которую из красоток сегодня
раскошелиться. Нельзя не признать, девочки в заведении Фионы – все, как на
подбор. Между тем, стоят они по-разному... хм..."
        if pc.bab <=50: #ja nav naudas, netiek dota izvēle
            "У вас недостаточно денег!"
        menu:
            "Руви(50 таленов)." if pc.bab >=50: #izvēle katrai meitenei tiek dota,
tikai ja pietiek nauda
                $ pc.bab -=50 #samaksa
                jump bro_ruvi
            "Велана(70 таленов)" if pc.bab >=70:

```

```

    $ pc.bab -=70 #samaksa
    jump bro_vel
    "Армина(100 таленов)" if pc.bab >=100 and dop_var20==0:
        jump bro_arm
    "Армина(0 таленов)" if dop_var20 == 1: #var apmeklēt Arminu par brīvu -
    šis ir armaiņā pret noteiktu palīdzību Fionas uzdevumā
        jump bro_arm
    "Я зайду позже.": #atteikties

    jump bro_nogirl

label bro_ruvi: #TXT_003 - Ruvi
    fo "- Мне пригласулась Руви."
    if quest14 == 1: ##EDIT: ja darba autora veidotais uzdevums ir uzsākts
        $ q14_ru = True ##EDIT - atzīmēts, ka pavadīts laiks ar meiteni
    if q14_ru == True and q14_ve == True and q14_ar == True and quest14 == 1:
##EDIT: ja ar visām meitenēm pavadīts laiks
        $ showquest ("quest14", 2) ##EDIT: atzīmē uzdevumu kā pabeigtu
        scene bord_10 with dissolve

    fi "- Хороший выбор, господин Форлан. Ступайте в комнату для свиданий."
    scene black with dissolve
    $ renpy.pause(delay=1.0, hard=True)
    call bro_sex1 from _call_bro_sex1
    $ bro_sex_count +=1 #pieskaita jaunu meiteņu apmeklēšanas reizi
    if bro_sex_count ==5: #ja pietiekami daudz reizes apmeklētas meitenes
        $ GetAchievements ("zavs") #tiek reģistrēts sasniegums
        $ renpy.pause(delay=1.0, hard=True)
    jump time_pass

label bro_vel: #TXT_003 - Velana
    fo "- Мне пригласулась Велана."
    if quest14 == 1: ##EDIT: ja darba autora veidotais uzdevums ir uzsākts
        $ q14_ve = True ##EDIT - atzīmēts, ka pavadīts laiks ar meiteni
    if q14_ru == True and q14_ve == True and q14_ar == True and quest14 == 1:
##EDIT: ja ar visām meitenēm pavadīts laiks
        $ showquest ("quest14", 2) ##EDIT: atzīmē uzdevumu kā pabeigtu
        scene bord_10a with dissolve

    fi "- Хороший выбор, господин Форлан. Ступайте в комнату для свиданий."
    scene black with dissolve
    $ renpy.pause(delay=1.0, hard=True)
    call bro_sex3 from _call_bro_sex3
    $ bro_sex_count +=1 #pieskaita jaunu meiteņu apmeklēšanas reizi
    if bro_sex_count ==5: #ja pietiekami daudz reizes apmeklētas meitenes
        $ GetAchievements ("zavs") #tiek reģistrēts sasniegums
        $ renpy.pause(delay=1.0, hard=True)
    jump time_pass

```

```

label bro_arm: #TXT_004 - Armina
fo "- Мне приглянулась Армина."
if quest14 == 1: ##EDIT: ja darba autora veidotais uzdevums ir uzsākts
    $ q14_ar = True ##EDIT - atzīmēts, ka pavadīts laiks ar meiteni
if q14_ru == True and q14_ve == True and q14_ar == True and quest14 == 1:
##EDIT: ja ar visām meitenēm pavadīts laiks un nav vēl pabeigts uzdevums
    $ showquest ("quest14", 2) ##EDIT: atzīmē uzdevumu kā pabeigtu
scene bord_66 with dissolve
if dop_var20 == 1: #ja dabūja apmeklēt par brīvu
    $ dop_var20 = 0 #tālākās apmeklēšanas būs par maksu
    fi "- Я помню, что обещала один раз за счет заведения."
else:
    fi "- Хороший выбор, господин Форлан. Ступайте в комнату для свиданий. "
    $ pc.bab -=100 #samaksa
$ persistent.gal_13 = True #atmiņa atzīmēta galerijā
call bro_arm2 from _call_bro_arm2
"Определённо, задница Армины стоила каждого талена, что Форлан на неё
потратил."
scene bord_93 with dissolve
"Оставив девушку отдыхать на смятой постели, Форлан поблагодарил Фиону и,
удовлетворённый, покинул «Вернскую розу»."
jump time_pass

```

*Pielikums 12 – fūrijas uzdevuma implementēšana; monster\_count palielināšanās demonstrēšana*

```

label forest_fur: #TXT_005 - fūrijas uzbrukums
scene fur_1e with dissolve
"Тварь появилась так внезапно, что Форлан не успел даже увидеть откуда она
взялась. Чешуйчатое тело, грациозные движения... Фурия гротескно походит на
женщину, но не стоит обманываться её кажущейся хрупкостью – силы этих тонких рук
хватает, чтобы разорвать пополам взрослого мужчину, а пальцы твари заканчиваются
острыми когтями."
"Чудовище издало хищный клёкот, приближаясь к человеку и охотник приготовился
встретить грозного врага острой сталью."
scene fur_1d with dissolve

$ pc2 = False
$ pc.rp = 0
$ enemy1 = "none"
$ enemy3 = "none"
$ enemy2 = fur #pretinieks - fūrija
scene forest_6_day with dissolve
call combat_engine from _call_combat_engine_fur
if not lost and dop_var4 == 0: ##ja nenomira un nebēga
    if quest11 == 1:
        $ q_fur.qty += 1 #palielina uzdevuma lietu skaitu ##
        $ monster_count +=1 ##pieskaita fūriju kā monsturu
jump time_pass

```

Kvalifikācijas darbs “Ren’Py dzinēja spēles koda analīze un labošana” izstrādāts Latvijas Universitātes Datorikas fakultātē.

Ar savu parakstu apliecinu, ka darbs izstrādāts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai un/vai recenzentam uzrādītajai darba versijai.

Autors: Agnese Griķe \_\_\_\_\_ .01.2024.

Rekomendēju darbu aizstāvēšanai

Darba vadītājs/a: Mg. sc. comp. Ilvars Miznics \_\_\_\_\_ .01.2024.

Recenzents: Diāna Zavadska

Darbs iesniegts **08.01.2024.**

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs (elektronisks paraksts)