

Slovenská technická univerzita v Bratislave Fakulta informatiky a informačných
technológií

Projektová dokumentácia

Tím 13 - Chaos

Vedúci tímu: Ing. Juraj Vincúr
Členovia tímu: Bc. Ivan Andrejkovič
Bc. Tadeáš Broniš
Bc. Gabriela Hózová
Bc. Nikolas Janec
Bc. Jaroslav Lišiak
Bc. Tomáš Ofčarovič
Bc. Michal Škuta
Názov témy: Pohlcujúci web [iWeb]
Akademický rok: 2017/2018

Dátum: 13.11.2017

Obsah

Dokumentácia k riadeniu	4
1 Úvod.....	5
2 Členovia tímu a ich roly.....	6
2.1 Predstavenie členov tímu.....	6
2.2 Rozdelenie rolí v tíme ZS.....	8
2.3 Rozdelenie rolí v tíme LS.....	9
2.4 Podiel práce na jednotlivých častiach dokumentácie.....	10
3 Aplikácie manažmentov.....	11
3.1 Manažment dokumentácie.....	11
3.2 Manažment komunikácie.....	11
3.3 Manažment úloh.....	11
3.4 Manažment testovania.....	11
3.5 Manažment vývoja.....	12
3.6 Manažment kvality.....	12
3.7 Manažment plánovania.....	12
4 Sumarizácie šprintov.....	14
4.1 Šprint 1 - Scooby-Doo.....	14
4.2 Šprint 2 – Lajka.....	14
4.3 Šprint 3 – Bobi.....	14
4.4 Šprint 4 – Lassie.....	15
4.5 Šprint 5 – Odie.....	15
4.6 Šprint 6 – Rex – Tedy.....	16
4.7 Šprint 7 – Beethoven – Tedy.....	16
4.8 Šprint 8 – Hačikó.....	16
4.9 Šprint 9 – Pluto.....	16
4.10 Šprint 10 – Lady a Tramp.....	17
5 Používané metodiky.....	18
5.1 Metodika dokumentácie.....	18
5.2 Metodika písania kódu.....	18
5.3 Metodika komunikácie.....	19
5.4 Metodika revízií.....	19
5.5 Metodika verziovania.....	19
5.6 Metodika testovania.....	19
5.7 Metodika manažmentu úloh.....	20
6 Globálna retrospektíva.....	21
Dokumentácia k inžinierskemu dielu.....	23
7 Úvod.....	24

8	Ciele na zimný semester	24
9	Ciele na letný semester	25
10	Analýza	26
10.1	Fotogrametria	26
10.2	Virtuálna realita	28
10.3	Technológie (nástroje)	29
10.3.1	Tango Point Cloud	29
10.3.2	3DSOM	29
10.3.3	Scann3D	30
10.3.4	3DF Zaphyr	30
10.3.5	Autodesk ReCap	30
10.3.6	Walkabout Worlds	30
10.3.7	Blender	31
10.3.8	RTAB-Map	31
10.3.9	ZED Mini	31
10.3.10	Samsung Gear 360	32
10.4	Multiplatformovosť	34
10.4.1	WebGL	34
10.4.2	WebVR	35
11	Návrh	36
11.1	Diagram tried	36
11.1.1	Opis tried	36
11.2	Prípady použitia	38
12	Implementácia	46
12.1	Prototypy	46
12.1.1	3D Room verzia 1	46
12.1.2	3D Room verzia 2	47
12.1.3	3D Room verzia 3	47
12.1.4	Panorama verzia1	50
12.1.5	Panorama verzia2	50
13	Testovanie	52
13.1	Testovanie prototypu zo zimného semestra	52
13.2	Testovanie finálnej aplikácie	56
Príloha A – Metodiky		61
Príloha B – Inštalačná príručka		61
Príloha C – Používateľská príručka		61

Dokumentácia k riadeniu

1 Úvod

Dokumentácia vznikla v rámci predmetu Tímový projekt a opisuje prácu na projekte Pohlcujúci web. Obsahom tejto dokumentácie je predstavenie členov tímu, podrobný opis manažmentu, zhrnutie jednotlivých šprintov a súhrn metodík. Špecifikácia vytváraného systému je opísaná v dokumentácii inžinierskeho diela.

Opis kapitol:

Kapitola 2 obsahuje predstavenie jednotlivých členov tímu, sú v nej definované zodpovednosti každého člena tímu a tiež podiel práce na tomto dokumente. Kapitola 3 bližšie špecifikuje procesy manažmentu projektu. Kapitola 4 sumarizuje priebeh a výsledky všetkých piatich šprintov vykonaných v priebehu zimného semestra. Poskytuje tiež zhodnotenie príslušného šprintu. Kapitola 5 je venovaná súhrnu metodík definovaných a využívaných v tíme. Táto kapitola sa odkazuje na prílohu A v ktorej sa nachádza podrobný opis použitých metodík. Kapitola 6 obsahuje globálnu retrospektívu práce v zimnom semestri. V retrospektíve je stručne zhrnuté v čom chceme pokračovať na projekte a s čím chceme skončiť.

2 Členovia tímu a ich roly

Táto kapitola sa zaoberá členmi tímu a ich úlohami na tomto projekte.

2.1 Predstavenie členov tímu

Bc. Ivan Andrejkovič

Študent druhého stupňa vysokoškolského štúdia na Fakulte informatiky a informačných technológií STU v BA. Dlhodobo aktívne programuje v c a venuje sa etickému hackingu a penetračnému testovaniu.

Bc. Tadeáš Broniš

Absolvent bakalárskeho štúdia na Fakulte Informatiky a Informačných technológií STU. Vo svojej bakalárskej práci sa venoval prehľadávania územia pomocou jedného, či viacerých dronov. Počas vyhotovenia tejto práce sa oboznámil s niekoľkými vyhľadávacími algoritmi a naučil sa programovať v jazyku Python. Počas bakalárskeho štúdia sa naučil programovať v jazykoch JavaScript, Java, C#. Momentálne je študentom prvého ročníka inžinierskeho štúdia v obore Internetové Technológie na FIIT STU.

Bc. Gabriela Hózová

Študentka prvého ročníka inžinierskeho štúdia v odbore Inteligentné softvérové systémy na FIIT STU. Na tejto fakulte absolvovala aj bakalársky stupeň v odbore Informatika. Jej bakalárskou prácou bolo vytvorenie systému na prípravu vzoriek pre porovnanie podobností. Počas štúdia na fakulte získala skúsenosti v programovaní v jazykoch C, Java, Python, SQL. Téma 3D virtuálnej prehliadky ju zaujala z dôvodu čoraz väčšej prístupnosti tejto technológie bežnému používateľovi.

Bc. Nikolas Janec

Bakalárske štúdium absolvoval na FIIT STU v odbore Internetové technológie. Jeho bakalárskou prácou bolo vytvorenie serverovej aplikácie pre autentifikáciu používateľov a následnú správu ich prístupových práv vo viacerých objektoch. Počas štúdia na fakulte získal skúsenosti s programovaním v PHP, Java, C#, C++, C.

Bc. Jaroslav Lišiak

Bakalárske štúdium ukončil na FIIT STU. Záverečná práca bola zameraná na návrh a implementáciu vlastného prototypu IoT zariadenia, komunikujúceho prostredníctvom technológie LoRa. Bol spoluautorom dvoch komunikačných protokolov, ktoré boli použité v

bakalárskej práci. Navrhnuté protokoly boli prezentované na konferencii IIT.SRC 2017. Najviac skúseností má s programovacími jazykmi C#, Java, a C.

Bc. Tomáš Ofčarovič

Študuje v prvom ročníku inžinierskeho štúdia vzdelávacieho programu Internetové technológie na FIIT STU. Bakalársku prácu spracoval na tému „Pokročilé monitorovanie sietí v reálnej prevádzke“, kde sa systém na monitorovanie prevádzky siete stále používa do. Počas školy nadobudol mnoho zaujímavých poznatkov o programovaní v jazykoch C, JAVA, C#, JavaScript a SQL. Ďalšími poznatkami sú sieťové technológie a ich implementácia, v ktorých sa zdokonaľuje účasťou na certifikovaných kurzoch.

Bc. Michal Škuta

Študent inžinierskeho štúdia na Fakulte Informatiky a Informačných Technológií Slovenskej technickej univerzity v Bratislave, ktorý sa amatérsky zaujíma o prácu s OpenGL. Popri škole sa zoznámil s množstvom sieťových technológií potrebných na fungovanie internetu.

2.2 Rozdelenie rolí v tíme ZS

Člen tímu	Dlhodobá úloha	Krátkodobá úloha
Bc. Ivan Andrejkovič <i>(manažér testovania)</i>	Vývojár Tester	Analýza problémovej oblasti Zapisovateľ Informovanie o aktuálnom stave projektu
Bc. Tadeáš Broniš <i>(manažér kvality)</i>	Správca architektúry projektu	Testovanie nástrojov potrebných na prácu Zapisovateľ Tvorca retrospektív
Bc. Gabriela Hózová <i>(manažér komunikácie)</i>	Dohliadanie na komunikáciu v tíme, dodržiavanie úloh a zodpovedností Správca projektovej dokumentácie	Tvorca retrospektív Zapisovateľ Analýza problémovej oblasti Scrum master
Bc. Nikolas Janec <i>(manažér dokumentácie)</i> tvorby	Správca fotografických podkladov pre tvorbu 3D modelov	Zapisovateľ Analýza problémovej oblasti Informovanie o aktuálnom stave projektu
Bc. Jaroslav Lišiak <i>(manažér úloh)</i>	Administrátor tímového webu (serveru)	Dohliadanie na dodržiavanie termínov Zapisovateľ Tvorca web stránky
Bc. Tomáš Ofčarovič <i>(manažér plánovania)</i>	Dohľad nad testovaním	Tvorca retrospektív Tvorca web stránky Zapisovateľ
Bc. Michal Škuta <i>(manažér vývoja)</i>	Správca repozitára Vývojár Tester	Analýza problémovej oblasti Zapisovateľ Informovanie o aktuálnom stave projektu

2.3 Rozdelenie rolí v tíme LS

Člen tímu	Úloha
Bc. Ivan Andrejkovič	Vývojár Tester
Bc. Tadeáš Broniš	Dokumentátor Grafické úlohy Tester
Bc. Gabriela Hózová	Dohliadanie na komunikáciu v tíme, dodržiavanie úloh a zodpovedností Správca projektovej dokumentácie Správca grafických a marketingových úloh Tester
Bc. Nikolas Janec	Vývojár Grafické úlohy
Bc. Jaroslav Lišiak	Administrátor tímového webu (serveru) Dokumentátor
Bc. Tomáš Ofčarovič	Dohľad nad testovaním Správca fotografických podkladov pre tvorbu 3D modelov Tvorca 3D modelov
Bc. Michal Škuta	Správca repozitára Manažér vývoju Tester

2.4 Podiel práce na jednotlivých častiach dokumentácie

Bc. Ivan Andrejkovič

- Dokumentácia inžinierskeho diela: analýza, návrh, implementácia

Bc. Tadeáš Broniš

- Dokumentácia inžinierskeho diela: návrh, testovanie
- Metodiky
- Inštalačná príručka

Bc. Gabriela Hózová

- Dokumentácia riadenia: Aplikácia manažmentov, sumarizácia šprintov, globálna retrospektíva
- Dokumentácia inžinierskeho diela: analýza
- Metodiky

Bc. Nikolas Janec

- Dokumentácia riadenia: úvod
- Dokumentácia inžinierskeho diela: úvod, analýza

Bc. Jaroslav Lišiak

- Dokumentácia inžinierskeho diela: testovanie
- Metodiky
- Používateľská príručka

Bc. Tomáš Ofčarovič

- Dokumentácia riadenia: metodiky
- Dokumentácia inžinierskeho diela: analýza
- Metodiky

Bc. Michal Škuta

- Dokumentácia inžinierskeho diela: analýza, implementácia
- Metodiky

3 Aplikácie manažmentov

3.1 Manažment dokumentácie

V rámci manažmentu dokumentácie vytvárame jednotlivé dokumenty. Priebežne vytvárame zápisnicu z každého tímového stretnutia, retrospektívu po každom šprinte a dokumentáciu zdrojového kódu. Tieto dokumenty sa buď priebežne menia, alebo sa priebežne vytvárajú nové verzie podľa potreby. Ďalšie vytvárané dokumenty sú napríklad tímové metodiky, dokumentácia riadenia a dokumentácia inžinierskeho diela. Každý z vytváraných dokumentov v rámci tímového projektu má zadefinovanú šablónu, podľa ktorej sa tento dokument vytvára. Šablóny sme si zadefinovali v metodike dokumentácie.

3.2 Manažment komunikácie

Komunikáciu môžeme rozdeliť na komunikáciu na stretnutí a komunikáciu mimo stretnutí. V rámci komunikácie na stretnutí má každý člen tímu priestor povedať o svojich výsledkoch a problémoch, ktoré nastali, a ak je to nutné, otvoríme priestor pre hlbšiu diskusiu o vzniknutých problémoch. Komunikácia mimo stretnutia funguje hlavne pomocou nástroja Slack. Pre rôzne účely sme vytvorili viacero kanálov podľa druhu témy, ktorej sa daná komunikácia týka. Komunikácia s produktovým vlastníkom projektu je taktiež veľmi dôležitá. Produktový vlastník tohto projektu je zároveň vedúci nášho tímu, a tak s ním komunikujeme taktiež najmä pomocou nástroja Slack. Vedúceho taktiež informujeme o aktuálnom stave projektu a úloh na pravidelných týždenných tímových stretnutiach.

3.3 Manažment úloh

V rámci manažmentu úloh je vytvorená metodika manažmentu úloh, ktorá opisuje postupy na správu úloh v nástroji GitLab. Je potrebné mať určené ako vytvárať, priradovať a dokončovať úlohy. Úlohy sa vytvárajú na základe konzultácii na začiatku šprintu ale aj počas šprintu. Metódou "planning poker" určujeme náročnosť úloh. Úlohy sú priradované členom tímu počas plánovania šprintu a vykonávajú sa podľa priority. Priorita úloh sa určuje podľa ich uloženia v "To Do" sekcii. Manažér úloh je zodpovedný za správu všetkých úloh v projekte v spomínanom nástroji GitLab.

3.4 Manažment testovania

Testovanie v našom projekte prebieha podľa testovacích scenárov opísaných v metodike testovania. Testovanie je zabezpečené pomocou HMD a ovládačov. Projekt je konkrétne testovaný pomocou HTC Vive. Pri testovaní sa zameriavame na funkčnosť projektu podľa

požiadaviek produktového vlastníka ale aj na vizuálny výstup. Pri testovaní je prítomný autor testovanej časti a nezávislý tester. Testovanie vizuálneho výstupu v sebe zahŕňa napríklad testovanie plynulosti zobrazovania, prekrývania objektov a podobne.

3.5 Manažment vývoja

V našom projekte pracujeme paralelne na dvoch spôsoboch riešenia. Tým, že máme dva možné spôsoby ako problém vyriešiť, rozhodli sme sa pracovať na dvoch prototypoch. Počas postupného vyvíjania oboch prototypov zistíme, ktorý je viac vhodný pre výsledný produkt. Vývoj na projekte je kontrolovaný manažérom vývoja, ktorý sa stará o to, aby boli dodržiavané pravidlá z metodiky písania kódu a aby boli vyvíjané časti včas hotové.

3.6 Manažment kvality

Na zabezpečenie dostatočnej kvality jednotlivých úloh máme zadefinované rôzne metodiky, ako napr. metodiku revidovania, metodiku verziovania, metodiku písania kódu a taktiež tímový “definition of done”.

“Definition of done” pre používateľský príbeh (user story):

Súbor úloh (story) spĺňa tímovú “definition of done” vtedy, keď:

1. Prejde testami, ktoré boli pre dané úlohy vytvorené
2. Kód bol zrevidovaný aspoň jedným ďalším členom tímu, ktorý sa nepodieľal na tvorbe kódu
3. Spĺňať požiadavky používateľského príbehu (akceptačné kritériá od produktového vlastníka)

“Definition of done” pre úlohy:

1. Kód implementovaný pre danú úlohu musel byť zrevidovaný
2. Ak sú pre túto úlohu vytvorené testy, kód musí byť taktiež otestovaný

“Definition of done” pre úlohy typu Story:

Dokončenie všetkých úloh typu Task, ktoré súvisia s danou úlohou.

3.7 Manažment plánovania

Na začiatku semestra boli vytvorené základné používateľské príbehy produktu aj s popisom, teda minimálne požiadavky, ktoré by mali byť splnené. Na stretnutí na začiatku šprintu vždy podľa priority používateľských príbehov zoberieme ich časť, pre ktorú sa zaviažeme, že ju

stihneme spraviť. Tieto používateľské príbehy následne rozdelíme na menšie úlohy, ktoré obsahujú všetko to, čo je potrebné splniť pre úspešné vykonanie používateľského príbehu. Jednotlivé úlohy sa pridelia členom tímu tak, aby každý člen tímu mal určenú prácu na šprint a taktiež tak, aby sme vedeli, že má predstavu o tom, čo bude robiť. Ku každej úlohe je uvedený jej názov, popis (description) a odhad (v počte story pointov), koľko bude táto úloha trvať. Pred tým, ako sa začne robiť na úlohe, musí mať úloha jasne určené, kto na nej bude pracovať. Po uplynutí prvého týždňa šprintu sa stretávame na tímovom stretnutí, kde si navzájom prezentujeme pokroky v našich prácach na projekte alebo riešime vzniknuté problémy. Ak je to potrebné, vytvoríme ďalšie úlohy, ktoré sme pri prvotnom plánovaní nevideli.

Úlohy, ktoré boli členom tímu zadané postupne prechádzajú určitými stavmi, pričom dodržiavame pravidlá o zmene stavu úlohy určené v metodike manažmentu úloh. Na začiatku má úloha stav "To Do", po ňom nasleduje "Doing", "Review", "Testing" a následne "Closed". Sledovaný je taktiež aj stav šprintu. Tento stav je reprezentovaný burndown grafom. Každý šprint má svoj vlastný burndown graf, na ktorom sledujeme, koľko percent zo šprintu je hotových. Po každom šprinte na stretnutí tím prezentuje produktovému vlastníkovi nový prírastok k produktu a aj všetky splnené a nesplnené úlohy. Všetky informácie týkajúce sa plánovania a stavov úloh sú uložené v nástroji GitLab.

4 Sumarizácie šprintov

Kapitola obsahuje sumarizáciu jednotlivých šprintov. Každý šprint je po jeho ukončení zhodnotený celým tímom na stretnutí. Retrospektíva šprintu je následne zapísaná a uverejnená na webovej stránke tímu.

4.1 Šprint 1 - Scooby-Doo

Pri prvom tímovom stretnutí sme sa lepšie zoznámili medzi sebou v tíme, s vedúcim tímu a v neposlednom rade s našou témou. Pri diskutovaní o téme a o predstavách a požiadavkách zákazníka (vedúci tímu) sme si určili základné úlohy, ktoré sme sa zaviazali v prvom šprinte splniť. Všetky úlohy sa nám nepodarilo dokončiť včas, takže sme boli nútení niektoré z nich presunúť do nasledujúceho šprintu. Cieľom tohto šprintu bolo najmä zanalyzovať oblasť, do ktorej naša téma zapadá. Oboznámili sme sa tiež s nástrojmi využívanými pri práci na projekte, pri manažmente a tiež pri komunikácii v tíme. Vytvorili sme a nasadili webovú stránku tímového projektu. Každý z nás ako jednotlivec zhodnotil tento prvý šprint a následne sme spravili spoločnú retrospektívu, kde sme navrhli, čo by bolo vhodné v budúcich šprintoch vylepšiť alebo prestať robiť. Spolu s nedokončenými úlohami sme vytvorili úlohy na ďalší šprint a tie si medzi sebou rozdelili. Najväčší problém, ktorý sme identifikovali, bol problém v komunikácii medzi členmi tímu.

4.2 Šprint 2 – Lajka

Počas druhého šprintu sme pokračovali v hlbšej analýze oblasti a začali sme pracovať na prototypu webovej aplikácie. Pri tvorbe prototypu sme narazili na niekoľko problémov týkajúcich sa zhotovovania 3D modelov objektov, ktoré sa nachádzajú v 3D labe. Nástroje, ktoré sme plánovali použiť na vytvorenie 3D modelov sa dajú zaradiť do dvoch kategórií. V tomto šprinte sme zistili, že výsledky nástrojov na skenovanie objektov nevyzerajú podľa našich predstáv a rozhodli sme sa v nasledujúcom šprinte použiť namiesto toho techniku fotogrametrie. Začalo sa taktiež pracovať na vytváraní jednotlivých metodík, podľa ktorých sa budeme riadiť počas celého vývoja. Počas retrospektívy sme opäť prediskutovali plusy a mínusy uplynulého šprintu. Problém s komunikáciou, ktorý sa v minulom šprinte objavil, sa nám v tomto šprinte podarilo eliminovať a komunikácia sa citeľne zlepšila.

4.3 Šprint 3 – Bobi

V priebehu tretieho šprintu pokračovali práce na prvom prototypu z minulého šprintu a taktiež bol vytvorený druhý prototyp. Začali sme pracovať na vizualizácii priestoru pomocou 360° fotografie. Doterajšie práce boli snahou o vymodelovanie celého priestoru pomocou 3D

objektov. Cieľom tretieho šprintu bolo taktiež aj dokončiť všetky metodiky. Všetky úlohy sa nám nepodarilo dokončiť včas, nakoľko sa nám skomplikovala situácia odchodom jedného člena tímu. Boli sme teda nútení niektoré z nich presunúť do nasledujúceho šprintu. Tento šprint bol pre nás najmenej úspešný z celého semestra a nastalo najviac komplikácií. Problémy, ktoré sme identifikovali sa nám podarilo odstrániť v ďalších šprintoch. Tieto problémy sa týkali najmä spracovávanía fotografií a to nás veľmi brzdilo.

4.4 Šprint 4 – Lassie

Štvrtý šprint sme začali refaktoringom zdrojového kódu. Prvý prototyp bol totiž vyvíjaný ešte bez použitia metodík a kód sme museli upraviť, aby spĺňal definované pravidlá. 3D modely vyhotovené v predchádzajúcich šprintoch nemali dostatočnú kvalitu. Bolo to spôsobené veľkými bielymi plochami, ktoré mal program pri zhotovovaní modelu problém identifikovať, čím vznikali prekryvy alebo hluché miesta. Na biele plochy sme rozmiestnili unikátne značky, čím sme zabezpečili ich jedinečnosť a program ich následne vedel správne identifikovať a korektne s nimi pracovať. Vygenerovali sme správne modely jednotlivých častí miestnosti. Ďalej sa nám podarilo dokončiť prototyp z 360° fotografií. Pre presun v rámci miestnosti sme v prototypu vytvorili špeciálne body, "teleporty". Pomocou "teleportov" sa môže používateľ jednoducho pomocou VR ovládača alebo štandardnej počítačovej myši v miestnosti intuitívne presúvať. Počas dodatočného testovania sme v prototypu objavili chyby, ktoré sme samozrejme odstránili. Na základe toho sme usúdili, že je potrebné navrhovať/vykonávať dôkladnejšie testy, čím vieme odhaliť väčšie množstvo chýb.

4.5 Šprint 5 – Odie

V poslednom šprinte zimného semestra bolo našou hlavnou prioritou upravenie zhotovených 3D modelov. Tie sme upravovali v programe Blender, čo je open source nástroj pre 3D tvorbu. Nakoľko však žiaden člen tímu nemal predchádzajúce skúsenosti s podobným nástrojom, bolo potrebné si program naštudovať. Tento fakt výrazne spomalil dokončenie nadväzujúcich úloh. V neposlednom rade bolo potrebné dokončiť výstupnú dokumentáciu po uzavretí všetkých šprintov zimného semestra. V rámci posledného stretnutia sme prediskutovali aktuálne používané metodiky, ako aj nami vytvorené riešenie. Všetky pripomienky i nápady pre zlepšenie, či už fungovania tímu alebo aktuálneho prototypu boli zapísané a budú východiskovým bodom prvého šprintu letného semestra.

4.6 Šprint 6 – Rex – Tedy

V zimnom semestri sme si role v tíme menili tak, aby si každý člen tímu postupne vyskúšal všetky role. Naopak teraz, po skúsenostiach zo semestra sme tento koncept prehodnotili a počas prvého stretnutia, kde sme sumarizovali našu prácu na projekte počas zimného semestra, bola každému členovi pridelená jedna úloha, ktorú bude zastávať až do ukončenia projektu. Jedným z ďalších cieľov projektu bolo vytvoriť „administračné rozhranie“, aby správca stránky vedel jednoducho editovať vlastnosti vizualizovaného 3D priestoru. Počas tohoto šprintu sme napríklad po niekoľkých testovaniach navrhli štruktúru JSON súboru, z ktorého budeme načítavať vlastnosti objektov v 3D priestore. Správca stránky bude môcť v tomto JSON súbore nastavovať napríklad počet, veľkosť a pozíciu objektov vo vizualizovanom priestore.

4.7 Šprint 7 – Beethoven – Tedy

V druhom šprinte letného semestra sme sa sústredili na rozšírenie funkcionality editora. Pretvorili sme ho tak, aby bolo možné editovať vlastnosti objektu nie len pomocou JSON súboru, ako bolo uvedené v predchádzajúcej kapitole, ale aby mohol používateľ meniť vlastnosti objektu priamo vo vizualizovanom priestore pomocou ovládačov. Taktiež sme implementovali zobrazovanie informácií o objekte po kliknutí na objekt. Informácie o objekte sú zahrnuté v rámci JSON súboru, odkiaľ sa načítavajú všetky informácie o objektoch.

4.8 Šprint 8 – Hačikó

V tomto šprinte sme si dali za cieľ dokončiť zobrazovanie textových informácií o objekte. Na tímovom stretnutí sme diskutovali o umiestnení textu. Najlepšie návrhy sme otestovali v reálnom prostredí a následne sme diskutovali o tom, ktoré umiestnenie je pre náš projekt najlepšie. Následne sme sa zhodli na umiestnení textu na ovládači. Začali sme tým, že sme začali zobrazovať samotný text. Po diskusií sme sa rozhodli pridať za text pozadie, aby bol zobrazený text lepšie čitateľný. Pri zobrazení veľmi rozsiahleho textu nebolo možné prečítať celý text, a tak sme pridali možnosť posúvania (scrollovania) textu. Počas tohto šprintu sme taktiež začali pracovať na pridávaní základov kolízií. Pridali sme základnú funkcionality, ktorá zabezpečovala zastavenie objektu pri kolízií s kolíznou plochou.

4.9 Šprint 9 – Pluto

V tomto šprinte sme pracovali na pokročilejšej kolízií objektov. Do virtuálneho priestoru sme pridali kolízne plochy na miesta, pri ktorých sme chceli vytvoriť kolíziu. Taktiež sme vylepšili funkcionality kolízií, tak aby bolo možné pri kolízií v jednom smere posúvať daný objekt v

iných smeroch, v ktorých kolízia nenastala. Takto sme umožnili „posúvanie“ objektu po kolíznej ploche.

Pre lepšiu možnosť informatizácie používateľa sme sa v tomto šprinte rozhodli pridať zobrazovanie videí a 360° videí. Dvojrozmerné video je zobrazované na ovládači, na rovnakom mieste, na ktorom je zobrazovaný text. 360° video je premietané na plochu gule, do ktorej je používateľ premiestnený. Pri takomto zobrazovaní videa sa používateľ môže ľubovoľne otáčať.

Počas tohto šprintu sme pridali aj základy gravitácie. Pridali sme jednoduchú funkcionality, ktorá zabezpečovala, že objekty umiestnené vo vzduchu budú „padať“ až do kým nenastane kolízia s kolíznou plochou (podlahou, stolom, skriňou, atď.)

4.10 Šprint 10 – Lady a Tramp

V poslednom šprinte sme pracovali na pridávaní zvukových a obrazových efektov. Pri niektorých objektoch sa pri ich detaile zobrazí obrazový detail ako napríklad bližšia fotografia alebo príklad použitia objektu. Tak isto pri niektorých objektoch dáva najväčší zmysel pri ukázaní detailu spustenie zvukovej stopy. Tieto efekty sa nám do projektu podarilo úspešne zakomponovať.

Veľmi podstatnou úlohou, ktorou sme sa zaoberali počas celého času prác na projekte bolo vytvorenie modelu miestnosti 3D Labu. Počas prác na projekte sme skúšali mnohé techniky ako vytvárať 3D modely objektov. Zistili sme, že na malé objekty ako sú počítače, kamery, komponenty virtuálnej reality ako sú okuliare, ovládače alebo lightboxy je ideálne vytvoriť pomocou metódy fotogrametrie. Na rozdiel od týchto menších predmetov, pri veľkých predmetoch a samotnej miestnosti sme nedostali očakávané výsledky a rozhodli sme sa sami tieto modely vymodelovať pomocou programu Blender. Textúry k objektom modelovaným v programe Blender sme získali z fotografií a pomocou UV máp.

5 Používané metodiky

Na bezproblémovú kolaboratívnu prácu na projekte, bolo nutné zaviesť určité pravidlá. Preto sme sa riadili niekoľkými metodikami.

V projekte sme zadefinovali nasledovné metodiky:

- Metodika dokumentácie
- Metodika písania kódu
- Metodika komunikácie
- Metodika revízií
- Metodika verziovania
- Metodika testovania
- Metodika manažmentu úloh

Táto kapitola sa venuje stručným opisom použitých metodík. Plné znenie príslušnej metodiky sa nachádza v prílohe A tohto dokumentu.

5.1 Metodika dokumentácie

V metodike dokumentácie sú definované konvencie písania dokumentu, rozsah dokumentu, forma a štýly pre jednotlivé dokumenty. Okrem dokumentácie riadenia a dokumentácie inžinierskeho diela sa spracovávajú po každom stretnutí zápisnice a po ukončení špritu sa spíše retrospektíva špritu. Po spísaní dokumentu je posunutý ostatným členom tímu pre zhodnotenie, odsúhlasenie, prípadne upravenie do finálnej podoby. Tieto dokumenty sú potom vložené na webové sídlo nášho tímu. Dokumentácia zdrojového kódu je riešená formou komentárov v zdrojovom kóde pre jednoduché pochopenie pridaného kódu.

5.2 Metodika písania kódu

Súčasťou tejto metodiky je konvencia písania kódu. Definovali sme, akým spôsobom budeme písať názvy premenných, formátovanie blokov zdrojového kódu ale taktiež aj štruktúru súborov a adresárov v repozitári na našom serveri na GitLabe. Najdôležitejšou súčasťou zdrojového kódu sú komentáre, ktoré napomáhajú ostatným ľuďom pomôcť rýchlo pochopiť upravený kód. Metodika je záväzná pre všetky zdrojové kódy, ktoré vzniknú počas prác na projekte a platí pre všetkých členov, ktorí sú súčasťou tímu. Určili sme programovací jazyk, ktorý sa bude v rámci tímového projektu využívať, a je ním Javascript.

5.3 Metodika komunikácie

Metodika komunikácie má za cieľ definovanie procesov komunikácie, ktoré v rámci tímového projektu uplatňujeme. Medzi hlavné nástroje na komunikáciu v tíme je nástroj na komunikáciu Slack s mnohými komunikačnými kanálmi, v ktorých sa riešia príslušné témy. Slack používame aj na formálnu aj na neformálnu komunikáciu. Ďalším nástrojom na komunikáciu v tíme je náš nástroj na manažment projektu Gitlab. V Gitlabe komunikujeme pomocou komentárov na daný problém (Issue), alebo vyberáme úlohu zo zoznamu úloh “na spracovanie” a presunujeme ju do zoznamu “práve spracovávané”. Podrobný popis práce s nástrojmi je popísaný v prílohe. Na instantnú komunikáciu je možné zvoliť aj telefonát, nakoľko každý zverejnil svoje telefónne číslo, na ktorom je dosiahnuteľný v prípade urgentnej potreby.

5.4 Metodika revízií

Metodiky revízií sme rozdelili do dvoch častí, a to:

- revízia dokumentácie
- revízia zdrojového kódu

Zdrojový kód musí pred spojením s hlavnou vetvou zrevidovaný. Ak boli vyžadované zmeny, je potrebné zdrojový kód upraviť podľa požiadaviek až do podoby, pokým zdrojový kód kontrolór neschváli. Dodržiavanie jednotlivých pravidiel písania kódu aj dokumentácie zvyšuje celkovú kvalitu a kód bude prehľadnejší a ľahší na pochopenie, preto je revízia dokumentu aj kódu dôležitá.

5.5 Metodika verziovania

Účelom tejto časti je prezentovanie pravidiel, ktoré musia byť dodržiavané pri verziovaní zdrojového kódu v projekte. Na manažment verzií využívame systém Git. V tejto metodike je popísané, ako postupovať od pridelenia novej úlohy, cez odovzávanie, požiadanie o predkladania príspevkov, až po spojenie novej vetvy pre danú úlohu s vyššou vetvou. Je tu dôkladne opísaný postup, ktorým musia autori kódu prejsť, aby vytvorili nový predklad príspevku.

5.6 Metodika testovania

Táto metodika určuje spôsob akým je nutné testovať funkcionálnosť pred tým, ako sa zmena v projekte schváli a integruje sa do vyššej vetvy. Uviedli sme kto bude zodpovedný za test časti

kódu, kedy bude testovať kód, a akým spôsobom bude vykonávať testy. Táto metodika sa dodržiava počas celého vývoja projektu, a platí pre každého testera.

5.7 Metodika manažmentu úloh

Táto časť sa venuje opisu metodiky manažmentu úloh v rámci práce na tímovom projekte. Ide o úlohy pri tvorbe backlogu, šprintov, úloh, príbehu, ale aj odstraňovaní a úprave úloh v nástroji Gitlab. Metodika je určená primárne pre manažéra úloh, ale slúži všetkým členom tímu.

6 Globálna retrospektíva

Aplikovaná metóda vývoja nášho projektu bola Scrum. Poctivo sme dodržiavali pravidlá a zásady počas všetkých desiatich uplynulých šprintov. Každý šprint trval 2 týždne (14 dní) a posledný šprint v zimnom semestri - piaty šprint bol skrátený na 12 dní. Na začiatku každého šprintu sme absolvovali stretnutie s vlastníkom produktu (vedúci tímu) a určili sme ciele na celý šprint. Pomocou týchto cieľov sme identifikovali úlohy, ktoré sme museli vypracovať aby boli ciele splnené. Úlohy sme si napokon rozdelili medzi všetkých členov spravodlivo. Úlohy sme rozdeľovali na základe preferencií a schopností členov tímu. Nestalo sa, že by niekto protestoval s pridelenou úlohou. Po uplynutí šprintu prebehla retrospektíva, na ktorej sme zhodnotili celý šprint najprv jednotlivo každý člen a následne aj spoločne ako celý tím.

Prvé dva šprinty sme venovali najmä analýze pracovnej oblasti, do ktorej zapadá náš projekt, a to konkrétne virtuálna realita, technika fotogrametrie, nástroje na tvorbu 3D modelov a už spomínanú fotogrametriu a taktiež sme do analýzy zhrnuli možnosti implementácie. Pri plánovaní druhého šprintu sme definovali nové požiadavky na MVP pre zimný semester. V polovici druhého šprintu sme začali získané informácie aplikovať aj prakticky a začali sme vytvárať prototyp výstupného produktu. V treťom šprinte sme sa analýze venovali už iba minimálne a viac sme sa zamerali už na samotnú implementáciu v dvoch rôznych prototypoch. Počas štvrtého šprintu sme sa venovali najmä refaktoringu kódu, čo viedlo k zvýšeniu prehľadnosti. Taktiež sa nám podarilo vytvoriť funkčný prototyp z 360° fotografií, kde pre presun v rámci miestnosti boli použité "teleporty". Posledný šprint zimného semestra bol zameraný na dokončenie všetkých aktívnych úloh a zhrnutie dosiahnutých výsledkov. Na poslednom stretnutí boli prediskutované nápady ako zlepšiť súčasné riešenie ako aj úprava súčasných metodík pre zvýšenie produktivity. Počas práce na projekte sme zaviedli metodiky, ktoré sme sa snažili dodržiavať celý semester.

Na začiatku letného semestra sme sa rozhodli začať skúšať aj iné metódy ako vytvoriť 3D modely a venovali sme tomu veľa času. Okrem metódy fotogrametrie sme skúšali rôzne softvéry na skenovanie objektov alebo na fotografovanie a následné spojenie fotografií do 3D objektu. Ďalšou možnosťou ktorú sme skúšali bolo vytváranie 3D modelov pomocou 360° fotografie priestoru a modelovanie priestoru a objektov v programe Blender. Po testovaní a vytváraní prototypov pomocou každej metódy sa najviac osvedčila metóda modelovania v programe Blender a už spomínaná metóda fotogrametrie. Pre menšie objekty bola metóda fotogrametrie s menšími úpravami úplne ideálna, čo sa bohužiaľ ale nedá povedať o modeli celej miestnosti. Na tú sme zvolili metódu vymodelovania objektu v programe Blender. Najväčšie problémy s fotogrametriou pri modeli miestnosti boli kvôli nedostatočným textúram na povrchu objektov v miestnosti. Vyskúšali sme aj použitie označení na stenách ako je uázané na obrázku č. ... ale bez úspechu. V skutočnosti rovné povrchy boli v modeli hrboľaté alebo „deravé“. V programe Blender boli textúry na konkrétny objekt získané

z fotografií a vložené pomocou UV máp. Nakoľko sme mali v letnom semestri pridelené úlohy medzi sebou podľa našich schopností a znalostí, ktoré sme zistili v zimnom semestri, pracovalo sa nám jednoduchšie a rýchlejšie. Všetku požadovanú funkcionality sme do produktu zahrnuli a snažili sme sa splniť všetky požiadavky vlastníka produktu (vedúceho), čo sa nám aj podarilo. Na projekte sa dá samozrejme ďalej pracovať, zdokonaľovať ho a pridávať novú funkcionality, či aplikovať novú metódu zobrazovania informácií na iný typ webovej stránky.

Po zimnom semestri:

Čo chceme robiť do budúcnosti:

- dôkladné dokumentovanie zdrojového kódu
- efektívnejšie plánovanie šprintov
- aplikovať párové programovanie
- konzultovanie pomocou spoločného hovoru ak nie je možné osobne sa stretávať
- monitorovanie času stráveného pri práci na úlohách pre odhalenie slabých miest

V čom chceme pokračovať:

- komunikácia cez nástroj Slack
- rozdeľovanie úloh podľa preferencií členov tímu
- stretnutia viackrát do týždňa

S čím chceme skončiť:

- komunikačný šum a rozptýlenie pri oficiálnych tímových stretnutiach

Po zimnom semestri sme si určili zopár bodov, ktoré sme chceli zmeniť, v ktorých sme chceli pokračovať alebo, s ktorými sme chceli prestať v zimnom semestri, tak ako je napísané vyššie. Z týchto bodov sa nám väčšina podarila splniť.

Dokumentácia k inžinierskemu dielu

7 Úvod

Obsahom tejto dokumentácie je pohľad na technickú stránku projektu vypracovaného v rámci predmetu Tímový projekt na tému Pohlcujúci web. Cieľom projektu je navrhnuť a vytvoriť nový druh webovej stránky, ktorá umožní prehľadávať informačný priestor v pohlcujúcom prostredí virtuálnej reality. Stránka bude slúžiť na prezentáciu spomínaného 3D Labu, jeho vybavenia a projektov realizovaných vďaka nemu. V kapitole 8 a 9 sú opísané ciele, ktoré sme si stanovili na zimný(kapitola 8) a letný(kapitola 9) semester. Obsahujú tiež požiadavky na systém a globálne ciele pre daný semester. Základným cieľom pre zimný semester je vytvorenie MVP prototypu aplikácie, avšak výstupom letného semestra je plnohodnotná aplikácia, ktorá spĺňa všetku funkcionality požadovanú zákazníkom. Obsahom kapitoly 10 je analýza projektu. V jej časti sa venujeme analýze virtuálnej reality, fotogrametrie pre tvorbu čo najlepších 3D modelov, použitými technológiami a multiplatformovosti. Kapitola 11 sa zameriava na celkový návrh systému, ktorého súčasťou je architektúra, definované prípady použitia a diagram tried. V kapitole 12 je opísaná implementácia a postupne všetky projekty, ktoré sme vyvíjali.

8 Ciele na zimný semester

Na začiatku prvého šprintu boli definované hlavné ciele na zimný semester:

- Analýza potrebných technológií
- Vytvorenie 3D modelu miestnosti 3D Lab
- Vytvorenie prototypu pokrývajúceho základnú funkcionality finálnej aplikácie

Po ukončení prvého šprintu sme vytvorili špecifikáciu výsledného produktu so zákazníkom (vedúci tímu). Definovali sme požiadavky na finálny projekt, vyhľadali dostupné a použiteľné technológie a nástroje na fotogrametriu, ktoré budú potrebné pri vývoji finálneho produktu.

Požiadavky na projekt

- zobrazenie vizualizovaného priestoru návštevníkovi prostredníctvom HMD a browsera
- navigácia návštevníka stránky vo vizualizovanom priestore
- možnosť interakcie návštevníka s objektami v scéne pomocou ovládačov
- pridávanie modelov
- pridávanie popisov a bližších informácií

MVP pre zimný semester

Za cieľ v zimnom semestri sme si vytýčili vytvorenie prototypu, ktorý bude musieť spĺňať nasledovné požiadavky:

- Interakcia používateľa s objektami v scéne

- Zobrazenie prostredia na základe reálneho pohybu osoby
- Vytvorenie prototypovej scény

9 Ciele na letný semester

Na začiatku letného semestra sme si definovali hlavné ciele pre daný semester a to:

- Plnohodnotné a univerzálne zobrazovanie informácií o objekte
- Vylepšenie interakcie a správania sa objektov
- Vytvoriť čo možno najreálnejšiu verziu danej miestnosti
- Pridať čo možno najlepšiu možnosť jednoducho a rýchlo upravovať virtuálny priestor.

Po ukončení zimného semestra sme analyzovali požiadavky, ktoré sme chceli splniť počas zimného semestra. Analyzovali sme naše nedostatky a naše slabiny a tie sme sa počas letného semestra snažili odstrániť. To ktoré nedostatky v našom tíme sa nám podarilo odstrániť je opísané v nasledujúcich kapitolách. Taktiež sme si spísali základné požiadavky na projekt v letnom semestri.

Požiadavky na projekt

- Umožniť jednoduchú úpravu 3D priestoru (editácia virtuálneho priestoru)
- Zobrazíť informácie o objekte formou textu, obrázku, zvuku, videa a 360° videa.
- Pridať kolíziu objektov s inými objektami
- Pridať základnú fyziku

MVP pre letný semester

Za cieľ v letnom semestri sme si vytýčili vytvorenie prototypu, ktorý bude musieť spĺňať nasledovné požiadavky:

- Interakcia používateľa s objektami v scéne
- Zobrazenie prostredia na základe reálnej predlohy
- Umožnenie jednoduchej editácie scény
- Pridanie základov kolízií a gravitácie
- Zobrazenie informácií o objekte

10 Analýza

Táto kapitola je zameraná na analýzu oblasti, do ktorej zapadá náš projekt.

10.1 Fotogrametria

Proces fotogrametrie sa zameriava na sériu fotografií z ktorých zostavuje presné polohy povrchových bodov a rekonštruje na základe polohy fotoaparátu, orientácie, typu objektívu, skreslenia šošoviek...

Ako nástroj na tvorbu 3D modelov z fotografií bol vybraný Autodesk ReCap na základe najlepšie dosiahnutých doterajších výsledkov. Pre čo najlepšie výsledné 3D moduly je potrebné dodržiavať nasledujúce pokyny:

Nastavenie fotoaparátu a základné pravidlá:

- Fotografie musia byť ostré po celej scéne nie len snímaný objekt (nevhodné fotografie zo série nech sú vymazané)
- Výsledné fotografie sa neupravujú
- Mal by byť použitý statív (každá fotografia v rovnakej výške, akýkoľvek pohyb môže mať vplyv na ostrosť fotografií)
- Ideálne je vyhotovovať fotografie s pevným objektívom, 50mm
- Priblíženie pri fotografovaní nesmie byť menené, pri potrebe treba presunúť fotoaparát bližšie alebo ďalej od objektu
- Počas rovnakej akvizície nesmie byť expozíciu zmenená
- ISO treba nastaviť na 100 (čím vyššie ISO tým väčší šum)
- Formát JPEG

Svetelné podmienky

- Je potrebné sa vyhnúť prostrediu, v ktorom svetlo vytvára veľké kontrastné tieň, pretože tieň sú jedným z najväčších nepriateľov fotogrametrie
- Nepoužívať blesk
- Vyhnúť sa časom počas najväčšieho pôsobenia slnka pri objektoch v exteriéri

Stratégie pri nastavovaní scény podľa objektov

Ak je objekt jednofarebný:

- Umiestniť ho na čo najviac farebné pozadie, alebo ak pozadie je jednofarebné tak pridať čo najväčšie množstvo rôznych farebných objektov, ktoré nebránia získaniu referenčných bodov

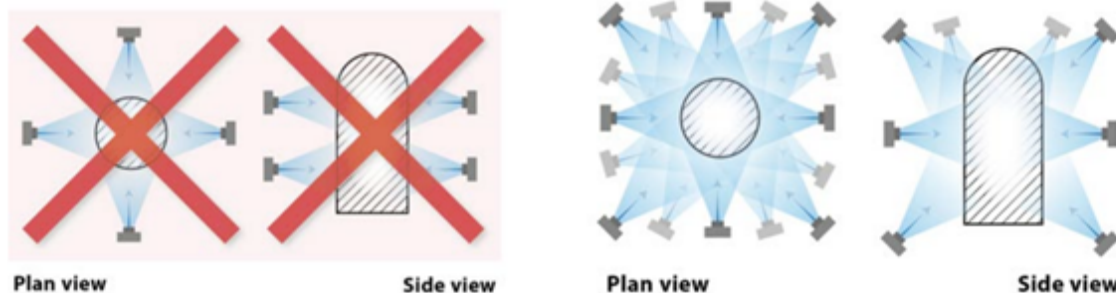
- Objekt by mal byť statický a pohybovať počas vyhotovovania fotografií by sa mal iba fotoaparát

Ak má objekt viac farieb:

- Umiestniť ho na nejaké jednofarebné pozadie - ideálne je ho umiestniť na biele pozadie
- Počas celého fotografovania je objekt statický

Všeobecne:

- Objekt by mal byť umiestnený v strede danej fotografie a pokrývať najmenej 70% z celej fotografie
- Pri fotografovaní nepohybujte s okolitými predmetmi, ktoré sú taktiež zachytené na daných fotografiách
- Snímajte celý objekt, aby ste zakryli každý jeho aspekt. Posúvať by ste sa mali po každých 5-15 stupňov v rovnakej výške z dôvodu dostatočného prekrytia záchytných bodov na fotkách. Pre čo najlepšiu úroveň detailov je potrebné fotografovať objekt v 2-3 rôznych výškach.
- Počet fotografií pre malé až stredné objekty 70-100, pre budovy 180-200, štýl fotografovania je ukázaný na obrázku č. 1

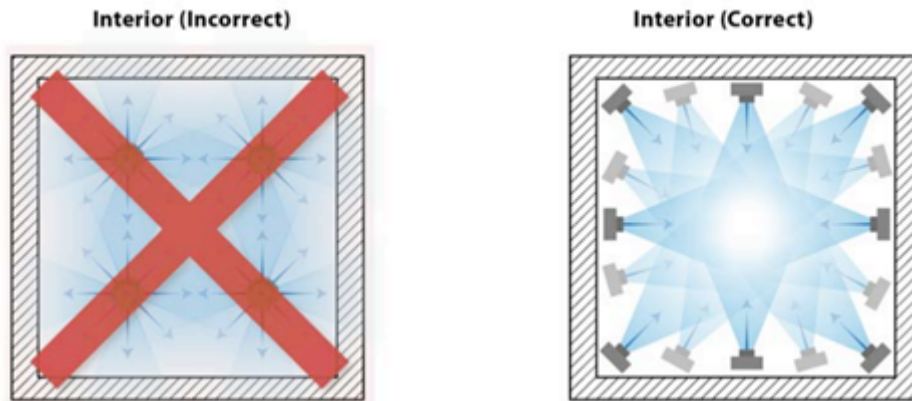


Obrázok č. 1: Ukážka štýlu fotografovania

- Čím viac fotografií tým lepšie ale stále platí podmienka aby sa fotografie dostatočne prekrývali

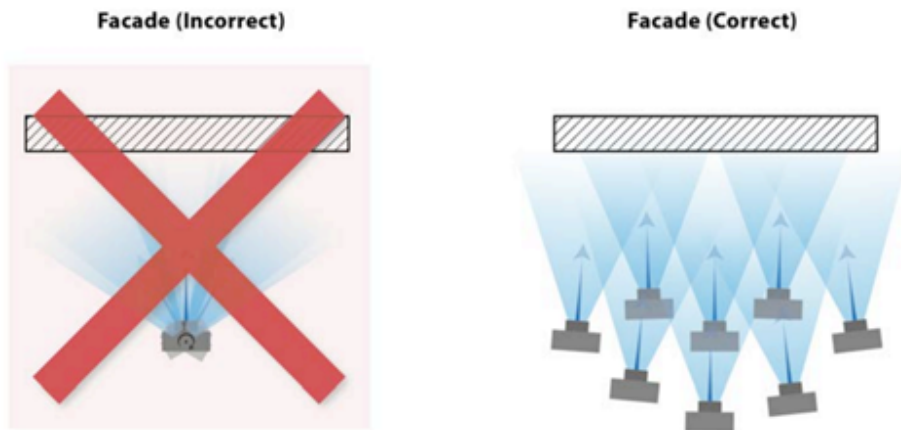
Miestnosti/Budovy

- Pri snímaní v interiéri treba postupovať podľa pokynov na obrázku č. 2. (Avšak pri interiéroch s prázdnyimi jednofarebnými stenami, zlyhajú všetky metódy fotenia, keďže obsahujú strašne málo záchytných bodov)



Obrázok č. 2: Ukážka správnej polohy fotoaparátu v interiéri

- Pri snímaní v exteriéri treba postupovať podľa pokynov na obrázku č. 3. (Pozor na pohybujúce sa objekty napr. chodci, vtáctvo...)



Obrázok č. 3: Ukážka správnej polohy fotoaparátu v exteriéri

10.2 Virtuálna realita

Všetko, čo poznáme o našej realite, prichádza prostredníctvom našich zmyslov. Z toho vyplýva, že naša celá skúsenosť s realitou je jednoducho kombinácia zmyslových informácií a nášho rozumu vytvárajúci zmysluplné mechanizmy pre tieto informácie. Je teda možné, že ak dokážeme prezentovať svoje zmysly s novo vytvorenými informáciami, naše vnímanie reality sa v reakcii na to zmení. Budeme mať predstavenú verziu reality, ktorá v skutočnosti neexistuje, ale z nášho hľadiska by bola vnímaná ako skutočná.

Z technického hľadiska v skratke je virtuálna realita termín, ktorý sa používa na opis trojrozmerného prostredia vytvoreného prostredníctvom počítača, ktoré môže človek preskúmať a s ním spolupracovať. Táto osoba sa stáva súčasťou tohto virtuálneho sveta,

alebo je ponorená do tohto prostredia a je schopná manipulovať s objektmi alebo vykonávať sériu akcií, pomocou ktorých dané prostredie skúma.

Čiže virtuálna realita je vytvorenie virtuálneho prostredia prezentovaného našim zmyslov takým spôsobom, že ho zažijeme, akoby sme boli naozaj v ňom. Na dosiahnutie tohto cieľa využíva množstvo technológií, ktoré musia zohľadňovať naše vnímanie a poznanie.

10.3 Technológie (nástroje)

Na vytvorenie 3D modelov či už miestnosti alebo objektov, ktoré sa v nej nachádzajú existuje niekoľko spôsobov. Ako prvý spôsob sme si vybrali skenovanie. Bolo vytvorených mnoho nástrojov na skenovanie objektov a následné vytváranie 3D objektov. Jedným z nich je Tango Point Cloud.

10.3.1 Tango Point Cloud

Aplikácia Tango umožňuje mobilnému zariadeniu snímanie hĺbky, respektíve vzdialenosti od objektu v reálnom čase. Technológia tango umožňuje snímanie objektov vzdialených od 0,5 m do 4m. Nie je teda vhodná na snímanie blízkych objektov. Tango snímané objekty prevádza do takzvaného "pointcloudu", teda do množiny bodov v priestore. Každý bod v takomto priestore má súradnice (x, y, z). Táto technológia je vhodná na snímanie jednoduchších objektov. Pri snímaní komplikovaných objektov vznikajú vo finálnom objekte defekty, ktoré je náročné odstrániť.

Po niekoľkých pokusoch sme túto aplikáciu zavrhlí, pretože výstup nebol postačujúci pre náš projekt.

10.3.2 3DSOM

Spoločnosť Creative Dimension Software Ltd vydalo 3DSOM pro, profesionálny nástroj na vytváranie a prezentáciu 3D modelov z fotografií. Sada nástrojov, ktorú softvér obsahuje, umožňuje používateľovi vytvárať modely, ako sú vázy, misy, alebo aj organické zložité objekty, ktoré sú tradične ťažko modelovateľné.

Ďalšou schopnosťou softvéru je modelovať objekty bez kalibračnej rohože (vytlačení list A4 umiestnený pod cieľovým objektom), ale aj napriek tomu dokáže zautomatizovať modelovanie. Kalibračný prístup funguje veľmi dobre pri malých objektoch, ale môže byť veľmi zložitý pri modelovaní väčších objektov, ako sú skrine, stoly, auto a pod.

Upustili sme od tohto nástroja, z toho dôvodu, že sa nám nepodarilo vygenerovať použiteľné objekty vo VR prostredí kvôli veľkým objektom, ktoré sa nachádzajú v 3D labe.

10.3.3 Scann3D

Scann3D je mobilná aplikácia od spoločnosti SmartMobileVision. Je funkčná iba na operačnom systéme Android. Využíva technológiu fotogrametrie na vytváranie 3D modelov z mobilného telefónu či tabletu. Na rozdiel od ostatných nástrojov Scann3D nepoužíva "cloud processing", čiže všetky výpočty sú realizované priamo na zariadení. Modely vytvorené pomocou tejto aplikácie sú najčastejšie využívané v rozšírenej alebo virtuálnej realite.

Tak ako všetky predchádzajúce programy, ani ten nemal požadovaný výstup pre náš projekt. Fotografie, ktoré boli predpripravené v príkladoch síce mali výstup pekný ale po použití nami vytvorených fotografií bol výstup nepoužiteľný.

10.3.4 3DF Zephyr

3DF Zephyr umožňuje automatické vytváranie 3D modelov z fotografií. Nie je potrebné žiadne manuálne upravovanie fotografií. 3DF Zephyr prichádza s jednoduchým používateľským rozhraním. Je vhodný na vytváranie 3D modelov z fotografií reálneho prostredia. 3DF Zephyr disponuje možnosťou exportovania do mnohých 3D formátov tiež vygenerovaním videa bez potreby externých nástrojov. Podporuje aj exportovanie "mesh" z fotografií do formátov ako: .ply, .obj/mtl a aj do nástroja Sketchfab. Verzia zadarmo umožňuje vytvárať 3D modely z maximálne 50-tich fotografií. Študentská verzia má tento limit nastavený na 500 fotografií.

Pri testovaní programu s nami vytvorenými fotografiami program zlyhal a výstup bol opäť nepoužiteľný.

10.3.5 Autodesk ReCap

Aplikácia Autodesk ReCap funguje v dvoch verziách, a to webová aplikácia a desktopová aplikácia pre Windows. Pomocou tohto nástroja sa nám podarilo po niekoľkých pokusoch vytvoriť model miestnosti, ktorý síce mal stále hluché miesta v textúre ale bol najlepší z doterajších modelov. Výsledný 3D model sme následne vložili do aktuálneho prototypu a pracovali s ním. Autodesk ReCap funguje na metóde fotogrametrie, ktorá je popísaná vyššie.

10.3.6 Walkabout Worlds

Softvér Walkabout Worlds používateľovi umožňuje rýchlo a jednoducho vytvárať 3D modely a 360° panoramatické virtuálne prehliadky skutočných miest. 3D modely sú vytvárané z 360° fotografií pomocou manuálneho upravenia a zarovnania. Softvér je zatiaľ v beta verzii ale pre doterajšie potreby na tomto projekte bola existujúca funkcionálna postačujúca.

10.3.7 Blender

Jedná sa o voľne dostupný open-source 3D nástroj, ktorý podporuje všetky spôsoby 3D modelovania, spracovávanía objektov, animácie a podobne. V našom projekte ho často využívame hlavne na vytváranie potrebných modelov a na editáciu modelov vytvorených pomocou fotogrametrie. Umožňuje nám upravovať veľkosť objektov, ich orientáciu a textúru. Taktiež umožňuje spájať viacero menších objektov do jedného väčšieho. Nakoľko sa jedná o softvér, ktorý je zadarmo a jeho používanie je veľmi všestranné je obľúbený aj komunitou. Preto je veľmi jednoduché nájsť potrebné rady a návody medzi komunitou. Čo nám počas implementácie veľmi pomáha.

10.3.8 RTAB-Map

RTAB-Map (Real-Time Appearance-Based Mapping) je RGB-D grafovo-orientovaný SLAM prístup založený na inkrementálnej slučke detektora uzávierky. Slučka detektora uzávierky používa prístup BoW (Bag-of-Words) pri určovaní, či nový obraz prichádza z predchádzajúcej alebo novej lokality. Keď je prijatá hypotéza slučky uzávierky, do grafu mapy sa pridá nové obmedzenie a optimalizátor grafov minimalizuje chyby na mape. Prístup na správu pamäte sa používa na obmedzenie počtu miest, ktoré sa používajú na detekciu slučiek uzávierok a optimalizáciu grafov, aby sa vždy rešpektovali obmedzenia v reálnom čase vo veľkom prostredí. RTAB-Map sa môže použiť samostatne s ručne držaným kinektom alebo stereo kamerou pre mapovanie.

Pri práci sme používali rovnomennú mobilnú aplikáciu a taktiež aj kameru ZED M, ktorá dokáže spolupracovať so softvérom založenom na RTAB-Map prístupe. Oba spôsoby mali pre našu prácu nedostačujúce výsledky, či už kvôli nedostatočnej pamäti mobilného zariadenia, s ktorým sme pracovali alebo s nedostatkami kamery, ktorá je zatiaľ novinkou a nemá odstránené niektoré chyby.

10.3.9 ZED Mini

ZED je stereo kamera od spoločnosti Stereolabs, ktorá umožňuje snímať až do vzdialenosti 20 metrov a môžeme s ňou pracovať vonku aj vnútri. Táto stereo kamera je pasívnym zariadením, ktoré funguje na princípe porovnania snímok odfotených dvoma snímačmi len niekoľko centimetrov od seba. Následne sú snímky porovnávané a na základe vzdialenosti pixelov v podobných častiach obrázku je vypočítaná hĺbka a vzdialenosť objektov v scéne. Aby toto všetko korektné fungovalo je nutné na začiatku spraviť veľmi presnú kalibráciu. Kameru je možné vidieť na obrá



Obrázok č. 4: ZED M Stereo kamera

Kameru sme mali v úmysle použiť primárne na skenovanie celej miestnosti, ale ukázalo sa to ako úplne nereálne. Mapovanie prostredia pomocou kamery nefungovalo správne aj napriek tomu, že aktualizácie vychádzali prakticky na dennej báze, ktoré síce optimalizovali výstup, no nie v dostatočnej miere na to aby sme ho dokázali použiť v projekte. V situáciách keď kamera smerovala na svetlé plochy alebo plochy odrážajúce svetlo nedokázala presne určiť svoju polohu v priestore a po nasnímaní tmavších plôch miestnosti začala mapovať v úplne iných súradniciach, čo malo negatívny dopad na výsledný model. Taktiež vývojové prostredie, ktoré StereoLabs poskytuje nie je ešte v dostatočnej miere použiteľné. Kamera nám nedala dostatočne kvalitný model ani pri spojení so spomínaným RTAB-MAP, pričom dodatočná úprava modelu by bola veľmi zdĺhavá. Do budúcnosti je to však zaujímavá technológia, ktorej vývoj sa určite oplatí sledovať.

10.3.10 Samsung Gear 360

Samsung Gear 360 kamera ukázaná na obrázku č. 5 je technológia ktorá vytvára 360° fotografie s 8MP fotoaparátom. Sprievodná aplikácia je kompatibilná iba s mobilnými telefónmi Samsung Galaxy vyššej triedy ako 6, no s novou verziou v roku 2017 pribudla aj kompatibilita s mobilnými telefónmi značky Apple.



Obrázok č. 5: Samsung Gear 360

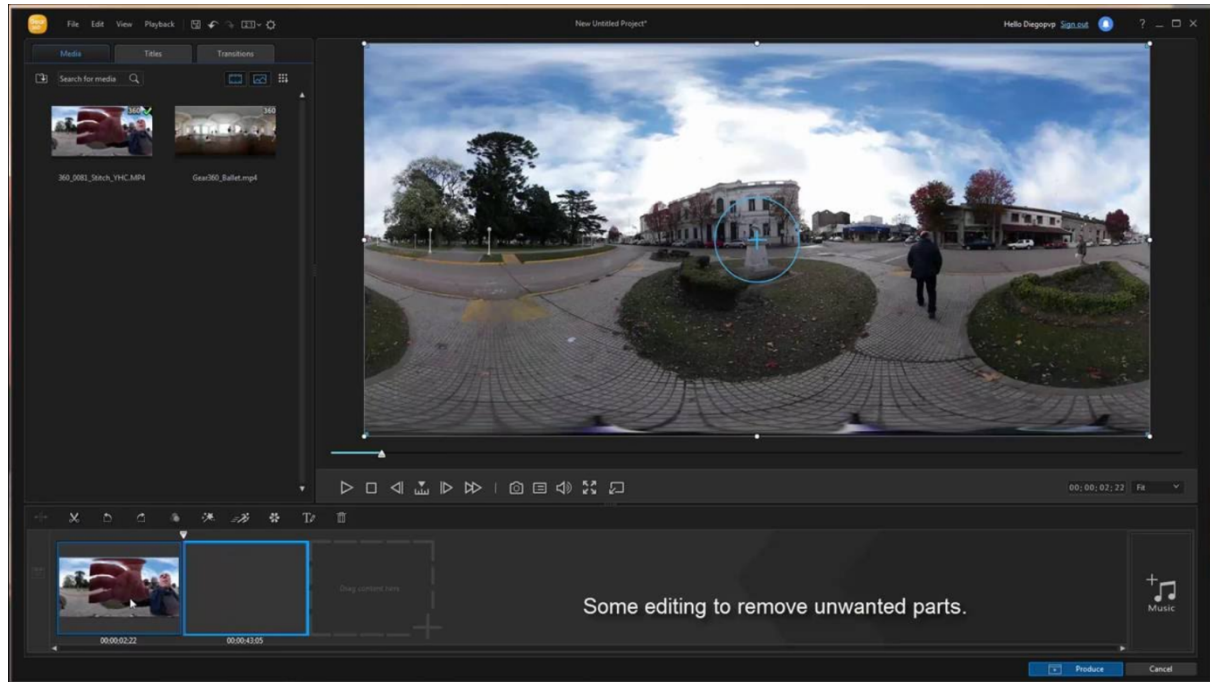
Fotoaparát sa spája s mobilným telefónom prostredníctvom pripojenia Wi-Fi. Aplikácia vám umožňuje jednoducho prepínať medzi režimami videozáznamov, fotografií, časových pásiem, videosekvencií a režimu HDR na šírku. Následne je možná kontrola nad podrobnejšími nastaveniami, ako je kompenzácia expozície a vyváženie bielej farby.

Veľkou výhodou je, že zariadenie máme k dispozícii v samotnom 3D Labe, čo nám umožňuje ho využiť na prototypu Panorama, ktorý je tvorený z týchto fotografií (Príklad fotografie je ukázaný na obrázku č. 6).



Obrázok č. 6: Fotografia časti 3D Labu vyhotovená pomocou kamery Samsung Gear 360

Na spracovanie výstupov zo zariadenia je navrhnutý špeciálny softvér Gear 360 ActionDirector, ktorý je ukázaný na obrázku č. 7. Softvér poskytuje niekoľko základných funkcií úprav, ako je orezanie, pridávanie titulov a vytváranie prechodov vo videách a fotografiách. Po spracovaní fotografií je zobrazený kvalitný 3D priestor a vo formáte, ktorý sa dá ľahko premietnuť vo virtuálnej realite pomocou HTC Vive.



Obrázok č. 7: Ukážka softvéru Gear ActionDirector

10.4 Multiplatformovosť

Významnou výhodou nášho projektu je jeho multiplatformovosť. Technológie WebGL a webVR nám umožňujú zobrazovať 3D realitu na všetkých zariadeniach, ktoré tieto technológie podporujú. Jedná sa o širokú škálu zariadení. V nasledujúcich kategóriách si opíšeme jednotlivé technológie.

10.4.1 WebGL

Technológia WebGL nám zabezpečuje bezplatný štandard vykresľovania 3D grafiky na HTML5 plátne (canvas). Najnovšia verzia WebGL 2.0 ponúka podobné API rozhranie ako OpenGL ES 3.0. Najväčšou výhodou tejto technológie je jej multiplatformovosť a podpora od bežných PC po mobilné telefóny (smartphony) rôznych výrobcov. Nižšie môžeme vidieť webové prehliadače, ktoré v súčasnosti (November 2017) podporujú WebGL technológiu.

Webové prehliadače podporujúce WebGL:

- Google Chrome – verzia 56+

- Mozilla Firefox – verzia 51+
- Midori
- Safari – podpora WebGL 1.0
- Opera + verzia 43+
- Internet Explorer – WebGL 1.0 čiastočne podporované od verzie 11
- Microsoft Edge – WebGL 1.0 podporuje od verzie 10240. Podpora WebGL 2.0 je plánovaná.
- Vivaldi

Mobilné prehliadače:

- BlackBerry 10 – WebGL 1.0 od verzie 10.0
- BlackBerry PlayBook – WebGL 1.0 od verzie 2.00
- Android Prehliadač – WebGL 1.0 v základe nepodporované, ale niektoré zariadená ho podporujú po nainštalovaní novšieho formware. Na väčšine zariadení je sprístupnený pomocou prehliadača Google Chrome
- Internet Explorer – WebGL 1.0 je podporované na Windows Phone 8.x (11+)
- Firefox for mobile – WebGL 1.0 pre Android a MeeGo zariadenia od verzie 4
- Firefox OS
- Google Chrome – WebGL 1.0 od verzie 25
- MeeGo – WebGL 1.0 v základe nepodporované, ale dostupne cez Firefox.
- Microsoft Edge – WebGL 1.0 je dostupne on Windows 10 Mobile
- Opera Mobile – Opera Mobile 12 supports WebGL 1.0 (iba na Android zariadeniach)
- Sailfish OS – WebGL 1.0 je podporované
- Tizen – WebGL 1.0 podporované
- Ubuntu Touch
- WebOS
- iOS – WebGL 1.0 je dostupné pre mobile Safari

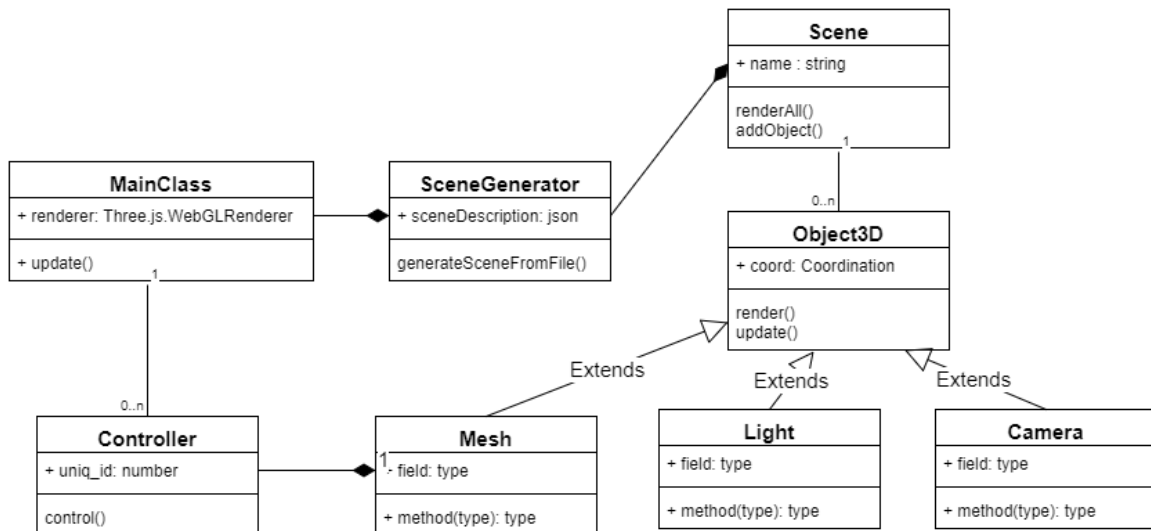
10.4.2 WebVR

Táto technológia nám umožňuje vytvárať aplikácie pre virtuálnu realitu pomocou webových technológií HTML5 a javascript. Vďaka webVR je možné vytvárať jednu aplikáciu pre virtuálnu realitu, ktorá bude fungovať na rôznych druhoch VR headsetov (náhlavných sústav pre Virtualnu realitu). Vďaka tejto aplikácii môžeme predpokladať širšie publikum pre našu aplikáciu.

11 Návrh

V tejto kapitole rozoberieme podrobný návrh architektúry vyvíjanej aplikácie a taktiež návrh vizualizácie.

11.1 Diagram tried



Obrázok č. 8: Diagram tried

11.1.1 Opis tried

V tejto podkapitole sú opísané triedy, ktoré sa nachádzajú v diagrame tried na obrázku č. 8.

MainClass

Hlavná trieda inicializuje všetky potrebné objekty na vytvorenie 3D scény pre virtuálnu realitu. Taktiež zodpovedná za spúšťanie hlavného cyklu vykresľovania a aktualizovania scény.

Controller

Trieda je určená na spracovanie informácií z pohybových ovládačov, akými sú aj informácie o stlačených tlačidlách alebo informácie o polohe ovládača v priestore. Na základe týchto informácií sa vykonávajú funkcie interakcie s ostatnými objektami v scéne.

SceneGenerator

Trieda načítava z predurčeného súboru štruktúru scény (objekty a ich parametre) a vytvorí komplexnú scénu potrebnú na vykreslenie 3D priestoru.

Mesh

Trieda dedí od triedy Object3D základne vlastnosti o objekte a pridáva ďalšie informácie na zobrazenie komplexných objektov skladajúcich sa z mnohých polygónov.

Object3D

Základná trieda pre všetky objekty obsahujúca potrebné vlastnosti pre zobrazenie v 3D priestore (napr.: x,y,z súradnice, rotáciu).

Scene

Trieda obsahujúca všetky potrebné objekty (3D objekty, kameru a svetlá).

Light

Trieda dedí od triedy Object3D základne vlastnosti a pridáva vlastnosti potrebné na opis svetla v priestore.

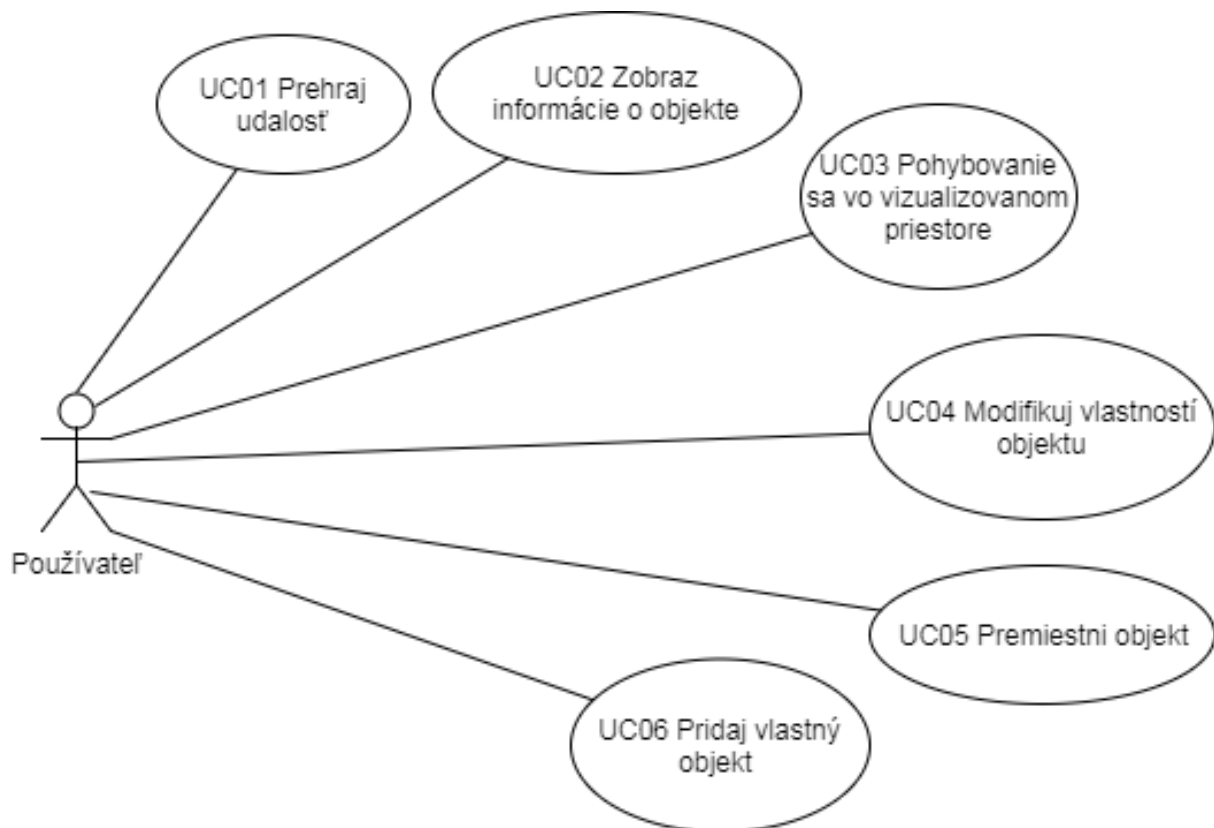
Camera

Trieda dedí od triedy Object3D základne vlastnosti a pridáva vlastnosti potrebné na opis kamery v scéne. Kamera je dôležitý prvok ovplyvňujúci výsledný vzhľad prostredia vo virtuálnej realite.

11.2 Prípady použitia

Prípady použitia nám definujú funkcionálne požiadavky aplikácie.

Diagram prípadov použitia zobrazuje závislosti medzi doteraz identifikovanými prípadmi použitia. Na obrázku č. 9 je diagram prípadov použitia. Používateľ môže jednoducho a priamočiaro používať systém.



Obrázok č. 9: Diagram prípadov použitia

Jednotlivé prípady použitia sú spísané formou tabuliek pre jednoduché čítanie a orientáciu sa. Doteraz sme z požiadaviek zákazníkov identifikovali nasledovné prípady použitia.

UC01 Prehraj udalosť

Prípado použitia	
ID:	01
Názov:	Prehraj udalosť
Popis:	<ul style="list-style-type: none"> - Po kliknutí na objekt sa používateľ ocitne v priestore, v ktorom sa môže otáčať a pozeráť tak na udalosti, ktoré sa dejú vôkol neho v 360° videu. - Používateľovi sa spustí hudba po kliknutí na objekt.
Používateľ:	Používateľ
Predpoklady:	Používateľ je vo vizualizovanom priestore a má k dispozícii HMD.
Dôsledky:	Používateľ sa ocitne v prehrávanom deji.

Základný tok:		
Číslo kroku	Používateľ	System
1	Používateľ klikne na objekt, ktorý umožní prehrať video	System zobrazí používateľa v strede prehrávanej udalosti.
2	Používateľ kliknutím na "ukončenie prehrávania" ukončí prehrávanie	System skončí prehrávanie videa a zobrazí používateľa na pôvodnej pozícii

UC02 Zobrazenie informácie o objekte

Prípád použitia	
ID:	02
Názov:	Zobrazenie informácie o objekte
Popis:	Používateľovi sa po kliknutí na objekt zobrazia podrobnosti o danom objekte v podobe textu alebo obrazu.
Používateľ:	Používateľ
Predpoklady:	Používateľ je vo vizualizovanom priestore a má k dispozícii HMD.
Dôsledky:	Používateľovi sa zobrazí vo vizualizovanom priestore okno s informáciami o objekte

Základný tok:		
Číslo kroku	Používateľ	Systém
1	Používateľ namieri na objekt	Systém zobrazí okno s informáciami o objekte
2	Používateľ namieri mimo daný objekt	Systém skryje okno s informáciami o objekte
Alternatívny tok:		
Číslo kroku	Používateľ	Systém
1	Používateľ namieri na objekt, ktorý neumožňuje zobrazenie popisu.	Systém nezobrazí nič

UC03 Pohybovanie sa vo vizualizovanom priestore

Prípadoúžitia	
ID:	03
Názov:	Pohybovanie sa vo vizualizovanom priestore
Popis:	Používateľ sa vo vizualizovanom priestore môže presúvať pomocou takzvaného teleportovania. Kliknutím na plochu, ktorá umožňuje presunutie sa používateľ objaví na danej pozícii.
Používateľ:	Používateľ
Predpoklady:	Používateľ je vo vizualizovanom priestore a nie je v móde prehrávania udalosti a má k dispozícii HMD.
Dôsledky:	Používateľ sa presunie vo vizualizovanom priestore na iné miesto.

Základný tok:		
Číslo kroku	Používateľ	System
1	Používateľ namieri na podlahu	System zvýrazní plochu
2	Používateľ sa kliknutím presunie	System zmení pozíciu používateľa tak aby sa zjavil na zvolenej pozícii

UC04 Modifikuj vlastností objektu

Prípad použitia	
ID:	04
Názov:	Modifikuj vlastnosti objektu
Popis:	Používateľ vo vizualizovanom priestore môže modifikovať vlastnosti objektu. Vybraním objektu môže meniť pomocou spôsobu na to určeného vlastnosti jednotlivých objektov ako sú napríklad rozmery a začiatkové umiestnenie.
Používateľ:	Používateľ
Predpoklady:	Používateľ je vo vizualizovanom priestore a nie je v móde prehrávania udalosti a má k dispozícii HMD.
Dôsledky:	Objekt ostane zmodifikovaný

Základný tok:		
Číslo kroku	Používateľ	Systém
1	Používateľ namieri na objekt	Systém zvýrazní objekt
2	Používateľ kliknutím zvolí možnosť modifikovať objekt	Systém umožní používateľovi modifikovať vlastností objektu
3	Používateľ modifikuje dáta objektu a uloží zmeny	Systém nevratne uloží vykonané zmeny a zobrazí používateľa na pôvodnej pozícii vo vizualizovanom priestore

Alternatívny tok:		
Číslo kroku	Používateľ	System
3	Používateľ preruší modifikovanie objektu	System zobrazí používateľa na pôvodnej pozícii vo vizualizovanom priestore

UC05 Premiestni objekt

Prípado použitia	
ID:	05
Názov:	Premiestni objekt
Popis:	Používateľ môže vo vizualizovanom priestore presúvať jednotlivé objekty. Kliknutím na objekt, podržaním tlačidla a jednoduchým pohybom ruky premiestni objekt na inú pozíciu.
Používateľ:	Používateľ
Predpoklady:	Používateľ je vo vizualizovanom priestore a nie je v móde prehrávania udalosti a má k dispozícii HMD.
Dôsledky:	Objekt ostane na novej pozícii

Základný tok:		
Číslo kroku	Používateľ	System
1	Používateľ namieri na objekt	System zvýrazní objekt.
2	Používateľ kliknutím vyvolá presúvanie objektu	System umožní používateľovi presúvať objekt.
3	Používateľ pohybom premiestni objekt	System priebežne zobrazuje aktuálnu pozíciu objektu.
4	Používateľ skončí s premiestňovaním objektu	System nevratne uloží novú pozíciu objektu.

UC06 Pridaj vlastný objekt

Prípadoúžitia	
ID:	06
Názov:	Pridaj vlastný objekt
Popis:	Používateľ môže vo vizualizovanom priestore pridávať vlastné objekty. Umožnené je to pomocou upravenia JSON súboru, kde používateľ pridá nový objekt medzi existujúce objekty.
Používateľ:	Používateľ
Predpoklady:	Používateľ je vo vizualizovanom priestore a nie je v móde prehrávania udalosti a má k dispozícii HMD.
Dôsledky:	Nový objekt zostane súčasťou vizualizovaného priestoru

Základný tok:		
Číslo kroku	Používateľ	System
1	Používateľ namieri na objekt	System zvýrazní objekt.
2	Používateľ kliknutím vyvolá presúvanie objektu	System umožní používateľovi presúvať objekt.
3	Používateľ pohybom premiestni objekt	System priebežne zobrazuje aktuálnu pozíciu objektu.
4	Používateľ skončí s premiestňovaním objektu	System nevratne uloží novú pozíciu objektu.

12 Implementácia

Ako bolo zistené v analýze, multiplatformovosť najlepšie dosiahneme ak budeme používať technológiu webVR umožňujúce tvorbu webových aplikácií pre virtuálnu realitu. Základom projektu je podporovaný webový prehliadač a naša webová aplikácia.

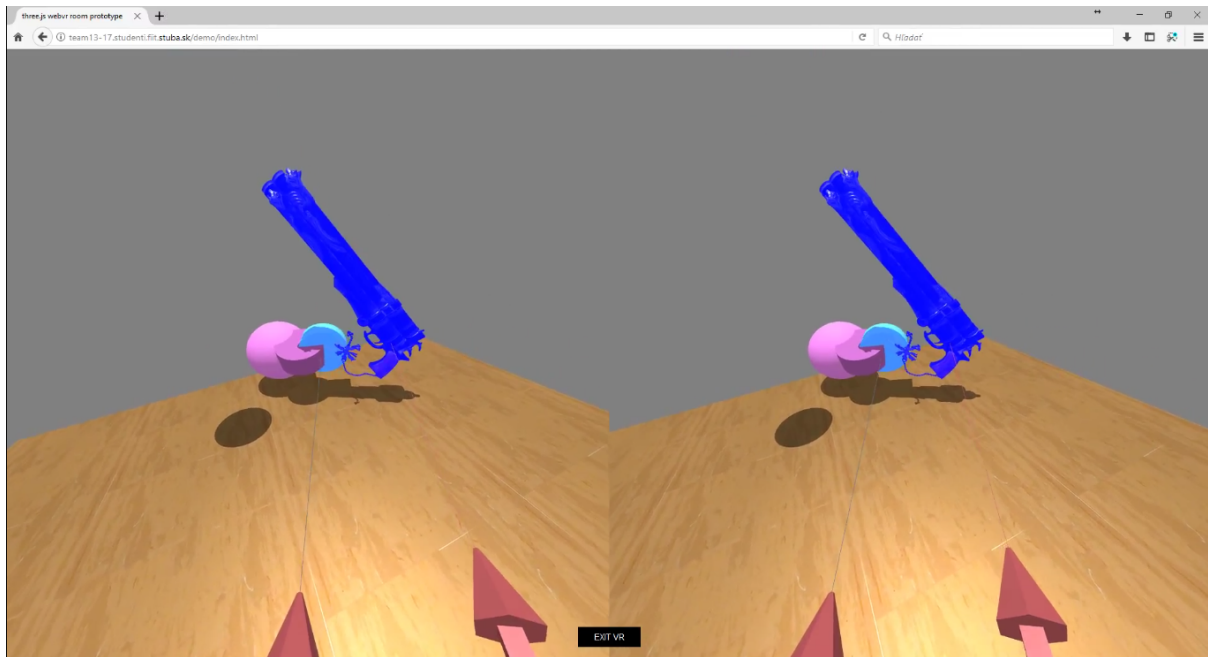
12.1 Prototypy

Počas vývoja projektu boli vytvorené 4 prototypy a finálna verzia. Všetky sú popísané nižšie.

12.1.1 3D Room verzia 1

Prvý prototyp aplikácie bol implementovaný pomocou knižnice Three.js dostupnej na nasledovnej adrese: <https://threejs.org/> Knižnica výrazne pomáha pri tvorbe WebGL 3D scén pomocou jazyka JavaScript. V posledných verziách bola vylepšená podpora pre virtuálnu realitu, čo nám pomohlo pri tvorbe našej prvej verzie aplikácie. Prototyp bol určený hlavne na využitie s virtuálnou realitou HTC Vive.

Implementovali sme základnú podporu pohybových ovládačov a aj pohyb kamery podľa pohybu človeka v reálnom priestore. Ovládače sú v prototypu reprezentované pomocou šípiek so svetelným lúčom. Pomocou lúča vieme vyberať objekty v scéne a pohybovať s nimi podľa pohybu ovládača. Na obrázku č. 10 je znázornená scéna a ovládače nášho prototypu.

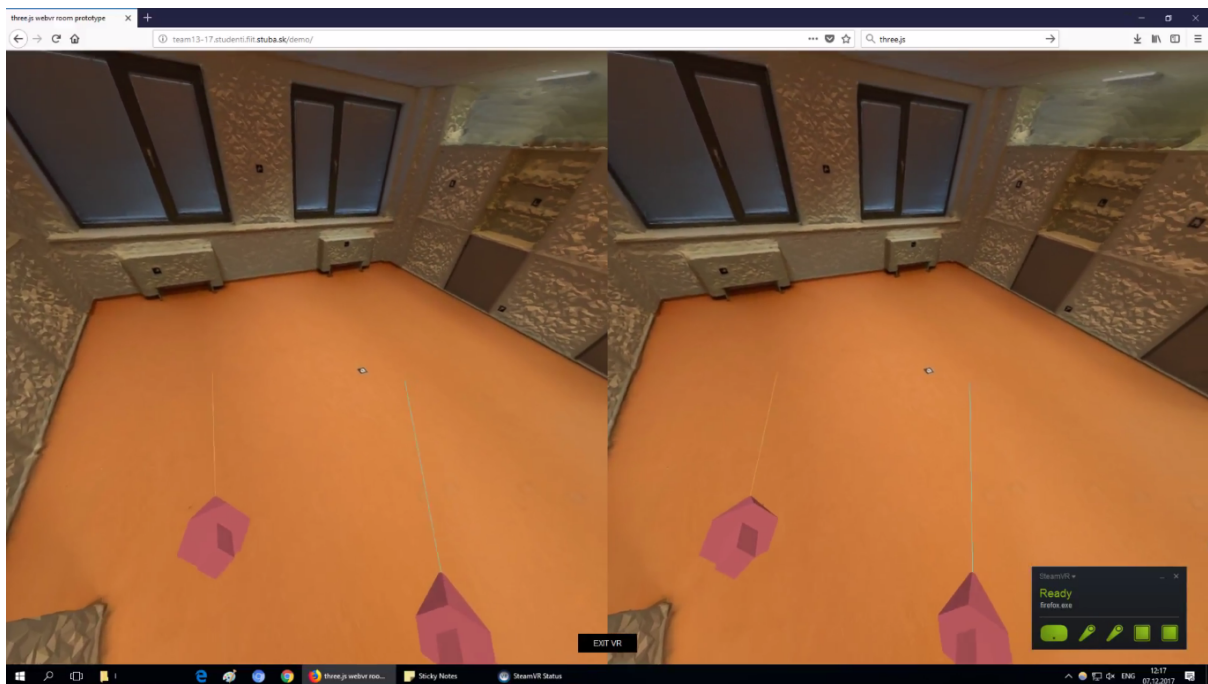


Obrázok č. 10: Scéna a ovládače v prototypu 3D Room

12.1.2 3D Room verzia 2

Druhá verzia prototypu stavia na základoch predchádzajúceho prototypu. Pridaná bola podpora materiálov vo formáte MTL, vďaka čomu sme mohli načítať 3D model (s textúrou) miestnosti, ktorý bol získaný technikou fotogrametrie opísanej v analýze. Načítanie modelu aj keď len prázdnej miestnosti zlepšilo zážitok používateľa a zvýšil sa dojem prítomnosti.

Podobne ako v predchádzajúcej verzii sú pohybové ovládače reprezentované šípkami, ktorými vieme vykonávať interakciu s vybranými objektmi v scéne.



Obrázok č. 11: Ukážka prototypu 3D Room

Do druhej verzie prototypu bola pridaná funkcionálna, ktorá umožňuje používateľovi premiestňovanie sa v 3D priestore. Všetko je implementované pomocou three.js knižnice. Keď používateľ namieri na podlahu a klikne, tak sa presunie na pozíciu, na ktorú namieril. Ide o presúvanie takzvaným "teleportovaním". Implementované to je tak, že sa nájde bod prieniku lúča z ovládača a objektu. Ak prvým prečatým objektom je podlaha, tak sa posunie celá scéna zatiaľ čo pozícia kamery sa nemení. Všetky objekty pridávame do takzvanej grupy. Groupa nám zoskupuje všetky objekty, ktoré sú v scéne. Posunutím pozície grupy sa posunú všetky objekty v scéne, čo nám značne zjednodušuje implementáciu aj škálovateľnosť kódu.

12.1.3 3D Room verzia 3

Tretia a zároveň finálna verzia 3D room je rozšírením predošlého prototypu – verzia 2. Úpravy, ktoré nastali v tejto verzii sú vytvorenie 3D modelu miestnosti a nasadenie do prototypu namiesto modelu vytvorenému pomocou fotogrametrie, nakoľko pôvodný model

nebol dostatočný a po niekoľkých experimentoch sme zistili, že technika fotogrametrie je dostatočná na menšie objekty. Na objekt vo veľkosti miestnosti s malým množstvom textúr nie je technika fotogrametrie vhodná. Ďalšími úpravami v novej verzii prototypu bolo pridávanie, upravovanie a mazanie objektov do miestnosti pomocou vytvoreného JSON súboru. S objektami je možná rôzna interakcia.

Prvou pridanou interakciou bolo pohybovanie s objektami na iné miesta v miestnosti. Ďalšou z interakcií je zobrazenie informácií o objektoch, v podobe textového popisu, obrazu, videa alebo 360° videa.

Pohyb s objektami – Vo vizualizovanom 3D priestore sa nachádzajú administrátorom pridané objekty, ktoré môže používateľ premiestniť podľa potreby a odkryť záhady 3D Labu, v ktorom je projekt vyvíjaný a slúži aj na jeho prezentáciu. Po opätovnom načítaní stránky sa objekty zobrazia v pôvodných koordinátach.

Uchopenie objektu – Aby sme používateľovi zjednodušili manipuláciu s objektami, tak v prípade veľkej vzdialenosti medzi používateľom a objektom má používateľ možnosť si predmet „teleportovať“ do ruky, tak ako je ukázané na obrázku číslo 12 a nemusí sa zbytočne a bezcieľne potulovať po miestnosti. Uchopením je detailné prezeranie objektu jednoduchšie.



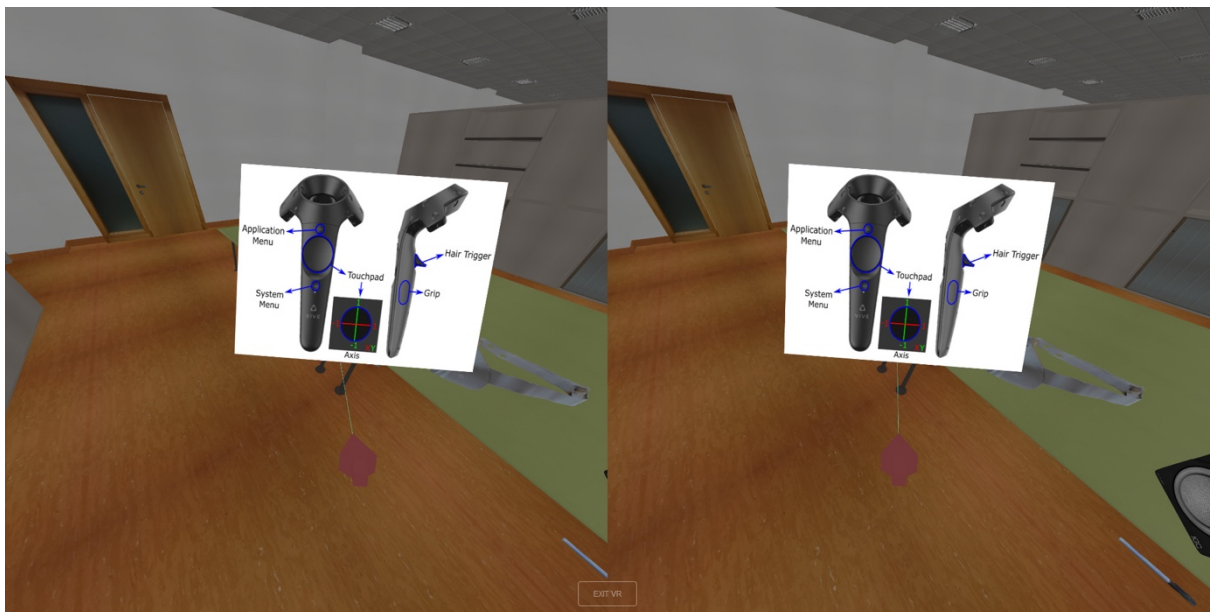
Obázok č. 12: Uchopenie objektu

Textový popis – Administrátorom vybrané objekty ponúkajú používateľovi možnosť textového opisu objektu s detailným technickým popisom, tak, ako je ukázané na obrázku číslo 13. Ak je zobrazený text príliš dlhý, používateľ je schopný rolovať textom nahor alebo nadol podľa potreby.



Obrázok č. 13: Textový popis objektu

Obraz – Administrátorom vybrané objekty ponúkajú používateľovi možnosť zobrazenia obrázku s technickým popisom objektu, tak, ako je ukázané na obrázku číslo 14.



Obrázok č. 14: Obrázový opis objektu

Video – V 3D labe sa uskutočnilo mnoho experimentov, ktoré môžu byť nahrávané na video. Po kliknutí na vybrané objekty si môže používateľ tieto videá prehrať a získať lepší prehľad o tom, čo sa dialo počas experimentu. Napríklad testovanie projektov vyvíjaných v 3D Labe.

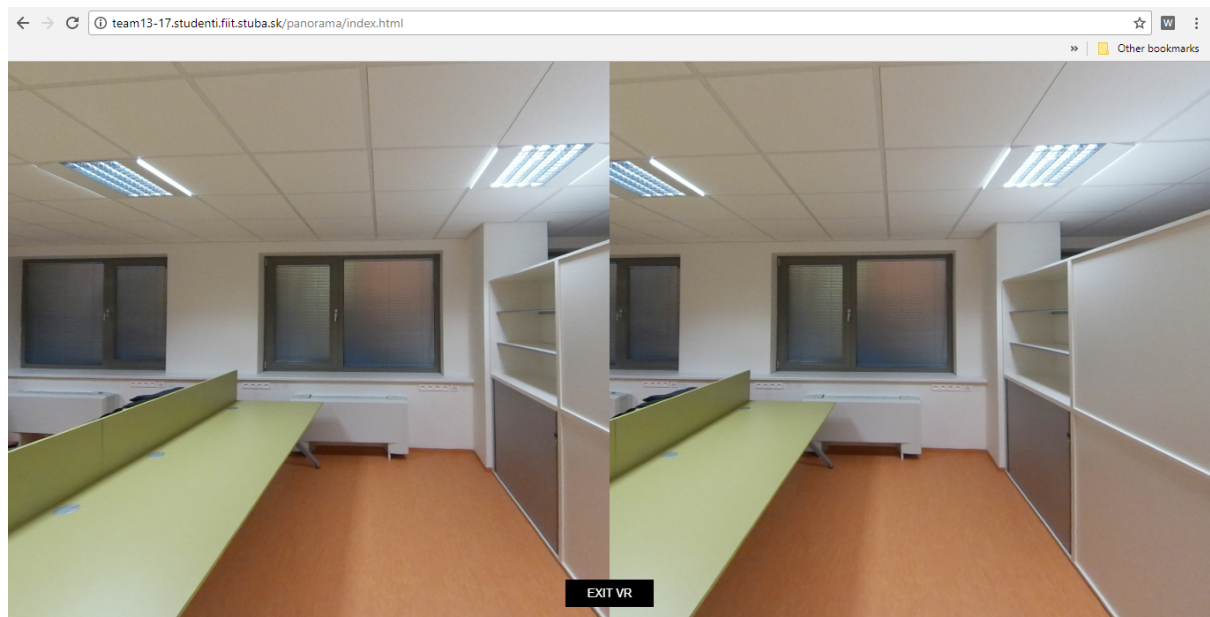
360° video – Aby sa používateľ vedel lepšie ponoriť do prebiehajúceho experimentu, tak niektoré objekty ponúkajú možnosť prehratia 360° videa.

Úpravy ďalej pozostávajú z pridania kolízií. Kolízie slúžia na navodenie reálneho pocitu vo vizualizovanom prostredí. Kvôli pocitu reálnosti je implementovaná v produkte aj gravitačná sila. Kolízia objektu s podlahou, kusom nábytku alebo inými objektami vloženými do 3D priestoru spôsobí zastavenie pohybu objektu. Gravitačná sila „ťahá“ objekty smerom nadol a zastavia sa pomocou už spomínaných kolízií.

Pridanou funkcionalitou v tomto prototypu je taktiež aj možnosť upravovania priestoru pre administrátora stránky a nastavovanie výšky kamery podľa výšky používateľa.

12.1.4 Panorama verzia1

Druhý prototyp aplikácie bol implementovaný v čistom java skripte využitím knižnice Three.js čo umožňuje veľmi dobrý výkon a teda realistický dojem. Implementovali sme 360° panoramu. používateľ sa nachádza v 3D sfére, ktorá je z vnútornej strany otexturovaná jednou alebo viacerými fotkami či videom. Používateľ sa v prototypu môže otáčať. Pohybovanie v tomto prototypu nie je umožnené. Na obrázku č. 15 je ukážka z prototypu.



Obrázok č. 15: Ukážka z prototypu Panorama

12.1.5 Panorama verzia2

Druhá verzia panorámy pridáva nové funkcionality do predchádzajúcej verzie. Do prototypu bola pridaná podpora ovládania pomocou kurzora myši a teda možnosť ovládať aplikáciu bez VR zariadenia.

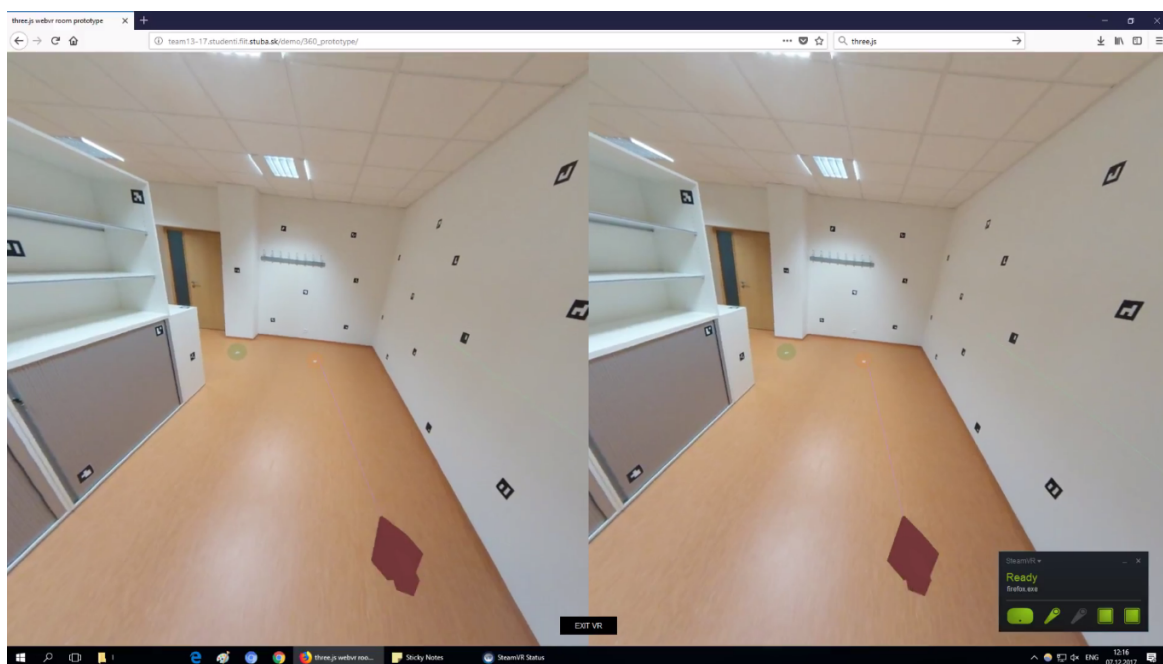
Aplikácia podporuje viacero 360 stupňových panorám a je možné sa medzi nimi prepínať. Je len potrebné nasmerovať ovládač alebo kurzor myši na zelenú bodku (smerník) v priestore a stlačiť hlavné tlačidlo. Textúra 3D sféry sa zmení na vybranú ďalšiu panorámu.

```
panoramaStructure.json 1.29 KB
[
  {
    "panoramaId": 0,
    "textureName": "360_0029_Stitch_XHC.JPG",
    "waypoints": [
      {
        "x": -23.19,
        "y": -338.94,
        "z": 365.84,
        "next": 1
      },
      {
        "x": -29,
        "y": -239,
        "z": 436,
        "next": 2
      }
    ]
  },
  {
    "panoramaId": 1,
    "textureName": "360_0028_Stitch_XHC.JPG",
    "waypoints": [

```

Obrázok č. 16: Ukážka súboru typu JSON

Tvorba sústavy panorám je implementovaná pomocou súboru typu JSON, ktorý špecifikuje množstvo panorám, ich textúry a aj umiestnenie zelených bodiek (smerník). Ukážka štruktúry súboru je na obrázku č. 16. Ukážka aplikácie s použitím VR zariadenia je na obrázku č. 17.



Obrázok č. 17: Ukážka prototypu Panorama verzia2

13 Testovanie

Testovanie sme vykonávali výhradne pomocou nižšie spísaných akceptačných testov, ktoré sú vo forme tabuliek. V testoch sme sa snažili čo najvernejšie nasimulovať správanie bežného používateľa. Na teste sa podieľajú vždy minimálne dvaja členovia tímu, autor (osoba, ktorá sa aktívna podieľala na vývoji) a tester/i (osoba/y nezasahujúca/e do vývoja testovanej časti). Testy boli vykonávané výhradne v priestoroch 3D Labu s HMD. Toto testovanie sa však časom ukázalo ako neefektívne, nakoľko bolo časovo náročné. Metodiku testovania sme zmenili a momentálne si autor vykoná testovanie najprv vo svojom webovom prehliadači, čím sa mu podarí odhaliť základné nedostatky. Následne sú vykonané dôkladnejšie testy v 3D Labe jedným alebo viacerými testerami.

13.1 Testovanie prototypu zo zimného semestra

Testovací scenár prehrávania udalosti	
Test Case ID:	01
Use Case:	UC05
Programátor:	Meno programátora:
Tester:	Meno testera:
Vstupné požiadavky:	Používateľ je už vo vizualizovanom priestore Používateľ má k dispozícii ovládač HTC VIVE. Používateľ si spustí demo z /tests/test1.html
Výstupné požiadavky:	Objekt ostane na novej pozícii

číslo	Akcie používateľa	Očakávaný výstup systému	Koment
1.	Namierte na pištoľ	Pištoľ sa zvýrazní	
2.	Podržte tlačítko a urobte niekoľko ľubovoľných pohybov	Systém bude podľa pohybov ruky meniť pozíciu pištole	
3.	Pouštíte tlačítko	Systém preruší presúvanie pištole	

Testovací scenár prehrávania udalosti	
Test Case ID:	02
Use Case:	UC01

Programátor:	Meno programátora:
Tester:	Meno testera:
Vstupné požiadavky:	Používateľ je už vo vizualizovanom priestore Používateľ má k dispozícii ovládač HTC VIVE. Používateľ si spustí demo z /tests/test2.html
Výstupné požiadavky:	Používateľ sa ocitne uprostred prehrávaného deja, v ktorom sa môže obzerať vôkol seba.

číslo	Akcie používateľa	Očakávaný výstup systému	Koment
4.	Kliknite na zelenú guľu	Zobrazí sa nový vizualizovaný priestor	
5.	Otočte sa okolo svojej osi a pozrite sa hore a dole.	Systém zobrazuje plynule prehrávaný dej v celom priestore.	
6.	Používateľ klikne na červenú guľu	Systém preruší prehrávanie deja a zobrazí používateľa na pôvodnej pozícii	

Testovací scenár zobrazenia podrobností o objekte	
Test Case ID:	03
Use Case:	UC02
Programátor:	Meno programátora:
Tester:	Meno testera:
Vstupné požiadavky:	Používateľ je už vo vizualizovanom priestore Používateľ má k dispozícii ovládač HTC VIVE. Používateľ si spustí demo z: /tests/test3.html
Výstupné požiadavky:	Zobrazenie popisu objektu a posúvanie textu

číslo	Akcie používateľa	Očakávaný výstup systému	Koment
1.	Namierte na pištoľ a kliknite trigger tlačítko	Vo vizualizovanom priestore sa zobrazí okno s popisom	
2.	Držte palec na vrchnej časti thumbpad-u	Text sa posúva smerom hore	
3.	Držte palec na spodnej časti thumbpad-u	Text sa posúva smerom dole	
4.	Podržte palec raz na ľavej a potom na pravej časti thumbpad-u	Nenastanú žiadne zmeny	
5.	Kliknite trigger tlačítko	Text zmizne	

Testovací scenár pohybovania sa vo vizualizovanom priestore	
Test Case ID:	04
Use Case:	UC03
Programátor:	Meno programátora:
Tester:	Meno testera:
Vstupné požiadavky:	Používateľ je už vo vizualizovanom priestore Používateľ má k dispozícii ovládač HTC VIVE. Používateľ si spustí demo z /tests/test4.html
Výstupné požiadavky:	Premiestnenie používateľa na zvolenú pozíciu.

číslo	Akcie používateľa	Očakávaný výstup systému	Koment
1.	Namierte na podlahu	Systém nič nezobrazí	
2.	Namierte na podlahu a stlačte trigger tlačítko	Systém premiestni používateľa na namierenú pozíciu	

Testovací scenár vzájomnej kolízie medzi objektami	
Test Case ID:	05
Use Case:	UC05
Programátor:	Meno programátora:
Tester:	Meno testera:
Vstupné požiadavky:	Používateľ je už vo vizualizovanom priestore Používateľ má k dispozícii ovládač HTC VIVE. Používateľ si spustí demo z /tests/test5.html
Výstupné požiadavky:	Používateľ nedokáže presúvať objekty cez seba.

číslo	Akcie používateľa	Očakávaný výstup systému	Koment
1.	Namierte na 360° kameru	Systém nič nezobrazí	
2.	Podržte trigger tlačítko a presúvajte kameru do fialovej kocky zo všetkých strán možných strán	Systém premiestni kameru iba po okraj kocky. Systém nedovolí vnoriť kameru dnu do kocky kvôli kolíziám	

Testovací scenár prehrávania udalosti	
Test Case ID:	06
Use Case:	UC01
Programátor:	Meno programátora:
Tester:	Meno testera:
Vstupné požiadavky:	Používateľ je už vo vizualizovanom priestore Používateľ má k dispozícii ovládač HTC VIVE. Používateľ si spustí demo z /tests/test6.html
Výstupné požiadavky:	Používateľovi sa prehrá zvuk a krátke video

číslo	Akcie používateľa	Očakávaný výstup systému	Koment
1.	Namierte na kameru	Systém nič nezobrazí	
2.	Stlačte trigger tlačítko	Systém spustí 360° video a prehráva sa aj zvuk	

13.2 Testovanie finálnej aplikácie

Uchopenie predmetu

Základné údaje	
Test Case ID	1
Use Case ID	UC05
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Používateľ presunie objekt na novú pozíciu Konečná pozícia zostane zachovaná

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na objekt	Objekt zmení farbu	Farba je príliš výrazná
2.	Používateľ stlačí hlavné tlačidlo a vykoná niekoľko ľubovoľných pohybov rúk	Objekt bude meniť svoju polohu v priestore podľa pohybov rúk	---

Zobrazenie informácií o objekte

Základné údaje	
Test Case ID	2
Use Case ID	UC02
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Používateľ budú vo vizualizovanom prostredí zobrazené informácie o objekte

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na okuliare pre virtuálnu realitu	Okuliare zmenia farbu	Opäť príliš výrazná farba objektu
2.	Používateľ stlačí sekundárne tlačidlo	Používateľovi sa zobrazí text s informáciami o Google Daydream	Pozadie textu mi príde veľmi priesvitné. Taktiež by som zvolil inú formu vykreslenia

Zobrazenie obrázku

Základné údaje	
Test Case ID	3
Use Case ID	UC01
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Používateľ uvidí informácie detailnejšie informácie o objekte v podobe obrázku

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na ovládač HTC VIVE	Ovládač HTC VIVE zmení farbu	Opäť farba
2.	Používateľ stlačí sekundárne tlačidlo	Používateľovi sa zobrazí obrázok popisujúci ovládač HTC VIVE	Obrázok mi príde trochu rozmazaný. Zaobliil by som rohy, vyzeralo by to podľa mňa lepšie

Prehratie 360° videa

Základné údaje	
Test Case ID	4
Use Case ID	UC01
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Používateľ sa ocitne uprostred 360° videa, kde môže pozorovať priebeh experimentu

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na model kamery Samsung Gear 360	Model kamery zmení farbu	Farba
2.	Používateľ stlačí sekundárne tlačidlo	Používateľovi sa začne prehrávať 360° video	---

Gravitácia pôsobiaca na objekty

Základné údaje	
Test Case ID	5
Use Case ID	Nemá Use Case
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Objekty vo vizualizovanom priestore budú voľne klesať k zemi. Používateľ nadobudne pocit, že na objekty pôsobí gravitácia

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na ľubovoľný objekt	Označený objekt zmení farbu	---
2.	Používateľ stlačí primárne tlačidlo a presunie objekt do ľubovoľnej nenulovej výšky	Objekt zmení svoju polohu podľa vykonaného pohybu	---
3.	Používateľ pustí primárne tlačidlo	Používateľ vidí ako objekt pomaly klesá dole	Gravitácia nie je úplne reálna. Rýchlosť pádu objektu je veľmi pomalá

Kolízie objektov

Základné údaje	
Test Case ID	6
Use Case ID	Nemá Use Case
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Objekty nebudú môcť prechádzať cez seba.

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na ľubovoľný objekt	Označený objekt zmení farbu	Farba zvýrazneného objektu
2.	Používateľ stlačí primárne tlačidlo a pokúsi sa presunúť objekt cez stôl	Objekt bude meniť svoju polohu podľa vykonávaného pohybu do momentu pokiaľ nekoliduje so stolom. Vtedy sa pohyb zastaví.	Pád objektu je veľmi pomalý

Zmena výšky vo vizualizovanej miestnosti

Základné údaje	
Test Case ID	9
Use Case ID	UC03
Vstupné požiadavky	Používateľ má k dispozícii webový prehliadač
Výstupné požiadavky	Vizualizovaný priestor bude mať inú výšku voči výške, ktorá bola nastavená na začiatku testovania

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ stlačí terciárne tlačidlo na ovládači HTC VIVE	Používateľovi sa zobrazí nápis „Height“	---
2.	Používateľ bude pomocou šípok hore a dole na ovládači upravovať svoju výšku	Používateľ bude pozorovať zmenu výšky vo vizualizovanom priestore	Nastavovanie výšky je „sekavé“

Transportovanie objektu priamo do „ruky“

Základné údaje	
Test Case ID	10
Use Case ID	UC05
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Zvolený objekt bude presunutý priamo ku ovládaču vo vizualizovanom priestore

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na ľubovoľný objekt	Označený objekt zmení farbu	---
2.	Používateľ stlačí na ovládači primárne tlačidlo	Objekt bude presunutý bližšie k používateľovi	---

Presun v priestore pomocou teleportovania

Základné údaje	
Test Case ID	11
Use Case ID	UC03
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Používateľ sa bude nachádzať na inom mieste vo vizualizovanom priestore

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na ľubovoľné miesto na podlahe	---	---
2.	Používateľ stlačí dotykovú plochu na ovládači	Používateľ bude premiestnený na vybranú pozíciu v rámci virtuálneho priestoru	---

Prehrávanie hudby

Základné údaje	
Test Case ID	12
Use Case ID	UC01
Vstupné požiadavky	Používateľ je vizualizovanom priestore Používateľ má k dispozícii slúchadla Používateľ má k dispozícii ovládače pre HTC VIVE
Výstupné požiadavky	Používateľ bude vo vizualizovanom priestore počuť hudbu

Priebeh testu:

Číslo	Akcia používateľa	Očakávaný výstup systému	Koment
1.	Používateľ namieri laser na rádio	Rádio zmení farbu	---
2.	Používateľ stlačí sekundárne tlačidlo na ovládači	Používateľ bude počuť hudbu	Zvolil by som iný druh hudby

Príloha A – Metodiky

Príloha B – Inštalačná príručka

Príloha C – Používateľská príručka