

VIDZEMES AUGSTSKOLA
INŽENIERZINĀTŅU FAKULTĀTE

**IZLAUŠANĀS TELPAS SPĒLES PROTOTIPA
IZSTRĀDE VIRTUĀLAJĀ REALITĀTĒ**

BAKALaura DARBS

Autors: Eduards Jesikens

Stud. apl. Nr. IT16009

Darba vadītājs: Dr.sc.ing., Asoc.prof. Arnis Cīrulis

VALMIERA 2020

KOPSAVILKUMS

Autors: Eduards Jesikens, Stud.apl.nr: IT16009

Darba vadītājs: Dr.sc.ing., Asoc.prof. Arnis Cīrulis

Izlaušanās telpas spēles prototipa izstrāde virtuālajā realitātē. – Bakalaura darbs, Valmiera: Vidzemes augstskola, 2020. – 56 lapaspuses, 4 tabulas, 30 attēli, 3 pielikumi.

Virtuālā realitāte pēdējo dažu gadu laikā strauji ir attīstījusies un turpina iekarot citas industrijas ar VR pielietojumu. VR kļuva populāra pateicoties spēlēm virtuālajā vidē - cilvēki, sēžot mājās, varēja piedzīvot aizraujošas emocijas, kuras arī varēja sasmīdināt citus. Viena no populārākajām spēlēm bija amerikāņu kalniņi, kur spēlētājs sēdēja uz vietas un piedzīvoja vēderu kutinošās sajūtas. No malas vērotājiem arī bija uzjautrinoši to skatīties, kā cilvēki reaģē. Bet virtuālās realitātes pielietojums nav saistīts tikai ar spēlēm un izklaidi, to arī pielieto mārketingam, medicīnā, dažāda veida terapijās, kā arī militārajā dienestā un pat kosmosā. Kā redzams, pielietojums ir ļoti plašs un svarīgs.

Darba mērķis ir izveidot interaktīvu izlaušanās spēles prototipu ar uzdevumiem un VR pamatiem – personāža kustība un mijiedarbība ar objektiem. Lai šo mērķi sasniegtu, autors izvirza tādas darba uzdevumus, kā izlaušanās telpu uzdevumu analīze, līdzīgu projektu izpēte, projekta izstrāde un uzdevumu realizēšana autora spēlē.

Darba laikā tika izstrādāts prototips virtuālās realitātes izlaušanās spēlei, kuru autors vēlas attīstīt pēc bakalaura aizstāvēšanas un, cerams, izlaist kā gatavu pārdodamu un kvalitatīvu produktu. Spēlē ir izstrādāti trīs autora izdomāti uzdevumi, kurus izpildot, spēlētājs iziet pirmo, mazo daļu no spēles.

SUMMARY

Author: Eduards Jesikens, Student's ID: IT16009

Supervisor: Dr.sc.ing., Asoc.prof. Arnis Cīrulis

Escape room game prototype in virtual reality. – Bachelor's Thesis, Valmiera, Vidzeme University of Applied Sciences, 2020. – 56 pages, 4 tables, 30 pictures, 3 annexes.

Virtual reality in the last couple of years has rapidly developed and continues to conquer other industries with VR usage in them. VR first became popular because of VR games – people, sitting at home, were able to experience amazing emotions, which may bring laughter to others. One of the most popular VR games at first were rollercoaster games, where the player was sitting down and experiencing the mind-blowing emotions. Others sitting nearby were always laughing at how player reacted to rollercoaster experience. But virtual reality is not only used in gaming and having fun, but also in marketing, medical field, therapy, military and even in cosmos. As seen VR can be used in variety of fields and it is important.

The aim of the work is to develop prototype for VR escape room game. Which will include puzzles and VR basic elements – player movement and interactions. To accomplish this goal author set some tasks, that include analyzing escape room puzzles, similar project analysis, project development and puzzle implementation in project.

During the work author developed VR escape room game prototype, which author wants to develop further so that the product is sellable and of good quality. Game includes three puzzles made by author, when those are completed, the player will have finished the first small part of the whole game.

KRIEVU VAL SAVILKUMS

Автор: Eduards Jesikens, свидетельство студента: IT16009

Научный руководитель: Dr.sc.ing., Asoc.prof. Arnis Cīrulis

Прототип игры квест комнаты (escape room) в виртуальной реальности. -

Бакалавриат, Валмиера, Видземский университет прикладных наук, 2020 г. - 56 страниц, 4 таблицы, 30 рисунков, 3 приложения.

Виртуальная реальность (VR) в последние пару лет быстро развивается и продолжает завоевывать другие индустрии. VR впервые стала популярной благодаря VR играм - люди, сидя дома, смогли испытать удивительные эмоции, которые могут вызвать смех у других. Сначала одной из самых популярных VR игр были игры на американских горках, где игрок сидел и испытывал потрясающие эмоции. Другие, сидевшие рядом, всегда смеялись над тем, как игрок реагирует на опыт американских горок. Но виртуальная реальность используется не только в играх и развлечениях, но и в маркетинге, медицине, терапии, армии и даже в космосе. Как видно, VR применение очень широкое и важное.

Целью работы является разработка прототипа для игры в виртуальной комнате, которая будет включать в себя головоломки и основные элементы VR разработки - движения игрока и взаимодействия. Для достижения этой цели автор поставил несколько задач, которые включают анализ головоломок квеста, исследование похожих проектов, разработку проекта и реализацию головоломок в проекте.

В течении работы автор разработал прототип игровой комнаты VR (escape room), который автор хочет развивать, чтобы продукт можно было продать. Игра включает в себя три загадки, сделанные автором, когда они будут завершены, игрок закончит первую небольшую часть всей игры.

ATSLĒGVĀRDI UN SAĪSINĀJUMI

3D – Trīs dimensionāls

6DoF – objekta kustība un rotācija 3D vidē pa 3 asīm. (angl. 6 degrees of freedom)

AR – Papildinātā realitāte

Github – versijas kontroles platforma

HMD – uz galvas velkams ekrāns (angl. *Head mounted display*)

Kinect – kustību izsekošanas ierīce pie Xbox

PC – personālais dators

PS – spēļu konsole PlayStation.

PS4 – spēļu konsole Playstation 4

PS5 – spēļu konsole Playstation 5

PSVR – spēļu konsoles PlayStation Virtuālās Realitātes brilles

Steam – spēļu platforma

SteamVR- programma, kas savieno VR ierīces ar datoru

VR – Virtuālā realitāte

VRTK – VR izstrādes palīgriks

XBOX – spēļu konsole Microsoft Xbox

SATURS

KOPSAVILKUMS	2
SUMMARY	3
KRIEVU VAL SAVILKUMS	4
ATSLĒGVĀRDI UN SAĪSINĀJUMI	5
SATURS	6
IEVADS	8
1. Virtuālā realitāte	10
1.1 Virtuālās realitātes ierīces	10
1.3.1 Virtuālās realitātes ierīču salīdzināšana	14
1.2 Virtuālās realitātes pielietojums mūsdienās	16
1.3 Virtuālā realitāte skaitļos	19
1.4 Virtuālās realitātes spēļu analīze, to tipi un pieejamība mūsdienās	20
2. VR izlaušanās telpu spēļu funkcionalitātes salīdzinājums	25
2.1 Mīklu hierarhija	27
2.2 Izlaušanās spēļu mīklu veidi	29
2.3 VR izlaušanās spēles	29
2.3.1 VR izlaušanās spēļu analīze	30
2.4 Autora veidotās spēles pamatojums un informācija	32
3. Izlaušanās spēles izstrādes pamatposmi	34
3.1 Izmantotie skripti	38
3.2 Autora veidotie skripti	41
3.3 Ekonomiskais ieguvums	48
3.4 Ekonomiskais novērtējums	48

Secinājumi.....	51
IZMANTOTĀ LITERATŪRA.....	53
I PIELIKUMS. Spēles skripti.	56
II Pielikums. Apliecinājums par autora mantisko tiesību nodošanu.....	68
III Pielikums. Apliecinājums par darba atbilstību.	69

IEVADS

Virtuālā realitāte pēdējo dažu gadu laikā strauji ir attīstījusies un turpina iekarot citas industrijas ar VR pielietojumu. VR kļuva populāra pateicoties spēlēm virtuālajā vidē - cilvēki, sēžot mājās, varēja piedzīvot aizraujošas emocijas, kuras arī varēja sasmīdināt citus. Viena no populārākajām spēlēm bija amerikāņu kalniņi, kur spēlētājs sēdēja uz vietas un piedzīvoja vēderu kutinošās sajūtas. No malas vērotājiem arī bija uzjautrinoši to skatīties, kā cilvēki reaģē. Bet virtuālās realitātes pielietojums nav saistīts tikai ar spēlēm un izklaidi, to arī pielieto mārketingam, medicīnā, dažāda veida terapijās, kā arī militārajā dienestā un pat kosmosā. Kā redzams, pielietojums ir ļoti plašs un svarīgs.

Apvienojot šo aizraujošo tehnoloģiju ar izlaušanās istabām, kas arī ir kļuvušas par ļoti populāru grupu izklaidi, pēc autora domām, ir iespēja radīt kaut ko tiešām aizraujošu. Spēlētājam ir iespēja piedzīvot sajūtas, ka ir ieslodzīts un vienīgais, kas palīdzēs izkļūt ir atjautība.

Izlaušanās telpas izstrāde notiek izmantojot spēļu dzinēju "Unity". Šajā platformā ir spraudņi, kas atvieglo VR izstrādi, lai spēļu veidotāji var koncentrēties uz svarīgākiem spēles aspektiem, nevis mēģināt savienot ierīces vai nodrošināt kustību piemēram. Autors spēli veidos kā prototipu, ar ideju to vēlāk attīstīt un piedāvāt vienai Rīgas VR studijai, lai varētu monetizētu šo ideju.

Problēma:

Latvijas VR studijas, uzņēmumi, kuri piedāvā uz vietas izmantot VR brilles un spēlēt VR spēles, nepiedāvā Latvijā veidotas VR izlaušanās spēles. Protams, var izmantot jau izstrādātas ārpus Latvijas spēles, bet tas nenāk par labu Latvijai. Autors uzsver tieši uz Latvijā izstrādātām spēlēm, lai popularizētu spēļu izstrādi Latvijā un iedrošinātu Latvijas spēļu izstrādātājus taisīt produktus, kurus var pārdot gan Latvijas tirgū, gan ārpus Latvijas robežām.

Darba mērķis:

Izveidot interaktīvu izlaušanās spēles prototipu ar uzdevumiem un VR pamatiem – personāža kustība un mijiedarbība ar objektiem.

Darba mērķa sasniegšanai izvirzītie uzdevumi:

1. Izpētīt izlaušanās telpas, to uzdevumus jeb mīklas
2. Izpētīt VR izstrādātas izlaušanās spēles
3. Iepazīties ar VR izstrādes pamatiem
4. Izdomāt autora spēlei stāstu un uzdevumus
5. Izveidot 3D vidi
6. Nodrošināt VR pamata kontroles – kustību, mijiedarbību ar elementiem
7. Realizēt izdomātos uzdevumus autora spēlē

Izmantotās metodes:

1. Teorētiskās literatūras analīze
2. Datu ieguve
3. Datu analīze
4. Unity 2019.3 Personal
5. HTC Vive

Mērķauditorija:

Spēles mērķauditorija ir liela, jo zem tās iekļaujas cilvēki, kam gan patīk virtuālā realitāte, gan tie, kas nekad to nav mēģinājuši un ir interese par šo tehnoloģiju. Tad ir arī tādi cilvēki, kam jau ir VR brilles un grib izmēģināt tieši VR izlaušanās spēles. Vai arī tādi, kas ir bijuši jau izlaušanās telpās un interesē, kā tas tiek realizēts virtuālajā realitātē.

Spēles tiešā mērķauditorija ir potenciālā VR studija, kurai varētu piedāvāt šo spēli, kad tā tiks pabeigta, kā arī spēļu platforma *Steam*.

1. Virtuālā realitāte

Virtuālā realitāte izmanto datora tehnoloģijas, lai simulētu mākslīgu vidi. Atšķirībā no tradicionālajām lietotāja saskarnēm, VR ievieto spēlētāju iekšā spēlē vai VR pieredzē. Tā vietā, lai sēdētu un skatītos priekšā uz ekrānu, VR lietotāji ir pilnībā iegremdēti VR pieredzē un spēj mijiedarboties ar 3D vidi, objektiem un citiem personāžiem. (What is virtual reality?, 2019)

VR visvairāk atpazīstamākā komponente ir uz galvas velkamais ekrāns (angl. head-mounted display), ar kura palīdzību lietotājs var skatīties apkārt spēlē vai pieredzē izmantojot savu galvu – kur pagriežies dzīvē, uz to pusi pagriežas virtuālajā realitātē.

1.1 Virtuālās realitātes ierīces

Virtuālās realitātes ierīču klāsts sākās no parasta viedtālruna līdz brillēm, kas skenē un atpazīst apkārtējo vidi un pirkstu kustības, lai spēlētājam būtu pēc iespējas labāka pieredze. Kā arī ir iespēja iegādāties papildus ierīces, kas padarīs pieredzi vēl īstāku, piemēram, skrejceliņš.

Ja runā par VR tirgu, tad ir četri vadošie uzņēmumi – *Oculus*, *PlayStation*, *HTC* un *Valve*, kas nesen izlaida tirgū “Index” brilles.

1. Brilles, kuras gaidīja ar milzīgu nepacietību, ir *Valve Index*. Tās uzskata par nākamās paaudzes brillēm.



1.1 attēls: Valve Index VR brilles ar sensorium un kontrolieriem (**Best VR headsets 2020, 2020**)

Plusi: Plašākais skatījuma leņķis, ērti kontrolieri ar pirkstu izsekošanas sensoriem

Mīnusi: Nepieciešama augstas klases grafiskā karte, dārgs

Index brillēm ir īpaši laba izšķirtspēja (angl. *ultra-crisp*), kas arī diezgan labi darbojas ar vecākām grafiskajām kartēm, plašāks skatījuma leņķis, lielāks atjaunošanas biežums un *Valves* kontrolieri “Knuckle”, kas spēj izsekot katra pirksta kustībai ļoti precīzi, laba istabas mēroga izsekošana un ļoti laba skaņas kvalitāte.

Valve Index ir noteikti milzīgs uzlabojums pēc *HTC Vive*, kā arī ir daudz plūstošāks nekā *HTC Vive Pro*, spēcīgas VR brilles, kurām nesanāca nodrošināt solīto augstās klases istabas mēroga VR pieredzi.

Un ar visu to, ka *Index* ir vislabākās VR brilles, tas ir līdzīgs jau ierastās VR niķības – tā iestatīšana var būt piņķerīga un ilga, atjauninājumi var radīt savienojuma problēmas, var būt jauna kļūda, kad nākamreiz izmanto, *SteamVR* ir nopietnas problēmas un joprojām izmanto bāzes stacijas sensorus.

Tas ir dārgs, jā, bet īstiem spēļu cienītājiem, kas vēlas nākamo VR pieredzes līmeni, *Valve Index* ir vienīgais variants. (Best VR headsets 2020, 2020)

2. Labākā patstāvīgā ierīce ir *Oculus Quest* – viss vienā virtuālās realitātes sistēmā.



1.2 attēls: Oculus Quest VR brilles (Best VR headsets 2020, 2020)

Plusi: Perfekta VR pieredze no patstāvīgas ierīces, laba cena, nav vadu, 6DoF, var izmantot mazās un lielās telpās

Mīnusi: Mazliet var redzēt gaismu un apkārtējo vidi uzvelkot brilles, baterija iztur divas stundas (Best VR headsets 2020, 2020)

Oculus ir misijā, lai VR varētu izmantot pēc iespējas vairāk cilvēku. Tam ir veikspējas jauda, kas mērojas ar datora sniegto VR, bet ir mazā patstāvīgā ierīcē. Nav dārgs un nav nekas papildus jāpērk.

Quest nodrošina 6DoF izsekošanu. Tas ir pateicoties diviem kustību kontrolieriem, kā arī sensoriem, kas ir iebūvēti brīļļu sānos. Tas nozīmē, kā lietotājs var staigāt droši apkārt, pietupesties, pieliekties un protams, nav jāuztraucas un jādomā, kur atrodas vadi, kas nāk no brīļlēm, un to visu pārvērtīs VR kustībā.

Kā arī liels panākums *Quest* ir tas, ka tas nesniedz nodrošināja roku kustību izsekošanu ar programmatūras atjaunojumu – lietotājs var spēlēt vai izmantot izvēlni bez kontroliera, bet izmantot savus pirkstus un rokas.

Neņemot vērā cenu un dažas mazas problēmas, *Quest* ir jautrs, imersīvs (angl. *immersive*) un piesaistošs VR risinājums, kas uzvar arvien vairāk un vairāk cilvēku. (Oculus Quest review, 2019)

3. Labākais VR risinājums spēļu konsolēm ir PlayStation VR.



1.3 attēls: Playstation VR brilles (Best VR headsets 2020, 2020)

Plusi: Maza cena, tuvu datora veikspējai, pietiekami labs spēļu klāsts

Mīnusi: Nav pieciešamo piederumu kastē, laiž garām gaismu, uzkrītoša/ raustās kustību izsekošana

Nevar noliegt, lai izmantotu augstas klases VR ierīču, lietotājam vajadzēs investēt dārgā datorā, kas lielākajai daļai nav maza summa. Bet tā nav ar PSVR – *PlayStation VR*, kam vajag tikai PS konsoli, lai varētu iegrimt VR pieredzē. Ņemot vērā veikspējas atšķirību starp datoru un PS4, PSVR ir pārsteidzoši konkurējošas VR brilles. Tā atjaunošanas ātrums ir atsaucīgs un nav bijušas problēmas ar galvas kustību izsekošanu. (Best VR headsets 2020, 2020)

PlayStation VR tika izlaists tirgū 2016 gadā un ir tagad (2019. gada 15. janvārī) ir vispirktākās augstās klases VR brilles. Tas nav pārsteigums, ņemot vērā, cik kārdinošs ir pirkums, ja lietotājam jau ir PS4. Kādēļ gan tiekties pēc lielas veikspējas, istabas mēroga, saslēgtu ar vadiem brīļļu, ja ir iespējas pieslēgt ierīci pie savas konsoles zem televizora.

2018. gada augustā *Sony* svinēja trīs miljonu pārdotu VR brīļļu un 21.9 miljonu pārdotu VR spēļu.

Kā arī klīst baumas, ka PSVR2 iznāks kopā ar *PlayStation* jaunās paaudzes konsoli PS5. (PlayStation VR review, 2019)

4. Un sarakstu noslēgs ar labākajām viedtālruna VR brillēm – *Samsung Gear VR*.



1.4 attēls: Samsung Gear VR brilles ar viedtālruni (Best VR headsets 2020, 2020)

Plusi: viegls, atbalsta vairākus *Samsung* viedtālrunus

Mīnusi: dārgāks nekā *Google Daydream View*, nevar uzlādēt kontroliera baterijas

Samsung Gear VR visu laiku ir bijis respektēta viedtālruna-darbināta virtuālā realitāte, bet tagad tam iznāca kustību kontrolieris. Tagad tas varētu būt labākā VR ierīce viedtālruna lietotājiem. Neņemot vērā jauno kontrolieri, atjauninātais *Gear VR* ir vieglāks un racionālāks nekā iepriekš, un tam tagad ir USB-C ports, ar kuru var pievienot savu *Galaxy* viedtālruni.

Savienojamie viedtālruni ir *Galaxy S8* un *S8 Plus*, kā arī *Galaxy S9* un *S9 Plus* (ja ir jaunākās *Gear VR* brilles) – labākajai pieredzei, bet *Gear VR* darbosies arī ar visiem *S6* un *S7* modeļiem un *Note 5*.

Protams, esot piesaistītam pie viedtālruna veiktspējas, brillu veiktspēja ir tieši saistīta ar pievienotā viedtālruna jaudu. Bet tā nav problēma, ja izmanto *Galaxy* jaunākos augstās klases viedtālrunus. Ja izmantos vecākus, tad VR pieredze varētu būt jūtami ietekmēta.

Un tā kā *Oculus* laida klajā *Oculus Go* patstāvīgās brilles, rodas jautājums, kur iekļaujas viedtālruna darbināmas VR ierīces kā *Gear VR*, kļūst aktuālāks. (Best VR headsets 2020, 2020)

1.3.1 Virtuālās realitātes ierīču salīdzināšana

<i>Ražotājs, modelis</i>	<i>Valve Index</i>	<i>Oculus Quest</i>	<i>PlayStation VR</i>	<i>Samsung Gear VR</i>
<i>Iznākšanas datums</i>	Jūnijs/2019	Maijs/2019	Oktobris/2016	Novembris/2015
<i>Pieejamība*</i>	Ir	Nav	Jā	Jā
<i>Operētājsistēmas</i>	Windows10, SteamOS, Linux	Android	Playstation	Android
<i>Svars (brillu)</i>	809g	571g	600/610g	~318g
<i>Izšķirtspēja</i>	1440x1600	1440x1600	1920x1080	1280x1440**

<i>Displeja tehnoloģija</i>	LCD	OLED	OLED	OLED
<i>Atjauninājuma ātrums (Herci)</i>	80/90/120/144Hz	72Hz	90/120Hz	60Hz
<i>Redzes leņķis, grādi(FOV)</i>	135°	90°	100°	101°
<i>Savienojamība</i>	USB 3.0, DisplayPort 1.2	USB-C	HDMI, USB	USB-C, Micro USB
<i>Izsekošana</i>	Galvas, rokas, pirksti	Galvas, rokas, pirksti	Galva, rokas	Galva, rokas
<i>Sensori</i>	Bāzes stacijas un pirkstu sensori	Akselerometrs, žiroskops	Akselerometrs, žiroskops	Žiroskops, tuvuma sensors (proximity)
<i>Skana</i>	Iebūvētas austiņas, 3.5mm ports	Iebūvēti skaļruņi, 2x3.5mm porti	3.5mm ports	No viedtālruņa
<i>Mikrofons</i>	Iebūvēts	Nav	Nav	No viedtālruņa
<i>Cena, EUR</i>	1079	600 – 64GB, 650 – 128GB	280	120

1.1 tabula: VR ierīču salīdzinājums. Avoti: (Tech specs, 2020), (Oculus Quest, 2020), (Technical Specification, 2020), (Specification, 2020)

* - no ražotāja

** - maksimālā izšķirtspēja, atkarīga no viedtālruņa, kuru lieto

Liela konkurence ir starp *Oculus Quest* un *Valve Index*, jo ierīces ir laistas tirgū gandrīz vienlaicīgi. Sīkāk aplūkojot parametrus uzreiz izceļas *Index* ar ekrāna atjauninājuma ātrumu un redzes leņķi. Šie parametri ir stipri lielāki nekā citām ierīcēm, toties *Index* ir vienīgā ierīce bez OLED ekrāna, kas nodrošina īstākās krāsas, it īpaši melnās. Sākumā *Index* bija ļoti gaidīts, jo tā būs ierīce ar pirkstu izsekošanu, taču pēc kāda laika *Oculus Quest* uz šo atbildēja, atjauninot *Quest*, ka tas spēj ar bilžu atpazīšanu izsekot arī līdz pirkstiem. Protams, liela atšķirība starp šīm ierīcēm ir tāda, ka viena *Index* ir ar

vadiem, bet *Quest* ir bezvadu risinājums, kas ir ļoti parocīgāk. Tomēr pēc autora pieredzes, izstrāde ir vieglāka uz ierīcēm ar vadu, jo var testēt uz ierīces un skatīties no malas parametrus datora. Uz *Quest* šī opcija sākumā nebija pieejama, bet *Oculus* izlaida *Oculus Link*, kas savieno *Quest* ar datoru un šo problēmu atrisināja. Kā arī *Index* ir vienīgās VR brilles ar iebūvēto mikrofonu. Vienīgais sliktais parametrs no tabulas ir *Index* cena. Tā ir vislielākā no visiem. Pārējās aplūkotās ir vairāk vai mazāk līdzīgas, izņemot *Gear VR* ir zema izšķirtspēja un *Playsation VR* ir augsts atjauninājuma ātrums ekrānam.

1.2 Virtuālās realitātes pielietojums mūsdienās

Virtuālā realitāte kļūst ar vien populārāka un var redzēt, ka uzņēmumi sāk izmantot virtuālās realitātes risinājumus produkta vai paša uzņēmuma mārketingam ar vien biežāk. Kā arī vairāk dzird, ka virtuālo realitāti izmanto izklaidei – spēles, izlaušanās telpas un VR pieredzes, bet virtuālā realitāte nav tikai izklaide un izpriecās, to var arī izmantot ļoti nopietnās nozarēs, kas var palīdzēt pašam cilvēkam, vai izmantot personāla apmācībām, lai darbinieks varētu labāk izprast savu darbu un labāk rīkoties bīstamās situācijās, jo tādas jau ir piedzīvotas virtuālajā vidē. Tālāk tiks apskatītas sīkāk nozares, kuras sāk izmantot virtuālo realitāti.

1. Militārais dienests

Militārais dienests Apvienotajā Karalistē un Amerikas Savienotajās Valstīs izmanto virtuālo realitāti savās apmācībās, jo tas palīdz iziet cauri daudz dažādiem scenārijiem. Tas tiek izmantots visos dienesta atzaros.

VR ievieto apmācāmo daudz dažādās situācijās, vietās un vidēs, tāpēc militārais dienests to izmanto lidošanas simulācijām, kaujas simulācijai, mediķu apmācībai, dažādu transporta līdzekļu apmācībai. Nozīmīgs labums no virtuālās realitātes militārajam dienestam ir izmaksu samazināšana apmācībām. Papildus labums ir tas, ka var droši attēlot un piedzīvot bīstamas situācijas.



1.5 attēls: VR lidošanas apmācība Gaisa Spēku
Dienesta kadetam (Insinna, 2019)

2. Sports

Virtuālā realitātē atjauno sporta industriju gan spēlētājiem, gan skatītājiem. To izmanto kā trenēšanās palīgu daudzos sporta viedos un palīdz sekot līdzi atlētiskajai veikspējai un analizēt dažādas trenēšanās tehnikas.

Virtuālo realitāti arī izmanto, lai uzlabotu ne tikai spēlētāju, bet arī skatītāja pieredzi. Pārraidīšanas organizācijas tagad virtuālajā realitātē straumē tiešsaistē dažādas spēles un gatavo kādu dienu pārdot virtuālas biļetes uz reālām spēlēm.



1.5 attēls: Futbola spēli skatās virtuālajā realitātē
(Omnivirt, 2018)

3. Garīgā veselība

Virtuālā realitātē ir kļuvusi par primāro metodi, lai ārstētu pēc-traumas stresu (angl. *post-traumatic stress*). Izmantojot VR terapiju, cilvēks vēlreiz izdzīvo traumatisku notikumu.

Virtuālo realitāti arī izmanto, lai ārstētu trauksmi, dažādas fobijas un pat depresiju. Ar VR tehnoloģiju tiek panākta droša vide, kur pacienti var mijiedarboties ar lietām, kas viņus biedē, bet tajā pašā brīdī paliekot kontrolētā un drošā vidē.



1.6 attēls: Fobiju ārstēšana ar VR palīdzību (virtualrealityandphobias, 2016)

4. Medicīniskā apmācība

Medicīnas un zobārstu studenti izmanto virtuālo realitāti, lai uzkrātu pieredzi operācijās un procedūrās, atļaujot mācīties bez traumatiskām un nāvējošām sekām. Virtuālie pacienti tiek izmantoti, lai studenti varētu uzkrāt pieredzi, kuru pēc tam izmantos īstajā pasaulē un uz īstiem cilvēkiem.



1.7 attēls: VR pielietojums zobārstu apmācībai
(Admin, 2019)

5. Izglītība

Virtuālā realitāte tiek izmantota mācību procesos – gan no skolotāja puses, gan no studenta/ skolēna puses. Studentiem ir iespēja mijiedarboties vienam ar otru trīs dimensionālā vidē. Studenti tiek arī virtuāli vesti ekskursijās, piemēram, uz muzejiem, apskata Saules sistēmu tuvāk un aizceļo uz citām ērām.

Studenti ar īpašām vajadzībām, piemēram, autismu, arī izmanto virtuālās realitātes tehnoloģiju. Pētījumi ir atklājuši, ka VR var būt motivējoša platforma, lai bērni varētu droši trenēt sociālo pieredzi. Uzņēmums “Floreo” ir izveidojis virtuālās realitātes scenārijus, kas atļauj mācīties un uzlabot tādas prasmes, kā objektu atpazīšanu, acu kontaktu un veidot sociālas saiknes. Bērnu vecāki var šim procesam sekot līdzi izmantojot planšeti, kas ir saslēgta ar bērna virtuālo pieredzi. (5 Uses for Virtual Reality, 2019)



1.8 attēls: VR pielietojums skolās (Walsh, 2017)

Kā jau autors bija minējis sākumā, tad virtuālā realitāte nodrošina ne tikai izklaidi, bet reāli palīdz cilvēkiem un viņu veselībai, vai arī palīdz uzņēmumiem efektīvāk un lētāk, kā arī interesantāk, apmācīt darbiniekus. Un šo darbinieku prasmes var būtiski ietekmēt apkārtējos cilvēkus vai klientus, tāpēc ir ļoti svarīgi, lai darbiniekiem ir pēc iespējas lielāka pieredze un ir gatavi dažādām situācijām, lai zinātu kā labāk ar tām tikt galā.

1.3 Virtuālā realitāte skaitļos

Virtuālās realitātes industrija attīstās strauji, ar tirgus daļu patērētāju virtuālās realitātes iekārtās un programmatūras paredzēto pieaugumu no 6.2 miljardiem Amerikas dolāriem 2019. gadā līdz vairāk kā 16 miljardiem Amerikas dolāru 2022. gadā.

Tiek prognozēts, ka 2020. gadā globālais papildinātās un virtuālās realitātes tirgus apjoms būs 18.8 miljardu Amerikas dolāru. Valstis, kuras visvairāk tērēs naudu uz VR un AR tehnoloģijām ir Ķīna un Amerika. Kā arī prognozēts, ka 2020. gadā patērētāji iztērēs 7 miljardu Amerikas dolāru saistībā ar AR un VR. (Virtual Reality (VR) - Statistics & Facts, 2020)

Pēc autora domām šie skaitļi būs mazāki, saistībā ar vīrusa COVID-19 epidēmijas izveidoto ekonomisko krīzi. "...Facebook sākotnēji plānoja, ka paziņos par *Quest 2* šogad konferencē "*Oculus Connect*". Šī konference parasti ir septembra beigās vai oktobra sākumā, bet tagad, nosūtījumu aizkavējumu un loģistikas problēmu dēļ, konference var nenotikt līdz 2021. gadam." (Pino, 2020). Šāds aizkavējums var izraisīt būtiskus zaudējumus *Oculus*.

1.4 Virtuālās realitātes spēļu analīze, to tipi un pieejamība mūsdienās

Lai labāk saprastu tālāk apskatīto informāciju, ir jāsaprot kādi ir spēļu žanri un ko tie nozīmē:

- Rīcības (angl. *action*) – jebkura spēle, kur spēlētājam ir jāuzveic fiziski uzdevumi – tādi, kas ir saistīti ar precizitāti un ātru reakcijas laiku. Var ietvert sevī sacensības, mīklas vai objektu vākšanu, bet tā nav spēles galvenā daļa. Spēlētāji var arī sastapties ar taktiskiem un izpētes izaicinājumiem. Spēlētājs bieži tiek pakļauts zem laika spiediena, tāpēc nav laika ilgai sarežģītai domāšanai. (Next Generation, 1996)



1.9 attēls: Rīcības spēle (Tolin, 2020)

- Piedzīvojumu (angl. *adventure*) – spēlētājs uzņemas galvenā varoņu lomu interaktīvā stāstā, kuru virza izpēte un dažādu pužļu un mīklu atminēšana. (Game Design, 2003)



1.10 attēls: Assassin's Creed Valhalla
(news.game.co.uk, 2020)

- Ikddienišķa (angl. *casual*) – attiecinā uz spēlēm, kurās nevajag ieguldīt milzīgi daudz laika, lai spēlētu, uzvarētu un izbaudītu. Ikddienišķs spēlētājs ir spēlētājs, kam patīk jebkura spēle bez liela laika ieguldījuma, spēlējot to spontāni un neregulāri jeb pa retam. (Casual Gaming, 2018)



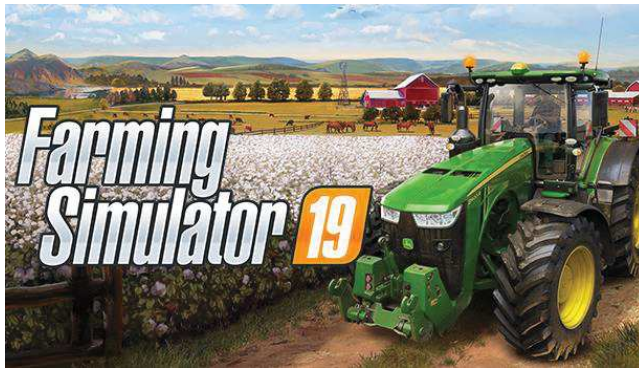
1.11 attēls: Hiper-ikddienišķas spēles uz telefona (Özistek, 2019)

- Šaušanas (angl. *shooter*) – atļauj spēlētājam rīkoties no distances izmantojot dažādus ieročus, izaicinot tos mērķēt precīzi un ātri. Bieži vien šaušanas spēles ir saistītas ar armijas konfliktus gan vēsturiskus, gan izdomātus, kur Otrais Pasaules Karš ir viena no populārākajām tēmām. Bet ne vienmēr ir saistītas ar karu un armiju, tās var iekļaut arī medības, sekot līdz noziedznieka stāstam. (Fundamentals of Game Design, 2006)



1.12 attēls: Call of Duty: Warzone (Winslow, 2020)

- Simulācijas (angl. *simulation*) – daudzveidīga superkategorija. Izstrādātas tā, lai pēc iespējas precīzāk attēlotu reālās pasaules aktivitātes vai loģiku. (Next Generation, 1996)



1.13 attēls: Farming Simulator 2019 (Steam, 2018)

- Mīklas (angl. *puzzle*) – mīklu spēles koncentrējas uz loģiku un konceptuāliem izaicinājumiem. Daudzās rīcības spēlēs un piedzīvojumu spēlēs ir pužļu elementi, bet īstā mīklu spēlē koncentrēsies uz mīklu atrisināšanu kā galveno spēles elementu. (Fundamentals of Game Design, 2006)

Aplūkojot, vienu no vispopulārākajām spēļu platformas, *Steam* VR spēļu piedāvājumu, var uzreiz secināt, ka vispirktākās spēles ir rīcības un piedzīvojumu spēles. No 10 vispirktākajām VR spēlēm 7 pieder pie rīcības žanra un 5 pie piedzīvojumu žanra. Otrā vispirktākā ir spēle “Beat Saber” un autora domās viena no viszināmākajām VR spēlēm šajā sarakstā. Šī ir spēle, kur ir ļoti vienkārši pamati – jāizmanto roku kustības, lai mūzikas ritmā ar diviem gaismu zobeniem trāpītu pa notīm. Ļoti vienkārša, bet ārkārtīgi aizraujoša spēle, kuru var spēlēt ļoti vieglā līmenī tie, kas pirmo reizi izmanto VR un kuru var spēlēt jau ļoti pieredzējuši VR lietotāji. *Steam* 10 vispirktākās spēles ir:

1. Half-Life: Alyx,
2. Beat Saber,
3. The Room VR: A Dark Matter,
4. Boneworks,
5. The Walking Dead,
6. Blade and Sorcery
7. Pavlov VR,
8. The Elder Scrolls VR
9. Rick and Morty: Virtual Rick-ality

10. Zero Caliber VR.

Šo spēļu vidējā cena ir 38.2 EUR, kur dārgākā ir *The Elder Scrolls* ar cenu 60EUR un lētākā ir *Blade and Sorcery* ar cenu 20EUR.

Kā jau tika minēts, tad populārākās ir rīcības un piedzīvojumu spēles, bet ir arī ritmiskas mūzikas spēles, mīklu un simulācijas spēles. Kā arī gandrīz visās, precīzāk 8 no 10, spēlēs ir iekļauta šaušana, cīnīšanās ar zobeniem vai maģiju.

Visas spēles atbalsta *HTC Vive*, *Oculus Rift*, *Windows Mixed Reality* un *Valve Index* ar vienu izņēmumu – *Rick and Morty* neatbalsta *Valve Index*.

Autors nākamo apskatīja *Oculus Quest* spēļu veikalu. Vispirktākās 10 spēles ir:

1. Superhot VR (rīcības, mīklas, šaušana),
2. Beat Saber (mūzika, sports, ikdienišķa),
3. Arizona Sunshine (rīcība, šausmas, šaušana),
4. The Room VR: A Dark Matter (piedzīvojumu, mīklas),
5. Color Space (māksla, ikdienišķa, relaksējoša),
6. Job Simulator (ikdienišķa, simulācija),
7. Gun Club VR (rīcības, šaušana, simulācija),
8. Red Matter (piedzīvojumu, mīklas),
9. Eleven Table Tennis (sports, simulācija),
10. Vader Immortal: Episode I (piedzīvojumu, stāstījums).

Šajā topā nav vadošās kategorijas. Ir seši populārākie žanri, pie kurām bija vienāds skaits spēļu – rīcības, mīklas, šaušana, piedzīvojumu, simulācijas un ikdienišķas.

Oculus Quest tirgū 10 pirktāko spēļu vidū visdārgākā spēle ir *Arizona Sunshine* ar 40EUR, vislētākā ir *Vader Immortal* ar 8EUR. Vidējā spēles cena ir 28.8EUR.

Šīs spēles ir lietojamas tikai *Oculus Quest* ierīcēm. Ar citiem vārdiem sakot, šīs ir spēles, kuras var dabūt uz ierīces. No citām vietām nav iespējams nopirkt spēles.

Diemžēl *Playstation* tirgū nevarēja atlasīt spēles pēc pirkumu skaita vai popularitātes, tāpēc autors izmantos saita uploadvr.com izveidoto top25 *Playstation* VR spēles:

1. Astro Bot Rescue Mission - rīcības
2. Firewall: Zero Hour - šaušana
3. Superhot VR – rīcības, šaušana, piedzīvojums
4. Wipeout: The Omega Collection - sacensību
5. Ghost Giant – piedzīvojumu, mīklas
6. Resident Evil 7: Biohazard – rīcības, piedzīvojumu, šausmu
7. Skyrim VR – lomu spēle, piedzīvojums, rīcība
8. Beat Saber – mūzika, ritms
9. A Fisherman's Tale – piedzīvojumu, mīklas
10. Blood & Truth – rīcības, piedzīvojumu

Šis tops ļoti līdzinās ar *Steam* spēļu topu, skatoties pēc vadošajiem žanriem – populārākie ir piedzīvojumu, ar sešām spēlēm, un rīcības žanrs ar 5 spēlēm. Dārgākā spēle ir Skyrim ar 48EUR un lētākā ir Fisherman's Tale ar 12EUR. Vidējā spēles cena topā ir 24EUR.

Pēc šiem trīs spēļu sarakstiem var secināt, ka vispopulārākās spēles, kas piesaista ir rīcības, piedzīvojumu, šaušanas un mīklas. Pēc autora domām, šie žanri ir topā, jo šo spēļu mehānika perfekti iederas virtuālajā realitātē – jākustas ar rokām un ķermeni, lai paceltu objektus, ieročus, ar rokām jātēmē un jāšauj. Var mijiedarboties ar objektiem un reāli liekas, ka tos turi rokās. Spēlētājam tas uzreiz dod pilnīgi citu pieredzi, nekā, ja šauj, piespiežot pogu uz datora peles vai tēmē un skatās riņķī kustinot datora peli. Tā vietā labāk izmantot savas rokas un galvu.

Darba autors izvēlējās izlaušanās telpas spēles izveidi, jo, pirmkārt, izlaušanās telpas ir kļuvušas ļoti populāras un cilvēkiem tās ļoti patīk – tās ir jautras, satuvina cilvēkus, liek cilvēkiem izdzīvot ne ikdienišķas situācijas kamēr ir jārisina dažādi grūti uzdevumi zem laika spiediena. Un, otrkārt, jo virtuālā realitātē sniedz iespējas būt pavisam kādā neiedomājamā vietā, vai darīt neiedomājamas lietas, piemēram, lai atvērtu durvis ir jāveic noteikta rokas kustība, lai aktivizētu kādu burvestību, vai spēlētājs atrodas kosmosa kuģī. Ar virtuālo realitāti var uzlabot tā jau aizraujošo izlaušanās telpu ar virtuāliem elementiem, kas nebūtu iespējami īstā izlaušanās istabā.

2. VR izlaušanās telpu spēļu funkcionalitātes salīdzinājums

“Tev ir viena stunda, lai sameklētu pavedienus, atrisinātu mīklas un atrastu atslēgu, kas atvērs šīs durvis. Vēlu veiksmi!” – tāds teksts idejiski tiek sniegts spēlētājiem spēles sākumā, kur tiek pastāstīti galvenie uzdevumi, tas ir, spēles laiks un, kas jāizdara, lai tiktu ārā no telpas jeb uzvarētu spēli.

Izlaušanās telpas, arī zināmas kā izlaušanās spēles, ir spēle, kur spēlētāju komanda strādā kopā, lai atklātu pavedienus, atrisinātu mīklas un mīklas, un izpilda uzdevumus vienā vai vairākās telpās, tādējādi tiekot tālāk un galu galā izpildot noteiktu uzdevumu noteiktā laika posmā.

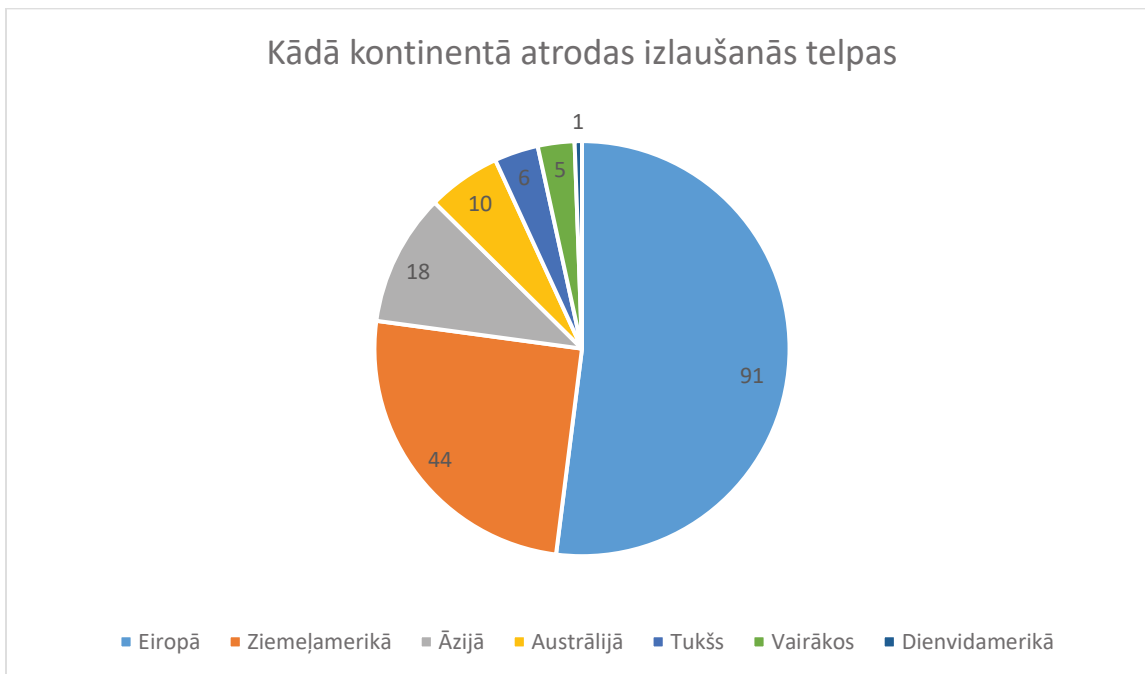
Spēlētāji izpēta istabu, sākumā īsti nezinot pat ko meklē, bet, kad laiks jau kļūst mazāk, spēlētāji sāk nopietnāk meklēt pavedienus visur kur vien iespējams. Parasti ir atrodamī skaitļi, simboli vai bildes pie sienām, bet nav saprotams, kur šo informāciju izmantot un kam tieši tā ir derīga. Pirmā izlaušanās telpu pieredze ir izmeklēt visas atvilktnes, kastes, kabatas, virs un zem objektiem. Komandas ar labu sadarbību paziņo, ja kaut kas tiek atrasts un mēģina kopā salikt mīklas. Kādā brīdī spēlētāji uzdursies mīklai un mēģinās saprast, kā ar to tikt galā – dažām mīklām ir norādījumi apmēram kā to izpildīt un dažām nav dotas nekādas papildus informācijas. Vienas mīklas atrisinājums, var aizvest pie kā cita – tas varētu būt kods slēdzenei, sākums citai mīklai, durvīm, kas atver citu telpu, maza daļa no lielās mīklas. Ja spēlētāji iesprūst, tad parasti tiem palīdz darbinieks, pasakot mājieni pareizajā virzienā. Šis darbinieks visam seko līdz caur kamerām. Ar laiku mīklas paliek ar vien grūtākas, bieži vien, esot daļa no kopējās lielās mīklas, kuras gala atrisinājums apbalvos spēlētājus ar kodu vai atslēgu, vai varbūt kādu specifisku veidu (angl. *pattern*) kā izkļūt no telpas un uzvarēt.

Izlaušanās istabām ir nepieciešama komunikācija, komandas sadarbšanās, kā arī kritiskā domāšana, jāpievērš uzmanība detaļām un jābūt laterālai domāšanai. Šīs izlaušanās spēles ir domātas plašam spēlētāju vecumam un nedod priekšroku noteiktam dzimumam; starp citu, veiksmīgākās komandas parasti sastāv no tādiem spēlētājiem, kuriem ir dažādas

pieredzes un prasmes, fiziskās dotības un dzīves pieredze. (Peeking Behind the Locked Door: A survey of escape room facilities, 2015)

Balstoties uz Skota Nikolosa (Scott Nicholson) veikto pētījumu, kur viņš veica aptauja 175 izlaušanās telpās apkārt visai pasaulei. Darba autors vēlētos dalīties ar S. Nikolosa iegūtājiem aptaujas datiem.

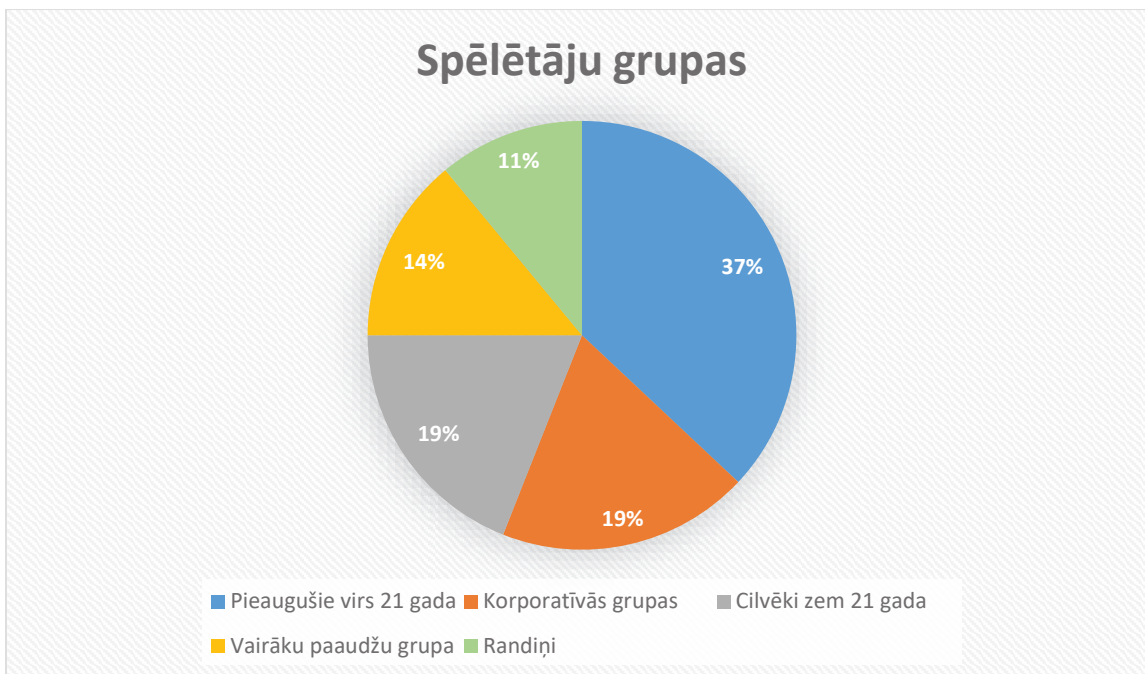
Kopā bija 175 izlaušanās telpu uzņēmumi, kas veica aptauju.



2.1 attēls: Uzņēmumu iedalījums pēc kontinenta

Pēc datiem grafikā 2.1 var secināt, ka lielākā daļa, kas ir mazliet vairāk kā puse no visiem uzņēmumiem atrodas Eiropā.

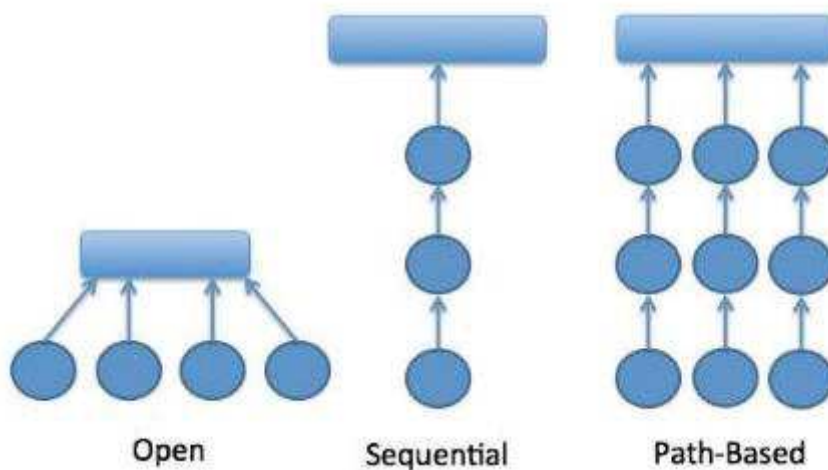
Tālāk, 2.2 grafikā tiks apskatīts, kādas cilvēku grupas apmeklē izlaušanās telpas. Ar vislielāko skaitu pirmajā vietā ierindojas pieaugušie virs 21 gadiem. Otrā un trešā populārākā apmeklētāju grupa ir korporatīvās grupas un grupas ar cilvēkiem zem 21 gada.



2.2 attēls: Spēlētāju grupas izlaušanās istabās

2.1 Mīklu hierarhija

Ir dažādi veidi, kā mīklas var būt sakārtotas. Tās var būt pasniegtas individuāli, kur katra mīkla ir daļa no lielās mīklas, vai arī secīgi, kur viena mīkla ir jāatrisina, lai tiktu pie kaut kā, kas ir nepieciešams nākamajai mīklai. Piemēri šiem veidiem tiks attēloti 2.1.1 attēlā. Attēlā redzami apli ir mīklas un četrstūri ir cita lielāka mīkla vai slēdzene, vai jebkurš cits uzvaras nosacījums posmam vai visai spēlei.



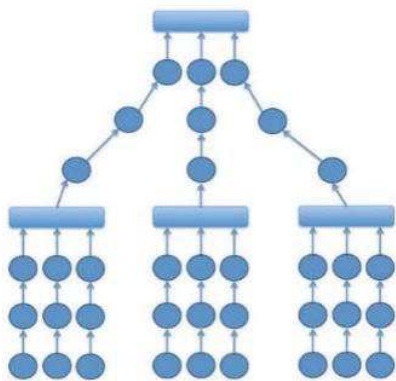
2.1.1 attēls: No kreisās uz labo pusi – Atvērtais jeb tiešais, secīgais, ceļu balstīts veids.

Visbiežāk izmantotais mīklu organizācijas veids ir ceļu balstīts (45%), kur komanda tiek iepazīstināta ar dažādiem mīklas ceļiem uzreiz. Katrs mīklas ceļš ir noteiktā secībā, kas beigās aizved pie gala rezultāta. Katra ceļa beigu rezultāts ir nepieciešams, lai atbloķētu gala mīklu vai uzvaras nosacījumu. Šīs metodes pārkums pār citām metodēm ir tāds, ka komandas locekļi var darboties ar dažādām mīklām vienlaicīgi. Parasti sākumā ir vieglākas mīklas, lai spēlētājs varētu saprast kā šī spēle strādā un kļūt drošāks un pazīstamāks ar spēli. Pēc autora domām, šis ir populārākais veids, jo parasti spēlē komanda, nevis viens vai divi cilvēki.

Otrajā vietā ir secīgais veids, kuru izmanto 37% spēlēs. Šajā metodē spēlētājiem ir dota viena mīkla, kuru atminot tiek atbloķēta nākamā mīkla. Kad atmin posma pēdējo mīklu, tad spēlētāji ir uzvarējuši. Šī metode strādā labāk, kad ir maza telpa vai arī kad mīklai ir nepieciešams, lai pie tās strādā visa komanda, lai to atminētu.

Vismazāk izmantotā metode ir atvērtā metode ar 13% pielietojumu. Spēlētājiem tiek dota iespēja minēt vairākas lielas mīklas vienlaicīgi telpā. Katras mīklas atrisinājums ir daļa no gala atrisinājuma. Šo metodi ir grūtāk izmantot, ja vēlas veidot plūstošas pieredzes, kur spēle kļūst grūtāka, ja tiek tālāk.

Ir dažas spēles (5%) ar daudz sarežģītāku mīklu izkārtojumu. Šīs metodes apkopo vairākas metodes savā modelī. Piemēram, kur komanda var sākt ar dažām secīgām mīklām un tad atbloķē vairākas citas mīklas, kas pārvēršas par ceļu balstītu metodi. Jo komanda jau ir apmēram sapratusi kā strādā istaba un ir iesildījušies. Šāda izdomātu hibrīda modeli var aplūkot 2.1.2 attēlā. Tas ir tikai viens no iespējamiem veidiem, kā šos trīs lielos mīklu veidus var savienot. (Peeking Behind the Locked Door: A survey of escape room facilities, 2015)



2.1.2 attēls: Piramīdas organizācijas veids

2.2 Izlaušanās spēļu mīklu veidi

Izlaušanās spēlēs mīklu veids ir daudzveidīgs un vienīgs, kas tās ierobežo, ir veidotāja atjautīgums un iztēle, bet tās var sakārtot lielākās kategorijās. Tabulā 2.2.1 varēs aplūkot visbiežāk izmantoto mīklu veidus. Procenti nozīmē to pielietojums daudzums spēlēs. Dati iegūti no jau iepriekš apskatītā Skota Nikolosa veiktā pētījuma.

<i>Mīklu veids izlaušanās telpās</i>	<i>Pētījuma atbiles (%)</i>
<i>Meklēt apslēptus objektus</i>	78
<i>Komandas komunikācija</i>	58
<i>Manipulācijas ar gaismu</i>	54
<i>Skaitīšana</i>	53
<i>Pamanīt kaut ko “acīmredzamu” telpā</i>	49
<i>Simbolu aizstāšana ar atslēgu (piem., skatīties simbolus no grāmatas kādas)</i>	47
<i>Ārpus rāmja domāšana</i>	47
<i>Meklēt objektus attēlos</i>	43
<i>Salikt kopā objektu (piem., puzli)</i>	40
<i>Matemātiskas operācijas</i>	39
<i>Rakstu (angl. pattern) identificēšana</i>	38
<i>Mīklas</i>	37
<i>Šifri bez atslēgas (piem., burtu aizstāšana)</i>	35
<i>Klausīšanās</i>	26
<i>Spoguļi</i>	26

2.2.1 Tabula: Pužļu veidi izlaušanās spēlēs

2.3 VR izlaušanās spēles

VR izlaušanās spēļu galvenā atšķirība no parastas izlaušanās istabas ir tāda, ka spēlētājs parasti atrodas kādā neikdienišķā vai reālā pasaulē neeksistējošā vidē. Apskatot, populārākās VR izlaušanās spēles, tad daudzas bija zinātniskās fantastikas vidē, dažas bija šausminošā vidē vai kosmosā. Ir arī parastākas vides, kā ofiss, mašīna, vilciens, mājās.

Objektu kontrole ir vienkārša, taču atšķirībā no spēles var mainīties iespēja apskatīt objektu reālāk, tas ir, dažām spēlēm objekts paņemams rokās fiksētā pozīcijā un rotācijā. Autors uzskata, ka tas samazina īstenības piegaršu un var izsist mazliet spēlētāju no VR pieredzes. Bet par laimi vairākums spēļu atbalsta, ka objektu var paņemt aiz tās vietas, kur spēlētāja roka tiešām arī atrodas. Kā arī autoram uzreiz acīs iekrita, ka spēlēs dažās izmanto kontrolierus, lai attēlot spēlētāju rokas. Lai būtu pēc iespējas labāka VR pieredze, būtu labāk jāizmanto 3D roku modeļi, lai izskatītos īstāk un patīkamāk.

2.3.1 VR izlaušanās spēļu analīze

Šīs izvēlētās spēles ir top izlaušanās spēles virtuālajā realitātē. Sāksim ar spēli, kas dominē vairāku platformu pirktaكو spēļu topu. Pārējās spēles tika atlasītas *Steam* veikalā pēc atslēgvārdiem VR un mīklas, un izvēlētas ir visvairāk pirtās spēles.

1. “The Room VR: A Dark Matter”

Šīs spēles laikmets ir maģisks veco laiku stils, precīzāk 1908. gads. Spēlētājam ir jāizmeklē, kas notika ar pazudušu personāžu. Spēlē nevar brīvi pārvietoties pa spēles telpu, ir noteikti punkti uz kuriem spēlētājs var teleportēties. Viss notiek vairāk kā vienā telpā. Šī spēle, salīdzinot ar citām, ir ļoti apjomīga un grafika arī ir ļoti laba. Spēles vidējais ilgums ir viena stunda. Ar daudz ko apkārtējā vidē var mijiedarboties un mijiedarbība ir ļoti precīza, piemēram, ar roku griežot slēdžus, personāža roka to uzliek uz slēdža un pagriež, tāpat kā to darītu īstenībā. Spēlei ir ļoti laba un īsta skaņa, kā arī animācijas ir ļoti augstas kvalitātes un tuvas reālām kustībām. Spēlē uzdevumi sākās ar salīdzinoši vienkāršu kodu atklāšanu, kas ir paslēpts blakus slēdzenei uz tāfeles rakstītajā tabulā. Jāsaprot sakarība starp vārdiem un kastes numuriem. Tālāk spēlē ir plašāki un grūtāk saprotami uzdevumi. Autors arī vēlētos piebilst, ka personāža rokas ir roku 3D modeļi, nevis kontrolieru 3D modeļi.

2. “I Expect You To Die VR”

Kā jau spēles nosaukums paziņo, tad spēle sagaida, ka spēlētājs nomirs. Tas viennozīmīgi notiks, jo spēle balstās uz izmēģini un kļūdies mehānismu (angl. *trial*

and error) – spēlētājam tiek dotas vairākas dzīvības, lai saprastu kāda ir pareiza secība notikumiem un ko noteiktā situācijā darīt. Piemēram, šajā spēlē spēlētājs ir mašīnā, kura ir lidmašīnā. Un spēlētājam ar šo mašīnu ir jāizbrauc no lidojošas lidmašīnas. Spēlētājs, skatoties ārā pa mašīnas logu, var redzēt, ka lidmašīnas telpā tiek ielaista zaļa, indīga gāze un uz aizmugurējā sēdekļa ir pistole. Ja spēlētājs izšauj cauri stiklam, tad spēlētājs nomirst un līmenis sākās no sākuma. Un pēc šī spēlētājs zina, ka tā vairs nevajag darīt. Spēlei ir ļoti laba grafika, bet tā nav realistiska. Spēlē var gandrīz ar visu mijiedarboties, un stāsts attīstās interesanti. Ar spēlētāju mēdz runāt spēlētāja “priekšnieks”, stāstot, ko dara nepareizi vai mazliet pasmejoties par to, ko spēlētājs pamēģināja izdarīt, kā arī dažreiz dodot mājienus, ko vajag darīt ar tikko iegūtu objektu. Piemēram, kad spēlētājs atrod spridzekli, spēlētājam pasaka, ka vajag to atminēt. Spēles skaņa ir ļoti labi, tā ir patiesa, ir visiem objektiem un visām mijiedarbībām. Personāža rokas ir 3D roku modeļi, nevis kontrolieri. Spēles uzdevumi nav sarežģīti kā iepriekš apskatītajā spēlē. Tie ir īsi un katra uzdevuma atrisināšana palīdz tikt tālāk stāstā, kas noved pie uzvaras.

3. “Tales of Escape”

Šī spēle ir unikāla ar to, ka to var spēlēt līdz sešiem cilvēkiem. Un šī spēle atbalsta, ka kopā var spēlēt no dažādām platformām. Atbalstītās platformas ir PC, Oculus un Vive. Viens spēlētājs, kurš uzsāk spēli, var dalīties ar kodu, kuru citi spēlētāji ievadot, var pievienoties šim spēlētājam, lai kopā spēlētu. Bet spēli var arī spēlēt vienatnē, ja vēlas. Spēlē ir šausmu elementi, fonā skan neomulīga mūzika un tā kā spēlētājs atrodas raktuvē, tad ik pa brītiņam ir kāda akmeņu krišanas skaņa attālumā. Grafikas stils ir reāls, bet kvalitāte nav tā labākā. Spēlē kopā ir četras telpas. Un stāsts attīstās lineāri – spēlētājs ir katrā telpā pēc kārtas, kad izpilda iepriekšējo. Uzdevumi ir interesanti un atjautīgi. Nav pārāk sarežģīti uzdevumi, bet dažreiz grūti saprast, ko spēle vēlas no spēlētāja. Spēles sākums bija pats sarežģītākais, jo ar daudz ko nevarēja mijiedarboties un bija grūti saprast, ko spēle grib, lai spēlētājs dara. Šajā spēlē personāža rokas arī ir roku 3D modeļi.

4. ” Abode”

Spēles grafiskais stils ir ar nākotnes tehnoloģiju elementiem – hologrāfiski ekrāni, lidojoši droni iekštelpās u.c. Bet grafika pati par sevi ir minimālistiska (angl. *low-poly*) veida. Abode no pārējām atšķiras ar to, ka ir jāatrod mazi čipi, kuru dators nolasa un parāda hologrāfisku tekstu ar pavedienu. Šie pavedieni noved pie citiem uzdevumiem. Spēlē var mijiedarboties ar pa lielam visu. Spēlei ir pozitīva atmosfēra un spēlētājs netiek pakļauts laika stresam – var mierīgi visu izpētīt un izdomāt. Objektiem ir vienkāršas animācijas un ne visām objektu mijiedarbībām ar citiem objektiem ir skaņa. Dažreiz tas var mazliet izsist spēlētāju no pieredzes. Bet kopumā vienkārša stila nomierinoša, ar pozitīvu atmosfēru, spēle. Personāžam rokās ir kontrolieru 3D modelis, bet tas ir nomainīts, lai pieskaņotos spēles grafiskajam stilam.

5. “Belko VR: An Escape Room Experiment”

Šī ir interesanta, īsa izlaušanās spēle, kur darbiniekam ir jāizklūst ārā no ofisa. Ar spēlētāju runā kolēģis no blakus ofisa un dod mājienu, kas ir jādara, ja kādu ilgāku laiku netiek izpildīts stāstā noteiktais uzdevums. Uzdevumi ir vienkārši un spēli var iziet 10-15 minūšu laikā. Spēles grafikas stils ir reāls un spēles grafikai ir laba kvalitāte. Kā arī spēles skaņa ir ļoti laba, nebija momentu, kad notika kāda mijiedarbība un nebija skaņas. Vienīgais, kas darba autoram nepatika šajā spēlē, ir tas, ka personāža rokas ir 3D kontrolieri. Kā jau iepriekš tika minēts, tas samazina VR pieredzes kvalitāti. Spēles objektu animācijas ir vienkāršas, bet pietiekošas spēles stilam. Vienīgā animācija, kas ir sarežģīta, ir kolēģa staigāšana un runāšana. Šo kolēģi spēlētājs redz ofisa monitorā. Un šis kolēģis ik pa laikam skraida apkārt un pie monitora pieskrien. Šo animāciju kvalitāte ir ļoti laba.

Autors pats izmēģināja virtuālajā realitātē spēles “Tales of Escape” un Belko VR, jo tās bija par velti pieejamas.

2.4 Autora veidotās spēles pamatojums un informācija

Autors izvēlējās veidot savu virtuālās realitātes izlaušanās spēli, jo Latvijā nav virtuālās realitātes izlaušanās telpu uzņēmumi. Tā kā darba autoram bija sadarbība ar

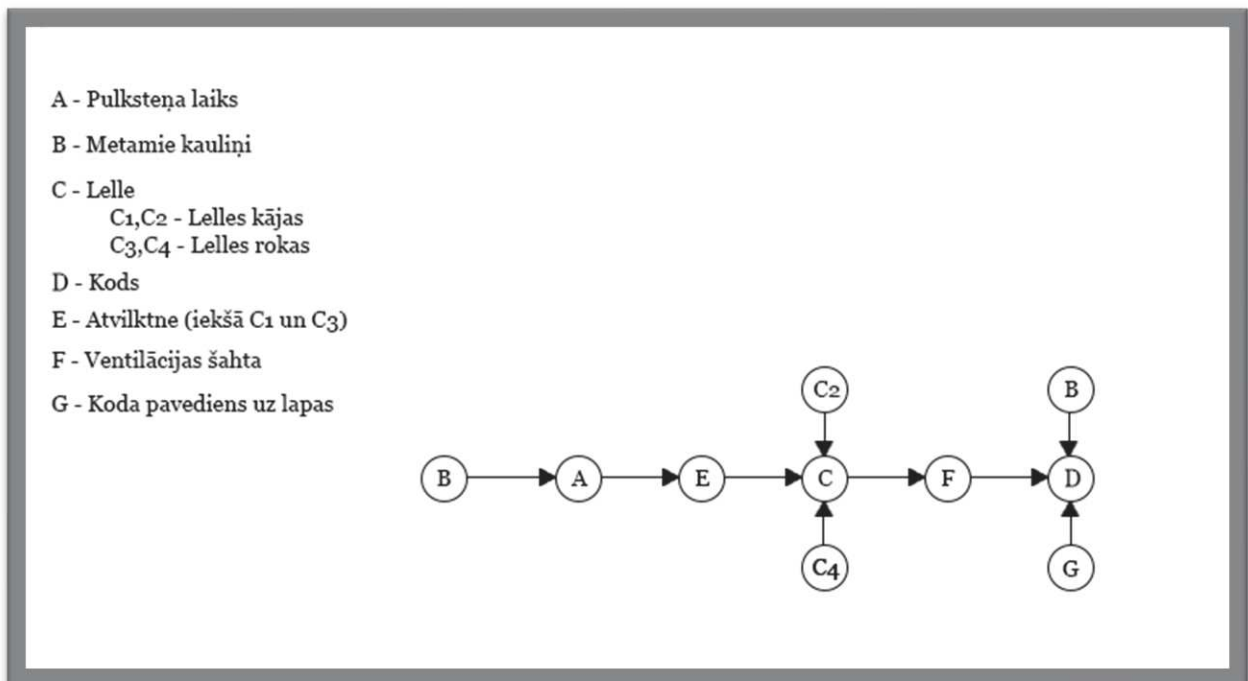
Biznesa Vadības virziena studenti, kura veica investīciju projektu virtuālās realitātes studijai, tad radās ideja – piedāvāt klientiem Latvijā veidotas VR spēles. Kā arī Latvijas VR studijas vai Latvijas izlaušanās istabu uzņēmumi nepiedāvā Latvijā veidotas spēles saviem klientiem. Viens no diviem uzņēmumiem, kas piedāvā VR izlaušanās spēles ir “PortālsVR”. Šis uzņēmums piedāvā lielu klāstu ar VR spēlēm, bet tagad sadarbojās ar spēļu veidošanas gigantu “Ubisoft”, lai klientiem būtu iespējams spēlēt dažas VR izlaušanās spēles, kuras veido “Ubisoft”. Otrs uzņēmums piedāvā VR izlaušanās spēli bērniem. Uzņēmuma nosaukums ir “Escape Room: Slepena istaba”. Autors izvēlējās arī tāpēc, ka izlaušanās spēles ir ļoti jautras, kur ir jādomā, lai tās izietu, nevis jāšauj kā citās spēlēs. Un tādas spēles, kur šauj ir daudz, bet tādas spēles, kur ir jādomā un jāpilda uzdevumi mierīgā vai, tieši pretēji, uztrauktā atmosfērā, ir maz.

3. Izlaušanās spēles izstrādes pamatposmi

Autors veido izlaušanās telpas prototipu izmantojot populāru spēļu dzinēju *Unity*. Darba autors pats ir taisījis visus 3D modeļus, izmantojot *Unity* piedāvātu spraudni *ProBuilder*, kas ļauj ērti pašam modelēt turpat iekš *Unity*. Spēlētāja un objektu mijiedarbību nodrošina spraudnis *VRTK*. Tas ir oficiāls spraudnis, kas nodrošina mijiedarbību, spēlētāja kustību virtuālajā vidē.

Autora pirmais uzdevums bija izdomāt izlaušanās vietu un stāstu, kā arī uzdevumus, kurus iekļaut izlaušanās spēlē.

Autors izdomāja, ka spēle notiks parastā ģimenes mājā, kur māte savu dēlu sūta uz savu istabu. Spēles personāžam ieejot savā istabā, aiztaisās telpas durvis un personāžam uzraksta draugs, lai nāk ārā. Tad spēlētājam ir jāizdomā, kā attaisīt durvis un tikt ārā.



3.1 attēls: Uzdevumu hierarhija

Darba autors izstrādi sāka ar 3D vides izveidi. Lai būtu interesantāks sākums, nevis parasta sākuma izvēlne (angl. *main menu*), uz kuru jāspiež pogas, tad autors vēlējās iekļaut neredzamās pamācības un sākt spēli mazā gaitenī, kas kalpos kā galvenā izvēlne. Šajā gaitenī spēlētājs varēs pārvietoties un mijiedarboties ar dažādiem objektiem, bet, lai sāktu spēli, vajadzēs paņemt noteiktu objektu un to aiznest uz istabu. Šie pāris objekti iemācīs spēlētājam pamata kontroles, lai mijiedarbotos ar objektiem. Prototipā tas nebūs iekļauts, taču idejiski personāža māte no pirmā stāva pabļaus, lai paņem šo objektu un iet uz istabu. Tādējādi nav tieši pateikts spēlētājam ar tekstu vai kādiem uzrakstiem, bet gan pateikts, kas jādara, atjautīgākā veidā, kas arī mazāk lauž virtuālās realitātes pieredzi. Attēlā 3.2 var



3.2 attēls: Sākuma gaitēņa izskats. Bildē var redzēt dažus objektus uz margām, ar kuriem varēs mijiedarboties

aplūkot šī gaitēņa izskatu.

Nākamā telpa ir personāža telpa. Aplūkot 3.3 un 3.4 attēlus. Šeit būs izklāstīti četri uzdevumi. Kur katrs parāda dažādu VR mijiedarbību veidu.



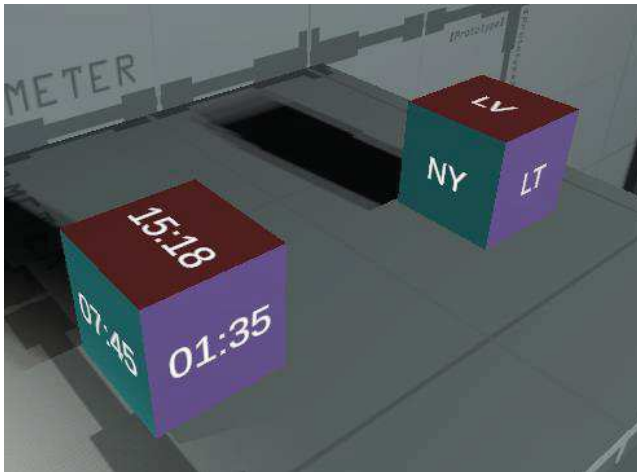
3.3 attēls: Personāža telpa. Gulta, darba galds un krēsls, pulkstenis un galda lampa



3.4 attēls: Personāža telpa. Skapis, TV, plaukts

Izklāstītie uzdevumi iekļaujas secīgajā hierarhijā, ar atvērtās hierarhijas elementiem. Kopā ir septiņi elementi, lai tiktu līdz brīvībai.

1. Viss sākās ar diviem metamajiem kauliņiem. Uz viena kauliņa ir rakstīti pulksteņa laiki un uz otra – burti. Katra kuba šķautne ir savā krāsā, tās sakrīt uz abiem kubiem un starp to ir sakarība – uz abiem kubiem ir sarkana krāsa, bet uz viena kuba sarkanās šķautnes ir rakstīts LV un uz otra kuba - 15:18. Ir jāatrod sakarība starp šiem kubiem un pulksteni uz sienas. Skatīt attēlu 3.5. Jāpiemin, ka šie kauliņi neatradīsies blakus viens otram, tie būs



3.5 attēls: Metamo kauliņu izskats. Pa kreisi ir kauliņš ar laikiem un pa labi ir kauliņš ar valstu saīsinājumiem

paslēpti pa telpu.

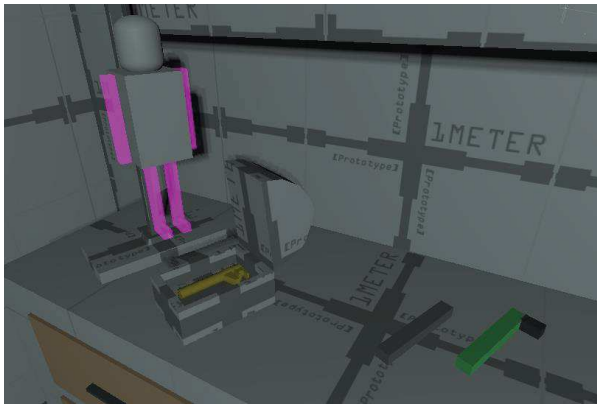
2. Uz sienas ir pulkstenis ar tekstu apakšā “New York” – saīsinājums NY. Ir jāatrod abi metamie kauliņi un pēc krāsām jāpaskatās, cik rāda laiku NY krāsai. NY atrodas uz gaiši zila lauciņa, tāpēc ir jāsameklē otrs kauliņš, jāatrod gaiši zila krāsa un tad būs atbilde. Šajā



3.6 attēls: Izskats pulkstenim pie sienas

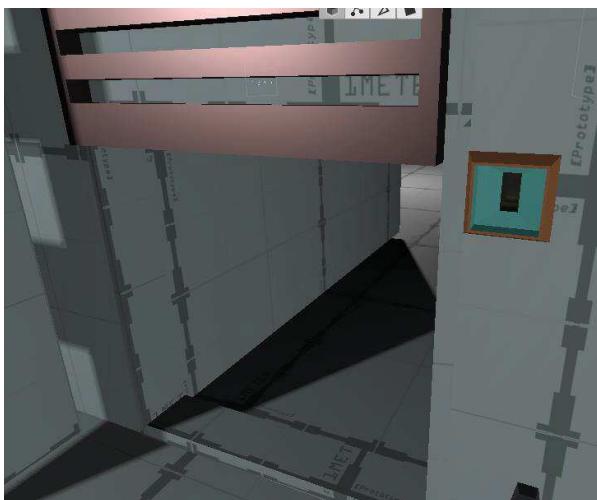
gadījumā atbilde ir 07:45. Spēlētājam ir jāaiziet pie siena pulksteņa un pašam jāuzgriež šis laiks. To izdarot, attaisās atvilktnē. Pulksteņa bilde aplūkojama attēla 3.6.

3. Uz skapīša stāvēs statīvs ar lelles rumpi un galvu, kurai priekšā ir maza kastīte. Viena roka un viena kāja būs nolikta istabā, bet otra roka un kāja ir paslēpta atvilktnē – ja neatrisina pulksteņa mīklu, tad spēlētājs netiek pie šīm lelles daļām. Pieliekot visas detaļas klāt pie statīva lelles, tā uzsāk animāciju un attaisa kastīti, kas atrodas lellei priekšā. Kastītē ir atslēga. 3.7 attēlā var aplūkot lelli, kasti ar atslēgu un lelles sastāvdaļas.



3.7 attēls: Kreisajā pusē ir redzama lelle, kur rozā vietas rāda, kur var nolikt piestiprināmos objektus – rokas un kājas. Tālāk pa labi ir redzama kaste, kura atveras, kad visi objekti ir pielikti pie lelles. Pa labi no kastes ir lelles viena roka un kāja.

4. Ar šo atslēgu ir jāatslēdz ventilācijas šahtas durvis. Attēlā 3.8 var aplūkot atslēgas caurumu un durvis.



3.8 attēls: Redzams atslēgas caurums (pa labi) un jau atvērtas durvis, pēc atslēgas ielikšanas.

5. Pēdējā uzdevumā ir jāatrisina mīkla telpā, kas ir iekšā ventilācijas šahtā. Tur būs jāievada kods, kuram blakus uz papīra ir norāde: LV hh MEX mm. Tas nozīmē, ka pirmie koda cipari būs LV laika stundas un pēdējie divi cipari būs MEX laika minūtes. LV laiks ir 15:18 un MEX laiks ir 09:05. Paņemot LV laika stundas, kas ir skaitļi 1 un 5, un paņemot MEX laika minūtes, kas ir skaitļi 0 un 5. Tad gala kods, kas spēlētājam ir jāievada, ir 1505. Attēlā 3.9 var aplūkot uzdevuma detaļas sīkāk.



3.9 attēls: Pēdējais uzdevums. Pa kreisi ir redzama koda pavediens. Pa labi ir ciparu klaviatūra un uz ekrāna jau ievadīts pareizais kods.

Spēlētājam ievadot šo kodu, atskanēs durvju atvēršanās skaņa un virs durvīm parādīsies zaļa gaisma. Pirms tam šī gaisma visu laiku bija sarkana.

Šī daļa ir tikai viena daļa no kopējās plānotās spēles. 3.1 attēlā aplūkojamie uzdevumi tiks izstrādāti Bakalaura darbā, bet kopējā spēle būs lielāka – pa visu māju. Pēc tam, kad atver durvis un spēlētājs iznāks no guļamistabas, tad pārējā māja būs mainījies uz mazliet tumšāku un drūmāku stilu, un būs pazudusi ģimene. Spēlētājam pa visu māju vajadzēs atrisināt mīklas, lai noskaidrotu, kas notika ar viņa ģimeni.

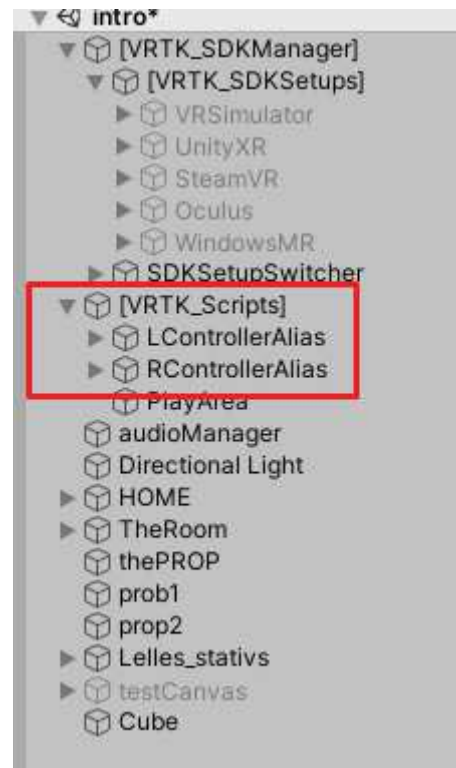
3.1 Izmantotie skripti

Kā jau autors minēja, tad pamatā tiek izmantoti *VRTK* skripti. Tie nodrošina fundamentālās vajadzības – pārvietošanās un mijiedarbība.

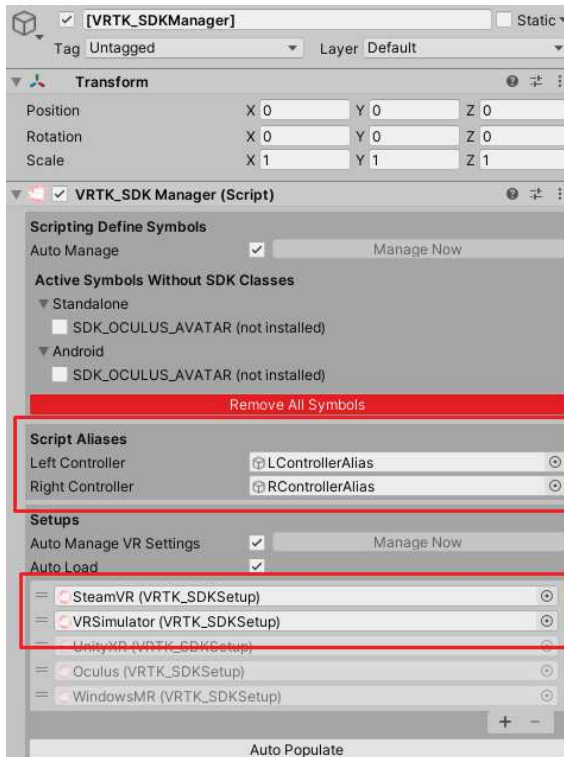
Lai nodrošinātu, ka *VRTK* darbojas, tad ainā vajag iekļaut *VRTK* objektu ar nosaukumu [*VRTK_SDKManager*]. Zem šī objekta ir dažādu VR platformu spēlētāju objekti – ar kuriem spēlētājs pārvietojas spēlē. Šie objekti ir:

- *VR Simulator* – ja nav pieslēgta neviena VR ierīce un vēlas simulēt VR uz datora, izmantojot klaviatūru un peli.
- *UnityXR*
- *SteamVR*
- *Oculus*
- *WindowsMR*

Uzspiežot uz *SDKManager* objekta, “Inspector” logā zem *SDKManager* skripta ir jāizvēlas platformas, kuras lietos. Tā kā autors strādā ar HTC Vive, tad pašā augšā ir jābūt *SteamVR* un zem tā ir *VR Simulator*. Kā arī ir jāievelk objekti, uz kuriem būs kontrolieru skripti, logos “Left controller alias” un “Right controller alias”. Uzstādījumus aplūkot 3.10 un 3.11 attēlos.



3.11 attēls: Objekti ainā, kurus izmanto *VRTK_SDKManager* skripti



3.10 attēls: VRTK_SDKManager uzstādījumi

Tālāk ir jāveic uzstādījumi objektiem “LControllerAlias” un “RControllerAlias”. Šie objekti saturēs skriptus, kuri nodrošinās kustību un mijiedarbību katram kontrolierim. Abiem kontrolieriem skripti būs vienādi.

Pirmo skriptu pievieno VRTK_Pointer. Šis skripts nodrošina, ka nospiežot izvēlētu pogu, šajā gadījumā skārienjūtīgo pogu jeb angl. *touchpad* pogu, tad parādīsies rādītājs no kontroliera, kas rāda uz kuru vietu spēlētājs varēs teleportēties. Lai to nodrošinātu ir jāpievieno skripts VRTK_BezierPointerRenderer un jāievelk VRTK_Pointer skriptā zem sadaļas “Pointer Activation Settings” logā “Pointer Renderer”. Šajā logā var ievilkt arī citus rādītāju skriptus, bet autors ir izvēlējis hiperbolas rādītāju. Tas nodrošina, ka var arī teleportēties mainot Y augstumu un pats rādītājs izskatās pievilcīgāk nekā taisns rādītājs.

Kontrolieru skriptu objektiem tālāk pievieno VRTK_ControllerEvents – atbild par nospiešajām pogām un padod informāciju citiem skriptiem, un VRTK_Touch – skatās, kad kontrolieris pieskarās kādam interaktīvam objektam. Šie divi skripti ir nepieciešami skriptam VRTK_InteractGrab – piespiežot noteiktu pogu, paceļ objektu. Nākamais skripts ir VRTK_InteractUse – nodrošina objektu izmantošanu, nospiežot izvēlēto pogu.

Kad kontroliera skripti ir pabeigti, tad var sākt veidot interaktīvus objektus. Autora spēlē viens no sarežģītiem interaktīviem objektiem ir durvis. Sarežģīts tāpēc, ka šis objekts var veikt tikai noteiktas kustības – tikai rotēt pa noteiktu asi. Durvīm ir nepieciešams skripts `VRTK_PhysicsRotator` – atbild par rotēšanas iestatījumiem, `VRTK_Interactive` – atbild par mijiedarbību. Šajā skriptā ir jāievelk skripts, kas nosaka mijiedarbības veidu abiem kontrolieriem. Zem “Grab attach mechanic script” ir jāievelk pievienotais `VRTK_RotatorTrackGrabAttach`. Tas nosaka kāds būs mijiedarbības veids. Šis skripts nosaka, ka objekts, kuru pacels – rotēs. Tad šim objektam ir jāpievieno *Rigidbody* – piešķir gravitāciju un masu. Pēdējā komponente ir *Hinge Joint* – eņģes savienojums, nosaka pa kuru asi rotēs. Šajā skriptā vajag pie X un Z ass ielikt 0, un Y asij 1. Lai durvis rotēt kā īstajā dzīvē, kad tām pieskarās un velk. Parastus objektus, kurus grib vienkārši pacelt, var uzstādīt izmantojot VRTK logu zem Window > VRTK > Setup interactive object un VRTK pats saliks visus nepieciešamos skriptus šim objektam. Šajā izvēlnē var izvēlēties pacelšanas veidu, pacelšanas iestatījumus un vai objekts ir izmantojams, izmantošanas iestatījumi un skaņas iespējas. Līdzīgi darbojas pulkstenis, tikai tam ir nomainīta ass. Kā arī līdzīgi darbojas atvilktnes, tikai tur rotācijas vietā maina objekta pozīciju pa noteiktu asi un noteiktu attālumu.

3.2 Autora veidotie skripti

Autors paskaidro kā strādā izveidotie skripti. Paši skripti atrodas pielikumā ar tādiem pašiem nosaukumiem.

1. **audioManager**

Satur apakšklasi `Sound`. Kalpo kā skaņu menedžeris – viegli ielikt jaunas skaņas un izmantot tās kodā. Skatās vai ainā ir tikai viens skaņu pārvaldnieks (angl. *audioManager*) un skaņu pārvaldnieks pārnes arī uz citām ainām, lai skaņa nebeigtos, ja tās maina. Katrai skaņai uztaisa komponenti, kuras iestatījumi tiek kontrolēti ar `sound` klases palīdzību. Koda redzams, ka sākumā pārbauda, cik šo audio pārvaldnieku ir ainā. Ja ir, tad to izdzēš. Pēc tam skriptā norādītās vērtības katrai skaņai, piešķir katras skaņas komponentei. Bildē 3.2.1 var aplūkot kā var

izveidot jaunu skaņu. Skaņas skriptā izsauc, izmantojot skaņai iedoto nosaukumu audioManager skriptā.

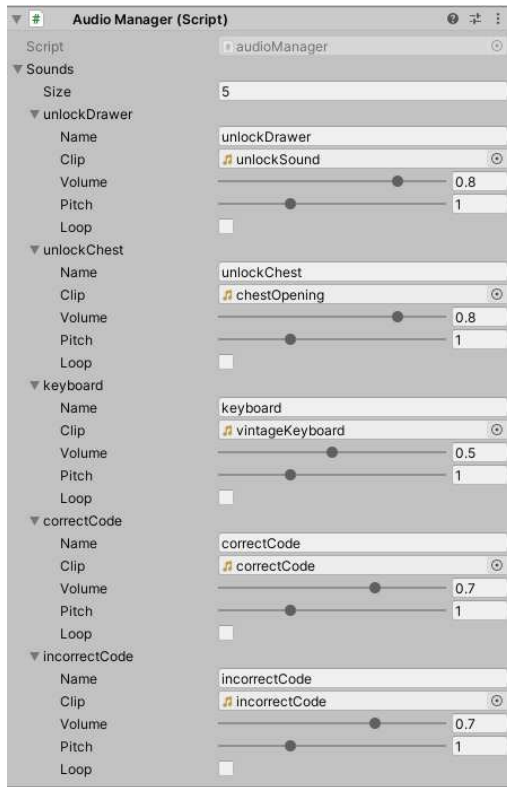
FindObjectOfType<audioManager>().Play("unlockChest") – šādi izsauc skaņu skriptā.

```
private void Awake()
{
    if(instance == null)//check if there already is instance of audiomanager in
scene
    {
        instance = this;//if not, this is the main instance
    }
    else//if there already is, then delete new instance
    {
        Destroy(gameObject);
        return;
    }

    DontDestroyOnLoad(gameObject);//save object between scenes

    foreach (Sound s in sounds)
    {
        s.source = gameObject.AddComponent<AudioSource>();
        s.source.clip = s.clip;

        s.source.volume = s.volume;
        s.source.pitch = s.pitch;
        s.source.loop = s.loop;
    }
}
```



3.2.1 attēls: Skaņu pārvaldnieka skripta parametru logs.

2. checkClock

Pārbauda, kad spēlētājs ir pulkstenī uzgriezis pareizo laiku. Skatās pulksteņa stundu un minūšu rādītāja Y rotāciju. Ja abi rādītāji ir iekļaujas ± 5 grādos no pareizā leņķa, tad skripts ieslēdz mijiedarbošanos ar atvilktni – to tagad var atvērt. Kodā ir parādīta bool metode, kas skatās, vai pulksteņa rādītājiem ir pareiza rotācija.

```
bool checkForCorrectTime()
{
    float minHour = correctHour - plusMinusCorrect;
    float maxHour = correctHour + plusMinusCorrect;

    float minMinute = correctMinute - plusMinusCorrect;
    float maxMinute = correctMinute + plusMinusCorrect;

    if ((hourTick.transform.localEulerAngles.y < maxHour &&
        hourTick.transform.localEulerAngles.y > minHour) &&
        (minuteTick.transform.localEulerAngles.y < maxMinute &&
        minuteTick.transform.localEulerAngles.y > minMinute))
    {
```

```

        print("CORRECT");
        FindObjectOfType<audioManager>().Play("unlockDrawer");
        drawer.isGrabbable = true;
        return true;
    }
    else
    {
        return false;
    }
}

```

3. clockCheck

Mazs skripts, kas ainā uz teksta objektiem rādīja pulksteņa rādītāju rotāciju. Tika izmantots testēšanā.

```

void Update()
{
    hourText.text = "Hours: " + hour.transform.localEulerAngles.y;
    minuteText.text = "Minutes: " + minute.transform.localEulerAngles.y;
}

```

4. crateCheck

Šajā skriptā ir divas svarīgas metodes – `OnSnap()` un `onDesnap()`. Šīs metodes izmanto `VRTK_SnapDrop` skripti. Metode `OnSnap()` izpildās, kad objekts ir pielicies klāt savā norādītajā vietā, `onDesnap()` izpildās, kad objekts tiek izņemts no savas norādītās vietas. Šis skripts sākās ar mainīgo `snaps = 0`. Katra `OnSnap()` metode pieskaita vienu vērtību klāt `snaps` un katra `OnDesnap()` noņem vienu vērtību no `snaps`. Kad `snaps` sasniedz vērtību 4, tad tas nozīmē, ka lellei visi 4 objekti ir pielikti klāt un spēlētājs ir izpildījis mīklu. Tikai tad tiek atskaņota kastes animācija, kas atver kasti, kur iekšā ir atslēga nākamajam uzdevumam. Koda fragmentā var redzēt, ka skripts visu laiku skatās *snaps* vērtību. Ja tā ir 4, un nav jau bijusi atvērta, tad atskaņojam skaņu un sākam animāciju.

```

void Update()
{
    if(snaps == 4)
    {
        //play anim
        if(anim != null && !hasOpened)
        {
            hasOpened = true;
            Debug.Log("Playing animation...");
            FindObjectOfType<audioManager>().Play("unlockChest");
        }
    }
}

```

```

        anim.SetTrigger("openCrate");
    }
}

```

5. deskLampSwitcher

Kontrolē lampas slēdzi uz katru reizi, kad tiek nospiesta lampas poga. Pie nospiešanas izpildās funkcija, kas nomaina bool vērtību uz pretējo un šo vērtību padod uz funkciju, kas ar šo bool vērtību vai nu ieslēdz, vai izslēdz gaismu.

```

void checkIfPressed(object sender, ControllableEventArgs e)
{
    lampLightOn = !lampLightOn;
    lampLightController(!lampLightOn);
}

void lampLightController(bool state)
{
    lampLight.enabled = state;
}

```

6. keyTrigger

Skatās, kad triggerī ieiet objekts ar birkas (angl. *tag*) nosaukumu “key”. Kad šis objekts ienāk triggerī, tad tiek atskaņota animācija, lai atvērtu ventilācijas šahtas durvis.

```

private void OnTriggerEnter(Collider col)
{
    if(col.gameObject.tag == "key")
    {
        anim.SetTrigger("openDoors");
        RoomLight.gameObject.SetActive(true);
    }
}

```

7. overrideScript

Autors nevarēja izdomāt veidu kā salabot problēmu ar pulksteņa rādītājiem, jo VRTK skripti izmainīja vērtības, kad spēle tiek palaista. Tāpēc šis skripts tos nomaina uz vērtībām, kas iespējo ar rādītājiem mijiedarboties. Ieslēdz, ka objekts ir kinemātisks – nekustās, un eņģes komponentē izslēdz, ka izmanto atsperi (angl. *useSpring*).

```

void check()
{

```

```

    if (!hasChecked)
    {
        print("");
        if (rb != null) rb.isKinematic = true;
        if (hinge != null) hinge.useSpring = false;
        hasChecked = true;
    }
}

```

8. pointerRaycast

Šis bija testa skripts, lai redzētu kā varētu mijiedarboties ar UI. Šis skripts visu laiku “šauj” staru konstantā rotācijā un skatās, kur šis stars atduras. Ja atduras pret objektu ar birku “vrtkCanvas”, tad konsolē izvada ziņojumu.

```

Vector3 raycastFwd =
raycastOrigin.transform.TransformDirection(Vector3.forward) * 10;
Debug.DrawRay(raycastOrigin.transform.position, raycastFwd * 3f,
Color.green);

```

```

    if (Physics.Raycast(raycastOrigin.transform.position,
raycastOrigin.transform.forward, out hit, 50f))
    {
        if (hit.collider.CompareTag("vrtkCanvas"))
        {
            //Debug.Log("Object hit: " + hit.collider.name);
            Debug.Log("vrtk canvas!");
        }
    }
}

```

9. rendererDisable

Izslēdz oriģinālo kontrolieru 3D objektu, lai autors varētu izmantot citu kontrolieru modeli – 3D rokas. Tā kā kontrolieri sastāv no vairākiem modeļiem, tad skripts atrod visus kontroliera apakšobjektus un tos izslēdz.

```

List<GameObject> list = new List<GameObject>();
for (int i = 0; i < transform.childCount; i++)
{
    list.Add(transform.GetChild(i).gameObject);
}
foreach (var child in list)
{
    MeshRenderer mesh = child.GetComponent<MeshRenderer>();
    if (mesh != null) mesh.enabled = false;
}

```

10. sound

sound ir apakšklase klasei *audioManager*. *AudioManager* šo skriptu izmanto, lai skaņu vērtības varētu rediģēt *audioManager* skriptā. *AudioManager* klase izmanto šīs vērtības un šī klase nomaina vērtības oriģinālajam skaņas objektam. Šāda veida skaņu menedžeris atvieglo un paātrina izstrādes procesu.

```
public class Sound
{
    public string name;
    public AudioClip clip;

    [Range(0f, 1f)]
    public float volume;
    [Range(0.1f, 3f)]
    public float pitch;
    public bool loop;

    [HideInInspector]
    public AudioSource source;
}
```

11. inputManager

Apstrādā koda ievadi un kā to parāda spēlē. Skatās kurā teksta objektā likt ievadīto ciparu un skatās vai kods ir ievadīti pareizi vai nepareizi. Šo metodi izsauc skriptā *keyboardKey*, kad tiek nospiests taustiņš.

```
public void getKey(string _char)
{
    //print("received key: " + _char);
    if(_char == "C")
    {
        codeField1.text = "";
        codeField2.text = "";
        codeField3.text = "";
        codeField4.text = "";
        inputCode = "";
        return;
    }
    if (codeField1.text == "")
    {
        codeField1.text = _char;
        inputCode += _char;
    }
    else if( codeField2.text == "")
```

```

    {
        codeField2.text = _char;
        inputCode += _char;
    }
    else if (codeField3.text == "")
    {
        codeField3.text = _char;
        inputCode += _char;
    }
    else if (codeField4.text == "")
    {
        codeField4.text = _char;
        inputCode += _char;
        checkCode();
    }
}

```

12. keyboardKey

Kad spēlētājs pieskaras klaviatūras pogai, tiek izpildīta metode, kas nosūta noteiktu mainīgo (ciparu vai burtu C) skriptam *inputManager*, kas šo mainīgo apstrādā.

```

void ifPressed(object sender, ControllableEventArgs e)
{
    if(keyToInput != null)
    {
        _inputManager.getKey(keyToInput);
        //print("sent key:" + keyToInput);
        FindObjectOfType<audioManager>().Play("keyboard");
    }
}

```

3.3 Ekonomiskais ieguvums

Autora cerētais ieguvums nav zināms, jo projekts vēl nav pabeigts un nav dots testēt citiem, lai redzētu kādas ir atsauksmes par produktu. Taču, autors plāno sadarboties ar VR studiju, kur, iespējams, par katru spēlēto reizi maksātu autoram noteiktu summu. Kā arī spēle tiktu izlikta pārdošanā platformā *Steam*. Cena spēlei platformā *Steam* būtu atkarīga no gala produkta kvalitātes un spēles apjoma.

3.4 Ekonomiskais novērtējums

Šajā apakšnodaļā tiks veikts apkopojums, cik daudz resursu ir nepieciešams, lai izveidotu šādu projektu. Tiks apskatīti programmatūras resursi, tehniskie resursi un izstrādes resursi.

Programmatūras resursos iekļaujas visu izstrādē izmantoto programmatūru izmaksas. Tā kā tika izmantots *Unity 2019.3*, ko autors izmanto par velti, jo ar to nepelna naudu, un *Gimp*, kas arī ir par velti pieejama programmatūra. Kopā sanāk, ka programmatūras izmaksas ir 0 EUR.

Tehniskie resursi iekļauj izmantotās tehniskās ierīces. Sākot no peles paliktņa beidzot ar pašu VR ierīci. Bet tā kā autors pats jau bija iegādājies datoru ar visām perifērijas ierīcēm un VR ierīci dabūja no Vidzemes Augstskolas VR Laboratorijas, tad šis uzskaitījums noderēs tikai tiem, kam šo tehniku vajadzēs pirkt – kam tās nav. Tehniskie izdevumi autoram kopā sasniedz 1845 EUR. Šo cenu ir iespējams samazināt, ja izmanto parastākas perifērās ierīces, mazāk jaudīgu datoru. Visas cenas ņemtas no veikala dateks.lv.

Resursi	Izmaksas (EUR)
Stacionārais dators	700
Ekrāns Benq 24"	120
Razer Blackwidow Chroma	130
Asus ROG Gladius II	80
Corepad peles paliknis	15
Steelseries Arctis 3 austiņas	90
HTC Vive VR brilles, ar divām bāzes stacijām un diviem kontrolieriem	710
KOPĀ:	1 845

3.4.1 tabula: Tehnisko resursu izdevumu uzskaitījums

Cilvēkresursos iekļauj izstrādes procesa izmaksas. Galveno darbību uzskaitīts kā *Unity* izstrādi, jo zem tā iekļaujas testēšana, kas tiek veikta paralēli izstrādei, tāpēc to atsevišķi autors neiedalīs. Kā arī vides izstrāde netiks atsevišķi izdalīta. Autors izvēlējās stundas likmi ņemot vērā statistiku no *algas.lv* vidējās algas C# programmētājam. Ņemot vērā, ka autors nav profesionālis, tad stundas likme varētu būt mazliet zemāka. Kopējās cilvēkresursu izmaksas ir 837 EUR. Tomēr ņemot vērā, ka autoram liekas, ka viņa pieredze neatbilst vidējam līmenim šajā jomā tad atbilstošā stundas likme varētu būt aptuveni 7 eiro stundā, kas kopēji par projekta izstrādi būtu 630 EUR.

Darba veids	Stundas likme (EUR)	Nostrādātās stundas
<i>Unity</i> izstrāde	9.3 EUR	90
KOPĀ:		837

3.4.2 tabula: Cilvēkresursu izdevumu uzskaitījums

Kopējās projekta izmaksas ir **2475 EUR**. Bet tā kā autors nepirka šo visu tehniku speciāli projektam un neviens nemaksāja autoram, tad reālās izmaksas ir tehnikas amortizācija un maksa par elektrību un internetu.

Secinājumi

- Autora uzstādītie darba uzdevumi tika izpildīti un mērķis sasniegts. Tika izdomāti 3 galvenie uzdevumi un realizēti VR vidē. Protams, gan uzdevumi, gan 3D vide ir prototipa stāvoklī un priekšā ir daudz uzlabojumu un papildinājumu, lai spēle būtu kvalitatīva.
- Aplūkojot jau izveidotas izlaušanās VR spēles, autors secina, ka pa velti ir tās, kuras pēc apjoma ir mazas un tās var izspēlēt pusstundas laikā. Taču tās, kuras jāspēlē stundu vai pat vairākās stundas, un ir pieejamas vairākas telpas, tās ir pa maksu.
- Tā kā autoram nav bijusi iepriekšēja pieredze ar izlaušanās telpām, tad autors, domājot uzdevumus, saprata, ka ir ļoti sarežģīti izdomāt interesantus un saistošus uzdevumus. Saistošus gan vienam uzdevumam ar otru, gan ar pašu spēli – vidi un stāstu.
- Autors šo projektu izstrādāja pakāpeniski. Sākumā izveidoja 3D vidi, bet bez liekiem objektiem, tikai telpas. Tālāk izveidoja uzdevumus, bet ievērojot to secību spēlē. Autors uzskata, ka tas atviegloja izstrādes procesu, jo ir vieglāk izsekot līdzī spēles attīstības gaitai. Jo tiek strādāts pie viena uzdevuma un tā pavedienu izveides, kad tas tiek pabeigts, tad sāk izstrādāt nākamo uzdevumu.
- Šī projekta izstrāde noteikti ir uzlabojusi autora prasmes un zināšanas par *VRTK* spraudņa pielietojumu *Unity* vidē. Sākumā bija ļoti grūti saprast kā šis spraudnis strādā, jo tas izmanto ļoti daudz komponentes. Autors tagad māk viegli veikt nepieciešamos uzstādījumus, lai nodrošinātu VR pamata vajadzības.
- Lai projekts būtu kvalitatīvs un nebūtu jātērē pārāk daudz laika, tad visticamāk vajadzētu sadarboties ar kādu 3D dizaineri, kurš izstrādātu kvalitatīvus 3D modeļus spēlei. Jo autoram nav lielas pieredzes, taču nav arī svešs ar 3D modeļu izstrādi.
- Projektu testēja vairāki cilvēki un autors secina, ka cilvēkiem bez iepriekšējas VR pieredzes, ir grūti saprast VR kontroles un principu. Tāpēc vajadzētu piedomāt pie ļoti labas pamācības spēles sākumā, kurā nav daudz tekstu un ir viegli saprotama.
- Turpinot projekta izstrādi, būtu vēlams izmantot *GitHub*, lai varētu līdzī sekot versijām un būtu ērtāka projekta izstrāde.

- Nākamais autora uzdevums būtu izdomāt tālāko stāstu spēlei, kad šī telpa ir izieta un spēlētājs var pārvietoties pa māju/ citām telpām. Tad realizēt izdomāto stāstu un uzdevumus, uzlabot 3D vidi un mijiedarbības, iekļaut vairāk skaņu un efektus, lai spēle būtu pievilcīgāka acij un vairāk imersīva.

IZMANTOTĀ LITERATŪRA

- 5 Uses for Virtual Reality*. (2019. gada 27. Maijs). Ielādēts no fdmgroup:
<https://www.fdmgroup.com/5-uses-for-virtual-reality/>
- Admin. (2019. gada Maijs). *HOME › VIRTUAL REALITY › VIRTUAL REALITY TRAINING EQUIPMENT “ACADICUS” LAUNCHED BY ARCH VIRTUAL*. Ielādēts no immersive-technology.com: <https://immersive-technology.com/virtualreality/virtual-reality-training-equipment-launched-by-arch-virtual/>
- Best VR headsets 2020*. (2020. gada 31. 1). Ielādēts no techradar:
<https://www.techradar.com/news/the-best-vr-headset>
- Casual Gaming*. (2018. gada 13. novembris). Ielādēts no computerhope:
<https://www.computerhope.com/jargon/c/casual-gaming.htm>
- Fundamentals of Game Design*. (2006). Prentice Hall.
- Game Design*. (2003). New Riders.
- Insinna, V. (2019. gada 6.. Decembris). *Wanted: Virtual reality headsets that aren't made in China*. Ielādēts no defensenews.com: <https://www.defensenews.com/digital-show-dailies/itsec/2019/12/06/wanted-virtual-reality-headsets-that-arent-made-in-china/>
- news.game.co.uk. (2020. gada 1.. Maijs). *6 DETAILS YOU MAY HAVE MISSED IN THE ASSASSIN'S CREED VALHALLA TRAILER*. Ielādēts no news.game.co.uk:
<https://news.game.co.uk/assassins-creed-valhalla-trailer-breakdown/>
- Next Generation. (1996). Gates on Games. *Next Generation*, 158.
- Oculus Quest*. (2020). Ielādēts no oculus: <https://www.oculus.com/quest/>
- Oculus Quest review*. (2019. gada 23. 5). Ielādēts no techradar:
<https://www.techradar.com/reviews/oculus-quest-review>
- Omnivirt. (2018. gada 11.. Februāris). *Examples of Virtual Reality in Sports*. Ielādēts no Omnivirt.com: <https://www.omnivirt.com/blog/virtual-reality-sports-examples/>
- Özistek, B. (2019. gada 22.. Augusts). *New Rising Trend: Hyper-Casual Games*. Ielādēts no medium.com: <https://medium.com/@baris.ozistek/new-rising-trend-hyper-casual-games-da6f6da1efb9>

Peeking Behind the Locked Door: A survey of escape room facilities. (2015). Kanāda:
<http://scottnicholson.com/pubs/erfacwhite.pdf>.

Pino, N. (2020. gada 5.. Maijs). *Oculus Quest 2 now delayed until 2021*. Ielādēts no
www.techradar.com: <https://www.techradar.com/news/oculus-quest-2-now-delayed-until-2021-according-to-new-report>

PlayStation VR review. (2019. gada 15. 1). Ielādēts no techradar:
<https://www.techradar.com/reviews/gaming/playstation-vr-1235379/review>

Specification. (2020). Ielādēts no samsung: <https://www.samsung.com/global/galaxy/gear-vr/specs/>

Steam. (2018. gada 20.. Novembris). *Farming Simulator 19*. Ielādēts no Steam.com:
https://store.steampowered.com/app/787860/Farming_Simulator_19/

Tech specs. (2020). Ielādēts no valvesoftware:
<https://www.valvesoftware.com/en/index/headset>

Technical Specification. (2020). Ielādēts no playstation: <https://www.playstation.com/en-gb/explore/playstation-vr/tech-specs/#vr2>

Tolin, M. (2020. gada 19. Februāris). *Top 21 New FPS Games for PC*. Ielādēts no
gamersdecide.com: <https://www.gamersdecide.com/articles/new-fps-games-upcoming-fps-games>

Virtual Reality (VR) - Statistics & Facts. (2020. gada 3. Marts). Ielādēts no statista:
<https://www.statista.com/topics/2532/virtual-reality-vr/>

virtualrealityandphobias. (2016. gada 9.. Maijs). *Virtual Reality Should Be Considered As a Helpful Tool for Mental Health.* Ielādēts no
<https://virtualrealityandphobias.wordpress.com/>
<https://virtualrealityandphobias.wordpress.com/>

Walsh, K. (2017. gada 14.. Jūnijs). *Real Uses of Virtual Reality in Education: How Schools are Using VR.* Ielādēts no www.emergingedtech.com:
<https://www.emergingedtech.com/2017/06/real-uses-of-virtual-reality-in-education-how-schools-are-using-vr/>

What is virtual reality? (2019. gada 26. 03). Ielādēts no marxentlabs:
<https://www.marxentlabs.com/what-is-virtual-reality/>

Winslow, J. (2020. gada 10. Marts). *Check Out These New Call Of Duty: Warzone Screenshots*. Ielādēts no gamespot.com:
<https://www.gamespot.com/articles/check-out-these-new-call-of-duty-warzone-screensho/1100-6474556/>

I PIELIKUMS. Spēles skripti.

1. audioManager.cs

```
using UnityEngine.Audio;
using System;
using UnityEngine;

public class audioManager : MonoBehaviour
{
    public Sound[] sounds;

    public static audioManager instance;
    private void Awake()
    {
        if(instance == null)//check if there already is instance of audiomanager in scene
        {
            instance = this;//if not, this is the main instance
        }
        else//if there already is, then delete new instance
        {
            Destroy(gameObject);
            return;
        }

        DontDestroyOnLoad(gameObject);//save object between scenes

        foreach (Sound s in sounds)
        {
            s.source = gameObject.AddComponent<AudioSource>();
            s.source.clip = s.clip;

            s.source.volume = s.volume;
            s.source.pitch = s.pitch;
            s.source.loop = s.loop;
        }
    }

    public void Play(string name)
    {
        //find sound in sounds array, where
        Sound s = Array.Find(sounds, sound => sound.name == name);
        if (s == null)
        {

```



```

        Debug.LogWarning("Sound: " + name + " not found!");
        return;
    }
    s.source.Play();
}
}

```

2. **checkClock.cs**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using VRTK;

public class checkClock : MonoBehaviour
{
    public GameObject hourTick;
    public GameObject minuteTick;
    public VRTK_InteractableObject drawer;
    public float correctHour;
    public float correctMinute;
    public float plusMinusCorrect;

    private bool correctTimeSet;

    // Start is called before the first frame update
    void Start()
    {
        correctTimeSet = false;
        drawer.isGrabbable = false;
    }

    // Update is called once per frame
    void Update()
    {
        if(!correctTimeSet) correctTimeSet = checkForCorrectTime();
    }

    bool checkForCorrectTime()
    {
        float minHour = correctHour - plusMinusCorrect;
        float maxHour = correctHour + plusMinusCorrect;

        float minMinute = correctMinute - plusMinusCorrect;

```

```

float maxMinute = correctMinute + plusMinusCorrect;

if ((hourTick.transform.localEulerAngles.y < maxHour &&
hourTick.transform.localEulerAngles.y > minHour) &&
(minuteTick.transform.localEulerAngles.y < maxMinute &&
minuteTick.transform.localEulerAngles.y > minMinute))
{
    print("CORRECT");
    FindObjectOfType<audioManager>().Play("unlockDrawer");
    drawer.isGrabbable = true;
    return true;
}
else
{
    return false;
}
}
}

```

3. **clockCheck.cs**

```

using System.Collections;
using System.Collections.Generic;
using TMPro;
using UnityEngine;

public class clockCheck : MonoBehaviour
{
    public TextMeshPro hourText;
    public TextMeshPro minuteText;

    public GameObject hour, minute;

    // Update is called once per frame
    void Update()
    {
        hourText.text = "Hours: " + hour.transform.localEulerAngles.y;
        minuteText.text = "Minutes: " + minute.transform.localEulerAngles.y;
    }
}

```

4. **crateCheck.cs**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

```

```

using UnityEngine.Experimental.XR.Interaction;

public class crateCheck : MonoBehaviour
{
    public int snaps;
    private bool hasOpened = false;
    public Animator anim;
    // Start is called before the first frame update
    void Start()
    {
        snaps = 0;
        hasOpened = false;
    }

    // Update is called once per frame
    void Update()
    {
        if(snaps == 4)
        {
            //play anim
            if(anim != null && !hasOpened)
            {
                hasOpened = true;
                Debug.Log("Playing animation...");
                FindObjectOfType<audioManager>().Play("unlockChest");
                anim.SetTrigger("openCrate");
            }
        }
    }

    public void onSnap()
    {
        snaps += 1;
    }

    public void onDesnap()
    {
        snaps -= 1;
    }
}

```

5. deskLampSwitch.cs

```

using System.Collections;
using System.Collections.Generic;

```

```

using UnityEngine;
using UnityEngine.Experimental.GlobalIllumination;
using VRTK.Controllables.PhysicsBased;
using VRTK;
using VRTK.Controllables.ArtificialBased;
using VRTK.Controllables;

public class deskLampSwitch : MonoBehaviour
{
    public Light lampLight;
    public bool lampLightOn = true;

    public VRTK_BaseControllable pusher;

    private void Start()
    {
        lampLight.enabled = lampLightOn;
        pusher.MaxLimitReached += checkIfPressed;
    }

    private void Update()
    {
    }

    void checkIfPressed(object sender, ControllableEventArgs e)
    {
        lampLightOn = !lampLightOn;
        lampLightController(!lampLightOn);
    }

    void lampLightController(bool state)
    {
        lampLight.enabled = state;
    }
}

```

6. **inputManager.cs**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using TMPro;
using VRTK.Controllables;

```

```

public class inputManager : MonoBehaviour
{

    public TextMeshProUGUI codeField1;
    public TextMeshProUGUI codeField2;
    public TextMeshProUGUI codeField3;
    public TextMeshProUGUI codeField4;

    private string inputCode;

    void Start()
    {
        codeField1.text = "";
        codeField2.text = "";
        codeField3.text = "";
        codeField4.text = "";
    }

    void Update()
    {

    }

    public void getKey(string _char)
    {
        //print("received key: " + _char);

        if(_char == "C")
        {
            codeField1.text = "";
            codeField2.text = "";
            codeField3.text = "";
            codeField4.text = "";
            inputCode = "";
            return;
        }

        if (codeField1.text == "")
        {
            codeField1.text = _char;
            inputCode += _char;
        }
        else if( codeField2.text == "")

```

```

    {
        codeField2.text = _char;
        inputCode += _char;
    }
    else if (codeField3.text == "")
    {
        codeField3.text = _char;
        inputCode += _char;
    }
    else if (codeField4.text == "")
    {
        codeField4.text = _char;
        inputCode += _char;
        checkCode();
    }
}

void checkCode()
{

    print("CODE inputed:" + inputCode);

    if(inputCode == "1505")
    {
        print("CODE CORRECT");
        FindObjectOfType<audioManager>().Play("correctCode");
    }
    else
    {
        print("CODE INCORRECT");
        FindObjectOfType<audioManager>().Play("incorrectCode");
    }
}
}

```

7. **keyboardKey.cs**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using VRTK.Controllables;
using VRTK.Controllables.ArtificialBased;

public class keyboardKey : MonoBehaviour
{
    public string keyToInput;
}

```

```

private VRTK_BaseControllable pusher;
private inputManager _inputManager;

// Start is called before the first frame update
void Start()
{
    pusher = gameObject.GetComponent<VRTK_ArtificialPusher>();
    pusher.MaxLimitReached += ifPressed;
    _inputManager = FindObjectOfType<inputManager>();
}

// Update is called once per frame
void Update()
{
}

void ifPressed(object sender, ControllableEventArgs e)
{
    if(keyToInput != null)
    {
        _inputManager.getKey(keyToInput);
        //print("sent key:" + keyToInput);
        FindObjectOfType<audioManager>().Play("keyboard");
    }
}
}

```

8. keyTrigger.cs

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class keyTrigger : MonoBehaviour
{
    public Animator anim;
    public Light RoomLight;
    // Start is called before the first frame update
    void Start()
    {
        RoomLight.gameObject.SetActive(false);
    }

    // Update is called once per frame
    void Update()

```

```

    {
    }

    private void OnTriggerEnter(Collider col)
    {
        if(col.gameObject.tag == "key")
        {
            anim.SetTrigger("openDoors");
            RoomLight.gameObject.SetActive(true);
        }
    }
}

```

9. overrideScript.cs

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class overrideScript : MonoBehaviour
{
    private bool hasChecked = false;

    public Rigidbody rb;
    public HingeJoint hinge;
    void Start()
    {
        rb = GetComponent<Rigidbody>();
        hinge = GetComponent<HingeJoint>();
    }

    private void FixedUpdate()
    {
        check();
    }

    void check()
    {
        if (!hasChecked)
        {
            print("");
            if (rb != null) rb.isKinematic = true;
            if (hinge != null) hinge.useSpring = false;
            hasChecked = true;
        }
    }
}

```



```

    }
}

```

10. pointerRaycast.cs

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.EventSystems;
using UnityEngine.UI;
using VRTK;

public class pointerRaycast : MonoBehaviour
{
    public GameObject raycastOrigin;
    RaycastHit hit;

    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        Vector3 raycastFwd =
raycastOrigin.transform.TransformDirection(Vector3.forward) * 10;
        Debug.DrawRay(raycastOrigin.transform.position, raycastFwd * 3f,
Color.green);

        if (Physics.Raycast(raycastOrigin.transform.position,
raycastOrigin.transform.forward, out hit, 50f))
        {
            if (hit.collider.CompareTag("vrtkCanvas"))
            {
                //Debug.Log("Object hit: " + hit.collider.name);
                Debug.Log("vrtk canvas!");
            }
        }
    }
}

```

11. rendererDisable.cs

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class rendererDisable : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {

        List<GameObject> list = new List<GameObject>();
        for (int i = 0; i < transform.childCount; i++)
        {
            list.Add(transform.GetChild(i).gameObject);
        }

        foreach (var child in list)
        {

            MeshRenderer mesh = child.GetComponent<MeshRenderer>();
            if (mesh != null) mesh.enabled = false;

        }
    }
}

```

12. sounds.cs

```

using UnityEngine.Audio;
using UnityEngine;

[System.Serializable]
public class Sound
{
    public string name;
    public AudioClip clip;

    [Range(0f, 1f)]

```

```
public float volume;  
[Range(.1f, 3f)]  
public float pitch;  
public bool loop;  
  
[HideInInspector]  
public AudioSource source;  
}
```

II Pielikums. Apliecinājums par autora mantisko tiesību nodošanu

APSTIPRINĀTS

ar Vidzemes Augstskolas rektora
2017.gada 17.maija rīkojumu Nr.7-r

APLIECINĀJUMS

par autora mantisko tiesību nodošanu

Bakalaura darbs (turpmāk – Darbs)

IZLAUŠANĀS TELPAS SPĒLES PROTOTIPA IZSTRĀDE VIRTUĀLAJĀ REALITĀTĒ

izstrādāts Vidzemes Augstskolas Inženierzinātņu fakultātē

Pamatojoties uz Autortiesību likuma 15.pantā noteiktajām mantiskajām tiesībām, kuras darba autors var nodot trešajām personām, piekrītu, ka mans Darbs tiek padarīts sabiedrībai pieejams bez maksas pilnā apjomā:

Nepiekrītu, ka manu Darbu padara sabiedrībai ☐ pieejamu

Lūdzam norādīt pamatotu iemeslu:

Darba autors: _____, Eduards Jesikens, 18.05.2020.

Paraksts

vārds, uzvārds

datums

