

Liepājas Valsts Tehnikums

Datorspēle “ScoreStorm”

Kvalifikācijas eksāmena praktiskās daļas dokumentācija

Profesionālā kvalifikācija Programmēšanas tehniķis

Grupas nosaukums 4PT

Projekta izstrādātājs Gustavs Narvils

Eksāmena datums 2024. gada 20. Jūnijs

Liepāja 2024

# Satura

[Ievads 4](#_Toc159432514)

[1. Uzdevuma formulējums 5](#_Toc159432515)

[2. Programmatūras prasību specifikācija 7](#_Toc159432516)

[2.1. Produkta perspektīva 7](#_Toc159432517)

[2.2. Sistēmas funkcionālās prasības 7](#_Toc159432518)

[2.2.1. Vispārīgās prasības 7](#_Toc159432519)

[2.2.2. Iestatījuma loga prasības 9](#_Toc159432520)

[2.2.3. Pirmspēles loga prasības 11](#_Toc159432521)

[2.2.4. Spēles loga prasības 12](#_Toc159432522)

[2.3. Sistēmas nefunkcionālās prasības 19](#_Toc159432523)

[2.3.1. Valoda 19](#_Toc159432524)

[2.3.2. Saskaņotība 19](#_Toc159432525)

[2.3.3. Vizuālais izskats 20](#_Toc159432526)

[2.3.4. Datorspēles veiktspējā 20](#_Toc159432527)

[2.3.5. Spēles skaņas 20](#_Toc159432528)

[2.4. Gala lietotāja raksturiezīmes 20](#_Toc159432529)

[3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums 21](#_Toc159432530)

[3.1. Izvēlēto risinājuma līdzekļu un valodu apraksts 21](#_Toc159432531)

[3.2. Iespējamo risinājuma līdzekļu un valodu apraksts 22](#_Toc159432532)

[4. Sistēmas struktūras modelēšana un projektēšana 23](#_Toc159432533)

[4.1. Sistēmas struktūras modelis 23](#_Toc159432534)

[4.2. Klašu diagramma / ER diagramma 23](#_Toc159432535)

[4.3. Funkcionālais un dinamiskais sistēmas modelis 23](#_Toc159432536)

[4.4. Aktivitāšu diagramma 23](#_Toc159432537)

[4.5. Lietotjumgadījuma diagramma 23](#_Toc159432538)

[4.6. Sistēmas moduļu apraksts un algoritmu shēmas 23](#_Toc159432539)

[5. Lietotāju ceļvedis 24](#_Toc159432540)

[6. Testēšanas dokumentācija 25](#_Toc159432541)

[6.1. Izvēlētais testēšanas metodes, rīku apraksts un pamatojums 25](#_Toc159432542)

[6.2. Testpiemēru kopa 25](#_Toc159432543)

[6.3. Testēšanas žurnāls 25](#_Toc159432544)

[7. Individuālais ieguldījums 26](#_Toc159432545)

[8. Secinājumi 27](#_Toc159432546)

[9. Lietoto terminu un saīsinājumu skaidrojumi 28](#_Toc159432547)

[10. Literatūras un informācijas avotu saraksts 29](#_Toc159432548)

[Pielikums 30](#_Toc159432549)

# Ievads

“ScoreStorm” ir datorspēle, kas ir izstrādāta priekš Windows operētājsistēmas datoriem, kur viens no izveidošanas iemesliem bija ka Latvijā nav daudz spēļu veidošanas uzņēmumu vai grupas, kuras izstrādā augstas kvalitātes spēles. Manuprāt, ja Latvijā būtu vairāki spēļu veidotāji, tad tas palīdzētu attīstīt Latvijas IT jomas sektoru un to uzstatīsit pazīstamāku pasaulē, kā arī ienest vairāk biznesu Latvijai. Jo vismaz pagaidām Latvijā nav vel īsti nekāda spēļu veidošanas uzņēmums vai grupu, kura ir ļoti pazīstama visā pasaulē. Tādēļ, manuprāt, šī spēle ir nepieciešama, jo tā varētu attīstīt Latvijas spēļu veidošanas jomu, jo ir iespējams, ka šī spēle var palikt ļoti populāra un tā rezultātā padarītu Latviju zināmāku un pievilcīgāku citiem IT uzņēmumiem gan arī citiem spēles veidošanas uzņēmumiem.

Šī spēle arī, manuprāt, ir nepieciešama, jo personīgi pats esmu ieraudzījis, ka vismaz pagaidām nav neviena spēlē tirgū, kura ir līdzīgi mana spēles idejai, jo viņas visas parasti ir citā spēles žanrā vai arī viņas nav pilnībā realizētas. Piemēram, ir spēlēs, kur galvenais mērķis ir spēlēt vienu un to pašu līmeni, lai dabūtu cik vien augstu rezultātu spēlētājs var, bet gandrīz visas šīs spēles ir citos žanros kā ritmu žanrā vai arī tajās spēlēs, kuras ir šaušanas kategorijas bieži neļauj tev kustēties, vai arī spēles gaita ir ļoti vienkārša un tādā rezultāta nav jautri tās spēlēt atkārtoti. Tādēļ ar savējo spēli es cenšos uzstatīsit to žanrā, kur nav tik daudz šādas veida spēles.

Vel viens iemesls kāpēc šī spēle, manuprāt, ir nepieciešama, jo es zinu, ka šāda veida spēlēm ir liels pieprasījums. Kā ir iepriekš minēts, eksistē spēles, kur spēlētājam vaig dabūt visvairāk punktus, bet viņi ir parasti citos žanros, bet vienalga šīs spēles ir ļoti populāras. Šīm veidu spēlēm parasti ir ļoti aktīvi spēlētāji, kuri sacenšas ar viens pret otru, lai dabūtu visslābāko rekordu. Tādēļ ja ir uztaisīta spēle, kura ir pievilcīga tāda veida spēlētājiem, spēle paliks populāra tāda veida spēlētājiem un arī vienmēr būs aktīvi spēlētāji, kuri centīsies pārspēt cita spēlētāja labāko rekordu.

Pēdējais iemesls kāpēc šī spēle ir nepieciešam ir tāpēc, ka daudzas šaušanas spēlēs pārsvarā ir domātas priekš pieaugušiem tādēļ tajās spēles ir bieži ļoti daudzas lietas, kas nav piemēroti visiem vecumiem, bet ar savējo spēli es izveidošu spēli, ko visi cilvēki vienalga no kāda vecuma var spēlēt un izbaudīt. Tādēļ šī spēle ir nepieciešama, jo tā ir arī piemērota tādiem cilvēkiem, kuri nespēlē tāda veida žanra spēles, jo viņas satur lietas, kas priekš viņiem nav piemērotas. Tas nenozīmē, ka šī spēle ir domāta pilnībā priekš maziem bērniem, bet tas nozīmē, ka spēle ir domāta cilvēkiem pusaudžu vecumā, kuri ir vieni no aktīvākajiem datorspēles spēlētājiem.

# 1. Uzdevuma formulējums

“ScoreStorm” ir datorspēle, kura vismaz pagaidām ir izstrādāta tikai priekš Windows operētājsistēmas, kur spēles galvenais mērķis ir dabūt cik vien daudz punktus fiziski spēles spēlētājs var. Spēles laikā spēlētājs redz savu cilvēku no trešās personas skata un var kontrolē spēlēs cilvēku izmantojot datora tastatūru, un pelīti, kur spēlētājs var skatīties, un staigāt apkārt, gan arī mērķēt un šaut vienu no viņu ieročiem. Pirms spēles sākuma spēlētājam ir dota iespēja izvēlēties vienu no diviem spēlējamiem cilvēkiem, kur katram ir doti divi ieroči, kur katrs strādā savādāk un ir savi plusi un mīnusi. Spēlētājam ir arī dota iespēja izvēlēties starp diviem spēles laukiem, kur atšķirība ir spēles lauka izskats un izkārtojums.

Spēles laikā spēlētājam ir dotas vairākas iespējas, kā viņš varētu iegūt punktus. Pirmkārt, spēlētājs var dabūt punktus nošaujot uz spēles lauka esošos pretiniekus, kuri tev vienmēr uzbruks un arī, kuri vienmēr nejaušās vietās parādīsies uz lauka, kad kāds cits pretinieks tiks nošauts. Spēlē ir divu veidu pretinieki parastie, kurus ir viegli nošaut un nedara tik daudz bojājumus tev spēles laikā un ir īpašie pretinieki, kur viņiem ir kaut kāda speciāls atribūts, kurš pataisa viņus stiprākus, bet ja spēlētājs var vienu nošaut, tad viņš dabūs lielu punktu bonusu. Viens no īpašiem pretiniekiem ir ļoti stiprs, kur viņu ir grūti nošaut un ja viņš tev trāpa tad uzreiz spēle beidzās, bet viņš ļoti lēni kustās, kamēr otrs ir ļoti ātri kustās, bet viņu ir viegli nošaut. Otrkārt, ja spēlētājs spēles laikā grib dabūt daudz punktus raundā, spēlētājs spēles laikā var dabūt reizinātāja kapsulas, kur uz īsu laiku spēlētāja iegūtais punktu daudzums tajā laikā tiks reizināts. Pēdējais veids kā spēlētājs var iegūt punktus spēles laikā ir atrast reti sastopamus priekšmetus, kur ja spēlētājs to atradīs un savāks, tad viņš dabūs lielu punktu bonusu.

Spēles laikā spēlētājam ir arī dots, laiks līdz spēles beigšanai, kur ja laiks beidzās tad spēle automātiski beidzas. Spēles laikā spēlētājam ir doti divi veidi, kā viņš varētu laiku pagarināt, pirmkārt, atrodot laika kapsulas, kuras atrodas apkārt spēles laukam un, otrkārt, nošaujot pretinieku, kur ja nošauj parasto pretinieku, tad laiks tikai pa mazlietiņ palielinās, bet ja nošauj īpašo tad laiks tiek pa vairāk palielināts.

Spēles laikā spēlētājam ir arī dots “kombo” skaitītājs, kur ja spēlētājs var nošaut divus pretiniekus īsā noteiktā laikā tad tas palielināsies, bet ja spēlētājs nenošauj pretinieku noteiktajā laikā, tad tas pazūd. Šis skaitītājs ir svarīgs, jo spēles beigās spēlētajā iegūtais augstākais “kombo” skaitītāja augstākais skaitlis tiek reizināts ar spēlētāja rezultātu un tas nozīmē spēlētājs var iegūt ļoti daudz punktus.

Spēle var beigties dēļ trīs iemesliem, spēles laiks beidzās, kāds pretinieks tevi nošāva vai arī spēlētājs nošāva pilnībā visus pretiniekus, kas ir piedāvāti uz spēles lauka. Kad spēle ir beigusies uz ekrāna tiek parādīts spēlētāja beigās iegūtais rezultāt un arī pasaka vai tas ir jauns rekords vai nē. Kad spēlētājs izvēlas iet atpakaļ uz spēles sākumekrāna viņa rezultāts tiks saglabāts ja tas bija jauns rekords un tad tiks parādīts pirmspēles ekrānā. Spēlētājam sākumekrānā ir poga, kas viņu aizvedīs uz iestatījuma lapas, kur spēlētājs var mainīt daudzas lietas saistībā ar viņu spēli, kā ar kurām pogām kontrolēt spēli, ekrāna rezolūciju, skaņās skaļumu un peles ātrumu.

# 2. Programmatūras prasību specifikācija

Šajā nodaļa tiks aprakstītas visas lietas saistītas par spēlēs perspektīvu, funkcionālajām prasībām, nefunkcionālas prasības un arī kāda ir spēles spēlētāja raksturiezīmes.

## 2.1. Produkta perspektīva

Datorspēle “ScoreStorm” ir trešās personas šaušanas spēle, kur galvenais mērķis tev ir dabūt cik vien daudz punktus tu vari. Šī spēle ir unikāla ar to, ka šī spēle tu var kustēties apkārt pa spēles lauku ar savu spēles tēlu, bet citās spēles tu parasti paliec uz vietas.

Pagaidām datorspēlei visas galvenās funkcijas un prasības ir jau izstrādāt, bet ir vel dažas lietas, kuras būtu vel vēlams izstrādāt vēlāk pēc projekta pabeigšanas. Pirmais ir izstrādāt valodas opcijas, kur lietotājs varēs pārmainīt spēles lietotāja interfeisa valodu uz citu valodu. Pirmā valoda, kura būtu izstrādāta ir angļu, bet pēc tam varētu būtu vācu, spāņu, franču utt. Otrais būtu izstrādāt vairāk spēles tēlus un arī līmeņus, kur katram ir savi unikāli ieroči un arī katram līmeni ir arī savi unikāli pretinieki. Varbūt vel pēc tam izstrādātu spēlei līmeņa veidotāju, kurš ļautu spēlētājiem izstrādāt pašam savus līmeņus, iestatījumus un varbūt vel pat pielikt klāt pašus savus modeļus un tekstūras. Trešā lietā, ko es spēlei gribētu nākotnē izstrādāt ir kaut kāds daudz spēlētāju režīms, kur divi vai vairāk spēlētāji varētu spēlēt vienlaikus vienā līmenī. Pēdējā lieta, ko es gribētu izstrādāt priekš šīs spēles ir atbalsts ar vairākām perifērijas ierīcēm, jo pagaidām vienīgais veids, kā spēlēt šo spēli ir ar peli un klaviatūru, bet nākotnē es gribēt arī dot iespēju spēlēt spēli ar vairākām ierīcēm. Viena ierīce, ko es gribētu atbalstīt nākotnē ir “Xbox” un “PlayStation” spēles pultis, jo ja spēle atbalstītu tiešu tās ierīces, tad ir iespējams nākotnē šo spēli pat uzlikt uz spēļu konsolēm. Otrā ierīce, ko es gribētu atbalstīt ir skārienjutīgie ekrāni, jo nākotnē es šo spēli gribētu uzlikt uz telefoniem tādēļ būtu jāizstrādā priekš šīs spēles virtuālās pogas, lai spēlētājs varētu spēlēt spēli izmantojot pirkstus.

## 2.2. Sistēmas funkcionālās prasības

Šajā nodaļa tiks aprakstītas visas datorspēles “ScoreStorm” funkcionālās prasības.

### 2.2.1. Vispārīgās prasības

**P.1. Datorspēles “ScoreStorm” aizvēršana**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam aizvērt spēli izmantojot pogu.

Ievaddati:

Peles klikšķis uz pogu “Iziet uz darbvirsmu”.

Apstrāde:

1. Tiek pārbaudīts vai spēlētājs atrodas galvenā izvēlne.
2. Tiek pārbaudīts vai spēlētājs nospieda pogu.

Izvaddati:

Tiek spēlēta pogas skaņa un pēc vienas sekundes spēle tiks aizvērta.

**P.2. Datorspēles “ScoreStorm” iestatījuma loga atvēršana**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam atvērt iestatījuma logu izmantojot pogu.

Ievaddati:

Peles klikšķis uz pogu “Iestatījumi”.

Apstrāde:

1. Tiek pārbaudīts vai spēlētājs atrodas galvenā izvēlne.
2. Tiek pārbaudīts vai spēlētājs nospieda pogu.

Izvaddati:

Tiek spēlēta pogas skaņa un pēc vienas sekundes spēlētājs tiek pārvietots uz iestatījuma logu.

**P.3. Datorspēles “ScoreStorm” pirmspēles loga atvēršana**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam atvērt pirmspēles logu izmantojot pogu.

Ievaddati:

Peles klikšķis uz pogu “Spēlēt”.

Apstrāde:

1. Tiek pārbaudīts vai spēlētājs atrodas galvenā izvēlne.
2. Tiek pārbaudīts vai spēlētājs nospieda pogu.

Izvaddati:

Tiek spēlēta pogas skaņa un pēc vienas sekundes spēlētājs tiek pārvietots uz pirmspēles logu.

**P.4. Datorspēles “ScoreStorm” galvenās izvēlnes atgriešanās**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam atgriezties atpakaļ galvenās izvēlnes logā izmantojot pogu.

Ievaddati:

Peles klikšķis uz pogu ar apzīmējumu “X”.

Apstrāde:

1. Tiek pārbaudīts vai spēlētājs atrodas pirmspēles vai iestatījuma logā.
2. Tiek pārbaudīts vai spēlētājs nospieda pogu.

Izvaddati:

Tiek spēlēta pogas skaņa un pēc vienas sekundes spēlētājs tiek pārvietots uz galvenās izvēlnes logu.

### 2.2.2. Iestatījuma loga prasības

**P.1. Datorspēles “ScoreStorm” skaņas skaļuma maiņa**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam mainīt spēles skaņas skaļumu ar bīdni.

Ievaddati:

1. Peles kreisā taustiņa turēšana uz skaņas bīdni.
2. Peli kustina pa labi vai pa kreisi.

Apstrāde:

1. Funkcija pārbauda vai spēlētājs kustina peli.
2. Funkcija pārbauda vai spēlētājs ir sasniedzis spēles maksimālo skaņas skaļuma limitu.
3. Funkcija pārbauda vai spēlētājs ir sasniedzis spēles minimālo skaņas skaļuma limitu.

Izvaddati:

1. Ja spēlētājs bīda bīdni uz kreiso pusi, spēles skaņas skaļums samazināsies.
2. Ja spēlētājs bīda bīdni uz labo pusi, spēles skaņas skaļums palielinās.

**P.2. Datorspēles “ScoreStorm” mūzikas skaļuma maiņa**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam mainīt spēles mūziku skaļumu ar bīdni.

Ievaddati:

1. Peles kreisā taustiņa turēšana uz mūzikas bīdni.
2. Peli kustina pa labi vai pa kreisi.

Apstrāde:

1. Funkcija pārbauda vai spēlētājs kustina peli.
2. Funkcija pārbauda vai spēlētājs ir sasniedzis spēles maksimālo mūzikas skaļuma limitu.
3. Funkcija pārbauda vai spēlētājs ir sasniedzis spēles minimālo mūzikas skaļuma limitu.

Izvaddati:

1. Ja spēlētājs bīda bīdni uz kreiso pusi, spēles mūzikas skaļums samazināsies.
2. Ja spēlētājs bīda bīdni uz labo pusi, spēles mūzikas skaļums palielinās.

**P.3. Datorspēles “ScoreStorm” rezolūcijas maiņu**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam mainīt spēles ekrāna rezolūciju ar izvēlni.

Ievaddati:

1. Spēlētājs nospiež peles kreiso taustiņu virsū uz rezolūcijas izvēlni.
2. Spēlētājs nospiež peles kreiso taustiņu virsū uz viņa izvēlēto rezolūciju.

Apstrāde:

1. Funkcija pārbauda vai spēlētājs ir nospiedis peles kreiso taustiņu.
2. Funkcija pārbauda spēlētāja ekrāna maksimālo rezolūciju.
3. Funkcija pārbauda vai spēlētājs nav izvēlējies tādu pašu rezolūciju kā tagad.
4. Funkcija pārbauda vai spēlētājs ir nospiedis peles kreiso taustiņu uz izvēlēto rezolūciju.

Izvaddati:

Spēlētājs ekrāna izšķirtspēja mainās uz to, ko viņš bija izvēlējies.

**P.4. Datorspēles “ScoreStorm” peles kursora ātruma maiņa**

Mērķis:

Šīs funkcijas mērķis ir nodrošināt iespēju spēlētājam mainīt peles kursora ātrumu ar bīdni.

Ievaddati:

1. Peles kreisā taustiņa turēšana uz peles kursora ātruma bīdņa.
2. Peles kustināšana pa labi vai pa kreisi.

Apstrāde:

1. Funkcija pārbauda vai spēlētājs kustina peli.
2. Funkcija pārbauda vai spēlētājs ir sasniedzis peles kursora maksimālo ātruma limitu.
3. Funkcija pārbauda vai spēlētājs ir sasniedzis peles kursora minimālo ātrumu limitu.

Izvaddati:

1. Ja spēlētājs bīda bīdni uz kreiso pusi, peles kursora ātrums samazinās.
2. Ja spēlētājs bīda bīdni uz labo pusi, peles kursora ātrums palielinās.

### 2.2.3. Pirmspēles loga prasības

**P.1. Spēles tēla izvēle**

Mērķis:

Mērķis ar šo funkciju ir dot iespēju spēlētājam izvēlēties spēles tēlu ar ko grib spēlēt.

Ievaddati:

Spēlētājs noliek peli virs tēla, ko viņš grib spēlēt un nospiež peles kreiso taustiņu.

Apstrāde:

1. Spēle pārbauda vai spēlētājs ir nospiedis peles kreiso taustiņu.
2. Spēle pārbauda vai spēlētājs ir ieguvis rekordus ar izvēlēto tēlu uz katra līmeņa.

Izvaddati:

Tiek atskaņota izvēlēšanās skaņa un tiek parādīts spēles līmeņa izvēlne kā arī zem katra līmeņa tiek parādīts, katra līmeņa rekords ar izvēlēto spēlētāju.

**P.2. Spēles līmeņa izvēle**

Mērķis:

Mērķis ar šo funkciju ir dot iespēju spēlētājam izvēlēties spēles līmeni, ko viņš grib spēlēt.

Ievaddati:

Spēlētājs noliek peli virs līmeņa, ko viņš grib spēlēt un nospiež peles kreiso taustiņu.

Apstrāde:

Spēle pārbauda vai spēlētājs ir nospiedis peles kreiso taustiņu.

Izvaddati:

Tiek atskaņota izvēlēšanās skaņa un spēlētājs tiek aiznests uz spēles loga.

**P.3. Informācijas loga atvēršana.**

Mērķis:

Mērķis ar šo funkciju ir dot iespēju lietotājam apskatīties informācijas logu, kur tiek aprakstīts kā var spēlēt spēli

Ievaddati:

Spēlētājs nospiež peles kreiso taustiņu virs pogas ar apzīmējumu “i”.

Apstrāde:

Spēle pārbauda vai spēlētājs ir nospiedis pogu.

Izvaddati:

Tiek parādīts informācijas logs.

**P.4. Informācijas loga aizvēršana.**

Mērķis:

Mērķis ar šo funkciju ir dot iespēju lietotājam aizvērt ciet informācijas logu.

Ievaddati:

Spēlētājs informācijas logā nospiež peles kreiso taustiņu virs pogas ar apzīmējumu “X”.

Apstrāde:

Spēle pārbauda vai spēlētājs ir nospiedis pogu.

Izvaddati:

Tiek aizvērts informācijas logs.

### 2.2.4. Spēles loga prasības

**P.1. Datorspēles “ScoreStorm” spēles uzsākšana**

Mērķis:

Mērķis ar šo funkciju ir dot spēlētājam iespēju sākt spēlēt.

Ievaddati:

Spēlētājs uzspiež ar peles kreiso klikšķis uz līmeņa, ko viņš gribētu spēlēt.

Apstrāde:

1. Spēle pārbauda vai spēlētājs ir izvēlējies vienu no spēles tēliem.
2. Spēle pārbauda vai spēlētājs ir izvēlējies vienu no spēlēs līmeņiem.

Izvaddati:

1. Spēlētājs tiek pārvietots uz spēles logu.
2. Sākas 3 sekunžu laika atskaita, pēc kuras ir dota lietotājam kontrolē par spēles tēlu un arī sāk skaitīt laiku.
3. Tiek spēlēta tēla stāvēšanas animācija.

**P.2. Datorspēles “ScoreStorm” spēles tēla kustināšana uz priekšu**

Mērķis:

Mērķis ar šo funkciju ir dot spēlētājam iespēju pakustināt viņa spēles tēlu uz priekšu.

Ievaddati:

Spēlētājs nospiež klaviatūras “W” taustiņu.

Apstrāde:

Spēle pārbauda vai lietotājs ir nospiedis klaviatūras “W” taustiņu.

Izvaddati:

1. Spēlētāja tēls tiek lēnām kustināts uz priekšu.
2. Tiek spēlēta atbilstošā animācija.

**P.3. Datorspēles “ScoreStorm” spēles tēla kustināšana uz atpakaļ pusi**

Mērķis:

Mērķis ar šo funkciju ir dot spēlētājam iespēju pakustināt viņa spēles tēlu uz atpakaļ pusi.

Ievaddati:

Spēlētājs nospiež klaviatūras “S” taustiņu.

Apstrāde:

Spēle pārbauda vai lietotājs ir nospiedis klaviatūras “S” taustiņu.

Izvaddati:

1. Spēlētāja tēls tiek lēnām kustināts uz atpakaļ pusi.
2. Tiek spēlēta atbilstošā animācija.

**P.4. Datorspēles “ScoreStorm” spēles tēla kustināšana pa labi**

Mērķis:

Mērķis ar šo funkciju ir dot spēlētājam iespēju pakustināt viņa spēles tēlu pa labi.

Ievaddati:

Spēlētājs nospiež klaviatūras “D” taustiņu.

Apstrāde:

Spēle pārbauda vai lietotājs ir nospiedis klaviatūras “D” taustiņu.

Izvaddati:

1. Spēlētāja tēls tiek lēnām kustināts uz labo pusi.
2. Tiek spēlēta atbilstošā animācija.

**P.5. Datorspēles “ScoreStorm” spēles tēla kustināšana pa kreisi**

Mērķis:

Mērķis ar šo funkciju ir dot spēlētājam iespēju pakustināt viņa spēles tēlu pa kreisi.

Ievaddati:

Spēlētājs nospiež klaviatūras “A” taustiņu.

Apstrāde:

Spēle pārbauda vai lietotājs ir nespiedis klaviatūras “A” taustiņu.

Izvaddati:

1. Spēlētāja tēls tiek lēnām kustināts uz kreiso pusi.
2. Tiek spēlēta atbilstošā animācija.

**P.6. Datorspēles “ScoreStorm” spēles skatīšanās**

Mērķis:

Mērķis ar šo funkciju ir dot spēlētājam iespēju skatīties apkārt spēles laukā izmantojot peli.

Ievaddati:

Spēlētājs pakustina datorspēli uz jebkuru virzienu.

Apstrāde:

1. Spēle pārbauda vai kamera nav sasniegusi viņa maksimālo vai minimālo X pozīciju.
2. Spēle pārbauda vai kamera nav sasniegusi viņa maksimālo vai minimālo Y pozīciju.
3. Spēle pārbauda vai lietotājs kustina peli.

Izvaddati:

Spēles kamera tiks pabīdīta uz to pusi, kur spēlētājs ir novietojis peli.

**P.7. Datorspēles “ScoreStorm” ieroča mērķēšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju ļaut spēlētājam mērķēt spēles ieroci.

Ievaddati:

Spēlētājs tur peles labo taustiņu.

Apstrāde:

Spēle pārbauda vai lietotājs tur peles labo taustiņu.

Izvaddati:

1. Spēles kamera tiek pietuvināta.
2. Tiek spēlēta mērķēšanas animācija.
3. Pazūd tēmēklis.

**P.8. Datorspēles “ScoreStorm” ieroča mērķēšanas beigšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju ļaut spēlētājam beigt mērķēt ieroci.

Ievaddati:

Spēlētājs atlaiž peles labo taustiņu.

Apstrāde:

Spēle pārbauda vai lietotājs ir atlaidis peles labo taustiņu.

Izvaddati:

1. Spēles kamera tiek attuvinātā.
2. Tiek spēlēta tēla stāvēšanas animācija.
3. Pazūd tēmēklis.

**P.9. Datorspēles “ScoreStorm” ieroča šaušana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju ļaut spēlētājam šaut spēles ieroci.

Ievaddati:

1. Spēlētājs tur peles labo taustiņu.
2. Spēlētājs nospiež peles kreiso taustiņu.

Apstrāde:

1. Spēle pārbauda vai lietotājs tur peles labo taustiņu.
2. Spēle pārbauda vai lietotājs ir nospiedis peles kreiso taustiņu.
3. Spēle pārbauda kāds ierocis ir spēlētājam.
4. Spēle pārbauda vai ierocī ir lodes.

Izvaddati:

1. Ja spēlētājam ir lodes tad tiek spēlēts atbilstošais ieroča šaušanas skaņā, tiek spēlēta šaušanas animāciju un ieroča lode tiek izšauta.
2. Ja spēlētājam nav lodes tad tiek spēlētā tukša ieroča skaņas efekts.

**P.10. Datorspēles “ScoreStorm” ieroča pārlādēšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlētājam pārlādēt ieroci ja ir beigušās lodes.

Ievaddati:

Spēlētājs nospiež taustiņu “R”.

Apstrāde:

1. Spēle pārbauda vai lietotājs ir nospiedis taustiņu “R”.
2. Spēle pārbauda vai lietotājam magazīnā ir par vismaz 1 lodi mazāk cik ir maksimums.
3. Spēlē pārbauda vai lietotājam ir vel lodes.

Izvaddati:

Ja spēlētājam ieroča magazīns ir tukšs un spēlētājam ir lodes ar, ko to pielādēt tad spēlē pārlādēšanas animāciju un spēlētāja magazīns pielādējas ar pieejamām papildu lodēm.

**P.11. Datorspēles “ScoreStorm” skriešana**

Mērķis:

Mērķis ar šo funkciju ir dot iespēju lietotājam skriet.

Ievaddati:

1. Lietotājs ir jāstaigā izmantojot “W”, “A”, “S” vai “D” taustiņus.
2. Lietotājs tur kreiso “Shift” taustiņu.

Apstrāde:

1. Spēle pārbauda vai lietotājs staigā.
2. Spēle pārbauda vai lietotājs tur kreiso “Shift” taustiņu.

Izvaddati:

Spēles spēlētājs sāk skriet un tiek atskaņota skriešanas animācija.

**P.12. Datorspēles “ScoreStorm” šaušana par pretinieku**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlētājam šaut par pretinieku.

Ievaddati:

Spēlētājs nospiež peles kreiso taustiņu virs pretinieka.

Apstrāde:

1. Spēle pārbauda vai vietu kur lietotājs mērķē atrodas virsū uz pretinieku.
2. Spēle pārbauda vai pretinieku dzīvības vērtība pēc trāpīšanas ir lielāka par 0.

Izvaddati:

1. Ja pretiniekam trāpa spēlētājs un viņa dzīvības vērtība ir lielāka par 0, tad tiek atskaņota trāpīšanas skaņas efekts un animācija, kā arī tiek atņemtas attiecībā daudz dzīvības atkarībā no ieroča ar, ko trāpīja.
2. Ja pretiniekam trāpa spēlētājs un viņa dzīvības vērtība ir mazāka par 0, tad tiek atskaņota miršanas skaņas efekts un animācija, kā arī tiek spēlētājam pieskaitītas 5 sekundes pie laika, un tiek pieskaitīts +1 spēlētāja “kombo” rādītājam.

**P.13. Datorspēles “ScoreStorm” pretinieku uzbrukšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju, ka spēles laikā pretinieki uzbruks spēlētājam.

Ievaddati:

Spēlētājs sāk spēli

Apstrāde:

Spēle pārbauda vai lietotājs atrodas tuvi pretiniekiem.

Izvaddati:

Ja spēlētājs atrodas tuvi pretiniekiem, tad viņi sāks lēnam uzbrukt, bet ja neatrod tad viņi stāvēs uz vietas.

**P.14. Datorspēles “ScoreStorm” pretinieku parādīšanās**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju, ka spēles laikā citi pretinieki paradīsies, ja tiks nošauti pietiekami daudzi pretinieki.

Ievaddati:

Spēlētājs nošauj pretinieku.

Apstrāde:

Spēle pārbauda vai uz lauka esošā pretinieku skaits ir mazāks par 10.

Izvaddati:

Uz spēles lauka nejaušā vieta parādīsies pretinieks.

**P.15. Datorspēles “ScoreStorm” dzīvības zaudēšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju, ka spēles laikā spēlētājs varēs zaudēt dzīvības, ja pretinieki trāpīs.

Ievaddati:

Spēlētājam trāpa pretinieks.

Apstrāde:

1. Spēle pārbauda vai lietotājam trāpīja.
2. Spēle pārbauda vai lietotājam dzīvības daudzums pēc trāpīšanas nav mazāks par 0.

Izvaddati:

1. Ja spēlētājam trāpa pretinieks un viņa dzīvības vērtība ir lielāka par 0, tad tiek atskaņota trāpīšanas skaņas efekts un animācija, kā arī tiek atņemtas dzīvības.
2. Ja spēlētājam trāpa pretinieks un viņa dzīvības vērtība ir mazāka par 0, tad tiek atskaņota miršanas animācija un spēlētājs tiek aizvests uz spēles beigu logu.

**P.16. Datorspēles “ScoreStorm” laika papildīšanas kapsulas savākšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlētājam palielināt spēles laiku savācos kapsulas, kas atrodas uz laukuma.

Ievaddati:

Spēlētājs uzskrien virsū uz laika kapsulu.

Apstrāde:

Spēle pārbauda vai lietotājs ir uzskrējis uz laika kapsulu.

Izvaddati:

Spēlētāja spēles laikam tiek pieskaitītas 30 sekundes.

**P.17. Datorspēles “ScoreStorm” lodes savākšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlētājam savākt lodes priekš ieročiem spēles laikā.

Ievaddati:

Spēlētājs uzskrien virsū uz lodēm.

Apstrāde:

Spēle pārbauda vai spēlētājs ir uzskrējis virsū uz lodēm.

Izvaddati:

Spēlētāja lodēm tiek pieskaitīts klāt 50 lodes.

**P.18. Datorspēles “ScoreStorm” “kombo” skaitītājs**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlētājam aktivizēt “kombo” skaitītāju, kurš spēles beigās dos lielu bonusu.

Ievaddati:

Spēlētājs nošauj pretinieku.

Apstrāde:

Spēle pārbauda vai lietotājs ir nošāvis divus pretiniekus vismaz 10 sekunžu laika intervālā.

Izvaddati:

Spēlētājam uz ekrāna paradās “kombo” skaitītājs.

**P.19. Datorspēles “ScoreStorm” “kombo” skaitītāja pazušana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlētājam pazaudēt iegūto “kombo”.

Ievaddati:

Spēlētājs nenošauj pretiniekus.

Apstrāde:

1. Spēle pārbauda vai ir pagājušas 10 sekundes bez pretinieka nomiršanas.
2. Spēle pārbauda vai iegūtā “kombo” vērtība ir lielāka par spēlē iegūto lielāko “kombo” vērtību.

Izvaddati:

Spēlētājam pazūd “kombo” skaitītājs un ja tā vērtība ir lielāka par tagadējo spēlē iegūto “kombo” tad šī vērtība pārraksta otru un tiek saglabāta.

**P.20. Datorspēles “ScoreStorm” spēles beigšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlei beigties.

Ievaddati:

Spēles laiks beidzas vai arī tiek nošauti visi spēlē pieejamie pretinieki, vai spēlētājs ir nomirs.

Apstrāde:

1. Spēle pārbauda vai ir beidzies spēles laiks.
2. Spēle pārbauda vai ir nošauti visi pretinieki.
3. Spēle pārbauda vai ir nomiris spēlētājs.

Izvaddati:

Parādās spēles beigu ekrāns un beidzās spēle.

**P.21. Datorspēles “ScoreStorm” iziet uz sākumekrāna poga**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju spēlētājam pārlādēt ieroci ja ir beigušās lodes.

Ievaddati:

Spēlētājs nospiež uz pogu “Iziet uz sākumekrānu”

Apstrāde:

1. Spēle pārbauda vai lietotājs ir nospiedis pogu.
2. Spēle pārbauda vai lietotājs atrodas beigu ekrānā.

Izvaddati:

Spēlētājs tiek aizvests uz sākumekrānu.

**P.22. Datorspēles “ScoreStorm” punktu saskaitīšana**

Mērķis:

Mērķis ar šo funkciju ir nodrošināt iespēju ka iegūtie punkti spēles laikā tiks saskaitīti.

Ievaddati:

Spēlētājs pabeidz spēli.

Apstrāde:

1. Spēle pārbauda vai lietotājs beidz spēli.
2. Spēle saskaita punktus izmantojot formulu: Punkti = (atlikušais laiks sekundēs + nošauto pretinieku skaits) \* lielākais spēle iegūtais kombo.
3. Spēlē pārbauda vai lietotāja iegūtais punktu daudzums ir jauns rekords uz tā līmeņa un ar to cilvēku.

Izvaddati:

Tiek parādīti spēlē iegūtie punkti un ja bija iegūts jauns rekords, tad spēle to arī parādā, kā arī to saglabā.

## 2.3. Sistēmas nefunkcionālās prasības

Šajā nodaļa tiks aprakstītas visas spēles nefunkcionālās prasības.

### 2.3.1. Valoda

Datorspēlei ir jābūt izstrādātai Latvijas Valsts Republikas oficiālā valodā.

### 2.3.2. Saskaņotība

Datorspēlei ir jābūt viegli saprotamai un izmantot tā, lai spēlētājām nebūtu problēmas vai jautājumi datorspēles spēlēšanas laikā. Datorspēles kontrolēm ir jābūt ērtām, loģiskām un ergonomiskām

### 2.3.3. Vizuālais izskats

Datorspēlei ir jābūt vienkāršam vizuālam izskatam, kur nav nekārtības. Vizuālam dizainā pamatā būs divas krāsas balts ar zilu.

### 2.3.4. Datorspēles veiktspējā

Datorspēlei ir jābūt tik optimizētai ka, tā varētu vismaz strādāt biroja datoriem. Spēle tiek uzskatīta, ka tā labi iet ja uz dotā datora tā iet ar ātrumu vismaz 60 kadri sekundē.

### 2.3.5. Spēles skaņas

Spēles mūzikai pēc žanra ir jābūt ātrai un enerģiskai. Spēles skaņas efektiem ir jābūt reālistiskam un atbilstošām spēles objektam, kuram piemīt tā skaņa.

## 2.4. Gala lietotāja raksturiezīmes

Datorspēles “ScoreStorm” galvenais spēlētāju grupa ir jaunieši no 15 līdz 20 gadu vecumam. Šī vecuma grupa tika izvēlēta, jo parasti šīs vecuma grupas cilvēki visbiežāk spēlē trešās personas šaušanas spēles. Šīs datorspēles galvenais spēlētāju grupa arī iekļauj cilvēkus, kuriem ir jau pieredze spēlējot šāda veida spēles un arī cilvēkiem, kuriem ir pieredze spēlējot šāda veida punktu krāšanas spēles.

Šī spēle ir izstrādāta tā, lai tā varētu turēt spēlētāju interesi uz ilgu laiku un likt viņiem gribēt spēlēt spēli ilgu laiku ar spēles punktu sistēmu. Spēle ir arī izstrādāta saprotama un ērta priekš spēlētājiem tā, lai viņiem nerastos problēmas kaut ko saprast, kamēr spēlējot spēli.

# 3. Izstrādes līdzekļu, rīku apraksts un izvēles pamatojums

Šajā nodaļā tiks aprakstīti izstrādes rīki, kuri tika izmantoti datorspēles izstrādes laikā, kā arī šajā nodaļā ir aprakstīts, kādi vel izstrādes rīki varēja būt izmantoti izstrādes laikā nevis tie kuri tika izmantoti.

## 3.1. Izvēlēto risinājuma līdzekļu un valodu apraksts

Izstrādājot šo datorspēli tika izmantoti daudzi līdzekļi kuri padarīja izstrādes procesu daudz vieglāku.

Datorspēles veidošanā galvenais rīks, ko izstrādātājs izmantoju bija Unity. Šī izstrādes vide galvenokārt tika izmantota, jo tā bija vide, ko projekta izstrādātājs bija izmantojis un arī bija pieredzes to izmantot. Vel viens iemesls kāpēc tika izvēlēts Unity bija pieeja Unity Asset Store, kur ir iespējams lejupielādēt vai iegādāties modeļus, skaņas un UI elementus, kurus pēc tam viegli ir pievienot eksistējošam Unity projektam. Kamēr notika datorspēles izstrāde Unity versiju, kuru izmantoja izstrādātājs bija tā laika jaunākā LTS versija 2022.3.16f1. Tā kā datorspēle bija izstrādāta Unity tas arī nozīmē ka tika izmantota C# programmēšanas valoda, kura ir vispārējas nozīme programmēšanas valoda tas nozīmē, ka šī valoda var būt izmantoti gan datorspēles izstrādei, bet arī var būt izmantoti citu lietu izstrādei. Ka izmanto Unity tā piedāvā izmantot MonoDevelop izstrādes vidi vai Visual Studio izstrādes vidi un projekta izstrādātājs izvēlējās Visual Studio. Visual Studio tika izmantots, jo tai bija daudz jaunāka un modernāka nekā MonoDevelop un arī tai bija labāka saderība ar jaunākām Unity versijām nekā MonoDevelop. Visual Studio arī vieglāk veikt programmas kodu atkļūdošanu nekā MonoDevelop izstrādes vidē, kā arī Visual Studio ir daudz stabilāka un arī piedāvā vairāk personalizēšanu nekā MonoDevelop.

Daži citi Rakstot dokumentāciju un testpiemērus tika izmantots gan Microsoft Word, gan arī Microsoft Excel. Šie rīki galvenokārt tika izvēlēti, jo projekta izstrādātajam bija jau pieredze ar šiem rīkiem. Vel viens iemesls kāpēc šie rīki tika izvēlēti bija jo šie rīki visvecākie tādēļ tie ir arī viss atjaunotākie, kā arī ar vairāk funkcijām nekā citi rīki priekš dokumentu rakstīšanas vai datu apstrādi.

Vel pēdējais rīks kurš tika izmantots datorspēles izstrādē bija GitHub. Šis rīks ļāva izstrādātājam augšupielādēt datorspēles projekta failus uz mākoni un turēt tos failus līdz izstrādātājs gribēja veidot projektu izmantojot citu datoru vai arī izstrādātājs gribēju atgriezties uz vecāka projekta versijas. GitHub tika tieši izvēlēts, jo tas ir veidots tieši priekš projektu izstrādēm. GitHub ir ļoti daudzas labas funkcijas, bet trīs galvenākie ir, viegla projekta versonēšana, viegla projektu augšupielādēšana un arī atbilstība ar vairākām izstrādes vidēm.

## 3.2. Iespējamo risinājuma līdzekļu un valodu apraksts

Izstrādājot datorspēli, projekta izstrādātajam bija arī liela izvēle ar alternatīviem rīkiem, ko izmantot datorspēles izstrādei.

Galvenā izstrādes rīks datorspēlei bija Unity, bet arī eksistē divi alternatīvi, kuri varēja būt izmantoti izstrādes laikā un tie bija Unreal Engine, un Godot. Abas šīs vides būtu bijušas labas izstrādes vides priekš šīs datorspēles, bet bija dažas lietas abiem, kuru dēļ projektu izstrādātājs šīs vides neizvēlējās. Galvenais iemesls kāpēc projektu izstrādātājs neizvēlējās šīs izstrāžu vides bija dēļ tā, ka projektu izstrādātajam nebija nekādas pieredzes ar šīm izstrāžu vidēm tādēļ nezinot vai ir iespējams izstrādāt šīs spēles vīziju tajās vidēs. Viens iemesls kāpēc netika izvēlēts Unreal Engine bija tas, ka salīdzinot ar Unity daudz vairāk datoru resursu intensīvs un tas nozīme, ja datorspēle būtu bijusi izstrādāta tajā vidē tad būtu bijuši vairāki datorspēles lietotāji, kuri nevarētu spēlēt datorspēli uz saviem datoriem, jo viņiem varbūt ir lēns dators. Viens iemels kāpēc netika izvēlēts Godot bija tāpēc ka tas salīdzinot ar Unity ir daudz jaunāks un arī mazāk izmantotāks nekā Unity, un tas nozīme ja projektu izstrādātajam būtu bijusi kaut kāda problēma datorspēles izstrādes laikā vai arī nezinātu, kā kaut ko izdarīt tajā vidē tad būtu grūtāk atrast resursus kā problēmu izdarīt vai novērst nekā ar Unity.

Rakstot dokumentāciju vai testa piemērus bija arī citi rīki ar ko to varētu izdarīt un tie bija Google Docs un Google Sheets, kā arī Libre Office. Ar šiem rīkiem projektu izstrādātājs varēja rakstīt dokumentāciju un testa piemērus, bet galvenais iemesls kāpēc tie netika izmantoti, jo tiem ir ļoti ierobežoti tekstu formatēšanas opcijas un tas nozīme, ka projektu izstrādātajam būtu bijis grūti veidot dokumentāciju pēc dotām prasībām. Rīkam GitHub arī eksistē alternatīvi kā Bitbucket, GitLab un Source Forge, bet šos rīkus neizvēlējas projektu izstrādātājs, jo nebija pieredzes ar šīm vidēm un arī GitHub ir labāks atbalsts priekš Unity salīdzinot ar alternatīviem.

# 4. Sistēmas struktūras modelēšana un projektēšana

## 4.1. Sistēmas struktūras modelis

## 4.2. Klašu diagramma / ER diagramma

## 4.3. Funkcionālais un dinamiskais sistēmas modelis

## 4.4. Aktivitāšu diagramma

## 4.5. Lietotjumgadījuma diagramma

## 4.6. Sistēmas moduļu apraksts un algoritmu shēmas

# 5. Lietotāju ceļvedis

# 6. Testēšanas dokumentācija

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

## 6.2. Testpiemēru kopa

## 6.3. Testēšanas žurnāls

# 7. Individuālais ieguldījums

# 8. Secinājumi

# 9. Lietoto terminu un saīsinājumu skaidrojumi

# 10. Literatūras un informācijas avotu saraksts

# Pielikums