

**SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE**

Fakulta informatiky a informačných technológií

**VIZUALIZÁCIA INFORMÁCIÍ V OBOHATENEJ REALITE**

Dokumentácia k riadení projektu

Členovia tímu č.4:

BC. MATÚŠ CIMMERMAN

BC. IRINA DYOMINA

BC. MICHAL FAŠÁNEK

BC. JAROSLAV GAZDÍK

BC. DENIS ILLÉS

BC. FILIP JURČACKO

BC. DALIBOR MÉSZÁROS

Predmet:

Tímový projekt I

Vedúci:

Ing. Peter Kapec, PhD.

Akademický rok:

2015/2016

# Obsah

Úvod .....	3
<b>1 Členovia tímu .....</b>	<b>3</b>
1.1 Rozdelenie manažérskych úloh .....	5
1.2 Podiel práce .....	5
<b>2 Aplikácie manažmentov .....</b>	<b>3</b>
2.1 Manažment dokumentácie .....	3
2.2 Manažment podpory vývoja .....	4
2.3 Manažment testovania .....	4
2.4 Manažment prehliadky kódu .....	5
2.5 Manažment rizík .....	5
2.6 Manažment plánovania úlohy .....	5
2.7 Manažment vývoja .....	6
<b>3 Sumarizácia šprintov .....</b>	<b>6</b>
3.1 Šprint 1 .....	6
3.2 Šprint 2 .....	7
3.3 Šprint 3 .....	7
3.4 Šprint 4 .....	8
3.5 Šprint 5 .....	8
3.6 Šprint 6 .....	8
3.7 Šprint 7 .....	9
3.8 Šprint 8 .....	9
3.9 Šprint 9 .....	9
3.10 Šprint 10 .....	9
<b>4 Používané metodiky .....</b>	<b>9</b>
4.1 Metodika písania a komentovania zdrojového kódu .....	9
4.2 Metodika vytvárania úloh v nástroji na evidenciu úloh .....	10
4.3 Metodika komunikácie .....	10
4.4 Metodika manažmentu verzií .....	11
4.5 Metodika testovania .....	11
4.6 Metodika prehliadky kódu .....	11
4.7 Metodika pre logovanie .....	11

<b>5</b>	<b>Globálna retrospektíva .....</b>	<b>11</b>
<b>6</b>	<b>Zoznam kompetencií tímu .....</b>	<b>13</b>
	<b>Čo sa nám podarilo počas šprintu .....</b>	<b>81</b>
	<b>Čo nešlo podľa našich predstáv .....</b>	<b>81</b>
	<b>Čo môžeme zmeniť a zlepšiť do budúceho šprintu.....</b>	<b>81</b>

# Úvod

Tento dokument slúži ako dokumentácia k predmetu Tímový projekt na Fakulte informatiky a informačných technológií. Opisuje prácu na projekte Vizualizácia informácií v obohatenej realite a to hlavne z pohľadu riadenia a manažmentu v tíme.

V kapitole 1 sú opísaní členovia tímu, ich role a podiel práce na dokumentácií. Kapitola 2 detailne opisuje aplikované manažmenty pre riadenie v našom tíme. Sumár doterajších šprintov je opísaný v kapitole 3. Metodiky pre podporu riadenia a manažmentu sú stručne popísané v kapitole 4, pričom ich celé znenie je priložené v prílohe A.

V prílohe B zápisnice a v prílohe C exporty z nástroja pre manažment tímu. V prílohe D sa nachádzajú retrospektívy z doterajších šprintov.

## 1 Členovia tímu

BC. MATÚŠ CIMMERMAN

- absolvent študijného odboru informatika na bakalárskom stupni na FIIT. Aktuálne študent programu informačné systémy na inžinierskom stupni. Najviac programuje v jazyku Java a Scala. Pracuje v Orange Slovensko ako IT Trainee v OSS & BI engineering tíme. Vo voľnom čase sa venuje vývoju pre platformu Android a projektom, ktoré používajú RaspberryPi

BC. IRINA DYOMINA

- absolventka študijného odboru aplikovaná informatika na Paneurópskej vysokej škole. V súčasnosti študuje softvérové inžinierstvo na FIIT STU a pracuje v Assec Central Europe ako IT Tester. Preferuje programovanie v Jave, vo voľnom čase sa venuje web programovaniu a počítačovej grafike.

BC. MICHAL FAŠÁNEK

- študent informačných systémov, preferujúci vývoj v Java. V súčasnosti pracuje ako technická podpora v Hewlett Packard a webový administrátor - junior pre Fabrici consulting. Vo voľnom čase záujmovo študuje oblasť umelej inteligencie.

#### BC. JAROSLAV GAZDÍK

- študent informačných systémov na FIIT STU. Preferuje programovací jazyk Java. Momentálne pracuje ako Java Junior Developer v Atose, odkiaľ má skusenosti s tvorbou webových stránok a testovaním aplikácií pomocou JUnit testov. Vo voľnom čase sa venuje oblasti počítačového videnia a počítačovej grafiky.

#### BC. DENIS ILLÉS

- študent informačných systémov, ktorá na bakalárskom štúdiu programoval prevažne v jazyku C rozšíreného do knižnicu CUDA. Preferuje programovanie v Jave. Vo voľnom čase sa venuje úprave a vytváraniu grafiky, tvorbe propagačných materiálov v nástrojoch od Adobe (PhotoShop, InDesign).

#### BC. FILIP JURČACKO

- študent softvérového inžinierstva, z programovacích jazykov preferuje C a Javu. Má ročnú pracovnú skúsenosť s vývojom webových aplikácií na pozícií Junior Java Developer a momentálne pracuje ako Junior Malware Analyst v ESETe. Vo voľnom čase sa venuje programovaniu Android aplikácií a vzdelávaniu v oblasti počítačovej bezpečnosti.

#### BC. DALIBOR MÉSZÁROS

- absolvoval bakalársky stupeň študijného oboru informatika na FIIT STU. Vo všeobecnosti preferuje programovací jazyk C++, alternatívne programuje v Jave. Má základné skúsenosti s počítačovou grafikou a tvorbou webových stránok. Vo voľnom čase sa venuje štúdiu japončiny a písaniu skriptov (napr. AutoHotkey)

## ***1.1 Rozdelenie manažérskych úloh***

Člen tímu	Rola v tíme
BC. MATÚŠ CIMMERMAN	Manažér komunikácie, rozvrhu a plánovania
BC. IRINA DYOMINA	Manažér rozsahu projektu
BC. MICHAL FAŠÁNEK	Manažér webu a podpory vývoja
BC. JAROSLAV GAZDÍK	Manažér dokumentácie
BC. DENIS ILLÉS	Manažér rizík
BC. FILIP JURČACKO	Manažér vývoja
BC. DALIBOR MÉSZÁROS	Manažér kvality a hlavný architekt

## ***1.2 Podiel práce***

BC. MATÚŠ CIMMERMAN

- dok. k riadenie: úvod, podiel práce, rozdelenie manažérskych úloh, manažment podpory vývoja, manažment rizík, manažment plánovania úlohy, manažment vývoja, sumarizácia šprintov 1, 2 a 1-5 v LS, exporty úloh z Jiry, metodika pre logovanie v projekte, globálna retrospektíva LS, export webového sídla projektu
- dok k inž. dielu: globálne ciele, úvod, leap, dynamické pozadie, automaticky generovaná technická dokumentácia zo zdrojových kódov

BC. IRINA DYOMINA

- dok. k riadenie: metodika prehliadky kódu, manažment prehliadky kódu,
- dok k inž. dielu: moduly systému, globálne ciele, diagramy projektu, automatické generovanie dokumentácie pomocou nástroja Sphinx. HTML dokumentácia projektu - spojenie všetkých dokumentácií do jednej.

BC. MICHAL FAŠÁNEK

- dok. k riadenie: úvod, zoznam kompetencií tímu, globálna retrospektíva
- dok k inž. dielu: moduly systému, globálne ciele, 3D myš

#### BC. JAROSLAV GAZDÍK

- s dok. k riadenie: metodika manažmentu verzií, manažment dokumentácie
- dok k inž. dielu: moduly systému, globálne ciele

#### BC. DENIS ILLÉS

- dok. k riadenie: metodika vytvárania úloh v nástroji na evidenciu úloh, manažment rizík, sumarizácia šprintu 3
- dok k inž. dielu: úvod, globálne ciele, inštalčná príručka, 3D zobrazenie pomocou Nvidia 3D Vision, Multiview (Wall)
- formátovanie dokumentov

#### BC. FILIP JURČACKO

- dok. k riadenie: metodika testovania, manažment testovania
- dok k inž. dielu: inštalčná príručka, globálne ciele, okuliare pre zobrazenie v obohatenej realite Vuzix, senzor pre snímanie ruky Leap

#### BC. DALIBOR MÉSZÁROS

- dok. k riadenie: metodika komunikácie, manažment vývoja
- dok k inž. dielu: inštalčná príručka, globálne ciele, 3D myš

## **2 Aplikácie manažmentov**

Každý člen je zodpovedný za príslušnú oblasť manažmentu. Na dodržiavanie metodík a manažment danej oblasti zodpovedá príslušný manažér.

### ***2.1 Manažment dokumentácie***

Dokumentácia k projektu bola vytváraná od začiatku práce na projekte. Na pravidelnom týždennom stretnutí sa zakaždým spisuje zápisnica. Túto zápisnicu má na starosti každý týždeň iný zapisovateľ. Zápisnicu tvorí vyhodnotenie úloh z minulého týždňa, písaný obsah priebehu stretnutia a zadanie úloh na budúci týždeň.

Na konci každého šprintu bola spísaná retrospektíva šprintu. Táto retrospektíva rekapituluje, čo sa nám počas šprintu podarilo, čo nešlo podľa našich predstáv a čo by bolo treba zlepšiť do budúcnosti.

Dokumentácia inžinierskeho diela bola vytvorená v spoločnom Google Docu. Prácu na dokumentácii rozdeľuje medzi členov tímu primárne manažér dokumentácie podľa toho, čo kto robil a čo je jeho úlohou v tíme.

## ***2.2 Manažment podpory vývoja***

Manažment podpory vývoja riadi problematiku správneho písania a komentovania zdrojových kódov v tomto projekte. Je dôležité, aby sa metodika, ktoré boli pre túto oblasť vytvorené, riadil každý člen tímu. Metodika bola čiastočne formalizovaná na konci tretieho šprintu. Do konca zimného semestra bude komunikovaná vo finálnej verzii a upravovaná len ad-hoc.

Na tento manažment priamo nadväzuje manažment prehliadky kódu kde sme schopní včas, pred zaradením zdrojového kódu do projektu, odhaliť nedostatky, ktoré vznikli počas vývoja. Najčastejšie to môže byť nedostatočné alebo nesprávne komentovanie kódu podľa predpisu. Či zlé formátovanie zdrojového kódu.

Pre uľahčenie formátovania si každý nastavil editor vývojového prostredia rovnako. Zodpovednosť za kód, ktorý je odovzdaný na prehliadku a žiadosť o zaradenie do projektu, nesie programátor. Každý by si mal teda dodržiavať príslušnú metodiku a riadiť sa ňou v čo najväčšej možnej miere, výnimky môžu nastať.

## ***2.3 Manažment testovania***

Každý si testuje svoj vlastný kód, a ak je to možné, vytvorí aj unit testy, ktoré vykonajú zvolený kód s daným očakávaním výsledkom.

Pred tým, ako sa ukončí šprint, prebehne regresné a akceptačné testovanie. Akceptačné testovanie má za úlohu overiť správnosť pridanej funkcionality. Regresné testovanie má za úlohu overiť, či sme nepokazili iný starší kód ktorého sa nové zmeny priamo netýkajú.

V prípade že odhalené chyby nevieme opraviť ihneď, sú k nim vytvorené úlohy v JIRE, ktoré sa následne riešia v ďalšom šprinte.



## ***2.4 Manažment prehliadky kódu***

V momente keď je kód otestovaný autorom, členovia tímu sa dohodneme kto bude kód prehliadať, prehliadku kódu spravia najmenej dvaja členovia tímu. Nový kus kódu sa prečíta, zjavné chyby budú ihneď opravené. Správca prehliadky kódu overí či je kód porozumiteľný, či je dostatočne okomentovaný, skontroluje formátovanie a dodržiavanie pravidiel napísania kódu v jazyku C++. Následne kód prejde overenie code-review nástrojmi a kontrolu warningov pri kompilácii, preverí sa ako po týchto zmenách funguje program. V prípade nefunkčnosti alebo pri chybách v kóde, autor kódu bude oznámený že treba svoju prácu prerobiť. Keď je všetko v poriadku, úlohu považujeme za spravenú a predlžujeme pracovať nad projektom ďalej.

## ***2.5 Manažment rizík***

Od prvých stretnutí sme priebežne identifikovali niekoľko rizík, ktoré môžu negatívne ovplyvniť vývoj projektu. Skutočnosť, že pracujeme na rozbehnutom a pomerne veľkom projekte je najväčšie riziko, s ktorým sme sa museli od začiatku aktívne vysporiadať. Museli sme pochopiť dôležité časti zdrojového kódu, o čo sme sa aktívne snažili štúdiom kódu, opravovaním chýb, či konzultovaním s členmi tímu a tiež s vedúcim projektu. Ďalšie podstatné riziká, ktoré by bolo potrebné opisovať sme neidentifikovali.

## ***2.6 Manažment plánovania úloh***

Pre manažment a plánovanie úloh sme po otestovaní viacerých nástrojov, zvolili systém JIRA licencovaný pre fakultu. Na začiatku prvého šprintu sme naplnili backlog a identifikovali kategórie úloh. Kategórie úloh budú v priebehu projektu pribúdať, pretože mapujú veľké (tiež angl. epic) míľniky (angl. milestone).

Na začiatku každého šprintu sme diskutovali o úlohách o výbere úloh z backlogu do začínajúceho šprintu. Každú úlohu sme ohodnotili story pointami. Následne Scrum master pridelil úlohy každému z členov tímu. V polovici šprintu sme zhodnotili stav úloh a prípadne podnikli potrebné kroky, aby sme šprint ukončili čo najlepšie.

Na konci každého šprintu sme spravili retrospektívnu za posledným šprintom a zhodnotili čo sme robili zle, ale hlavne čo a ako môžeme zlepšiť do budúceho šprintu. Priebežný status úloh exportujeme na granularite jedného týždňa, aby sme mali stopu o priebehu úloh počas šprintu.

## 2.7 Manažment vývoja

Pri projekte o takejto veľkosti bolo nutné použiť nástroje pre manažment verzií, zvolili sme si nástroj Git. Na projekte pokračujeme z existujúceho projektu, ktorý sa nachádza v vzdialenom repozitári na Githube. Tu sme si vytvorili vlastnú kópiu projektu, ktorú budeme počas dvoch semestrov udržiavať, pričom na konci by sa tieto projekty mali spojiť.

Práca s verziovacím nástrojom Git je riadená príslušnou metodikou. V skratke, každá nová funkcionálna, ktorú implementuje jeden z členov by mala byť vytváraná v samostatnej vetve. Po dokončení práce vo vetve je odoslaná vo forme pull-requestu (zlúčenie) do develop vetvy vo vzdialenom repozitári. Pred zlučováním (angl. merge) musí byť vykonaná prehliadka kódu aspoň dvomi členmi, toto opäť podrobne opisuje príslušná metodika.

Pri vývoji je dôležité aby boli zmeny komitované (angl. commit) do vetvy v primerane veľkých, resp. malých inkrementoch z dôvodu prehľadnosti zmien, ktoré boli vykonané. Na konci každého šprintu musia byť všetky hotové funkcionality spojené v develop vetve vo vzdialenom repozitári a následne by mali byť spojené do hlavnej vetvy, ktorá je upravovaná vždy len na konci šprintu.

S vývojom je silno spojené debugovanie programu. Pre jednoduchšie debugovanie takto veľkého programu sú často veľkou pomocou logy programu. Preto sme vytvorili v letnom semestri metodiky pre písanie logov v projekte spolu s priloženou knižnicou. Touto metodikou by sa mal riadiť *každý*, kto programuje novú funkcionálnu alebo refaktoruje starú v projekte.

## 3 Sumarizácia šprintov

### 3.1 Šprint 1

Na začiatku sme sa oboznamovali s projektom a jeho smerovaním. Identifikovali sme naše hlavné ciele projektu.

Postupne sme sa všetci členovia tímu zoznamovali s nástrojmi a metodikami pre riadenie a manažment tímu a to napríklad Jira, či Git.

Počas prvého šprintu sme vytvorili najdôležitejšie metodiky potrebné pre vývoj a fungovanie tímu. Tieto metodiky zatiaľ ešte neboli formalizované. Vytvorili sme webové

sídlo projektu a tiež sme sa snažili “automatizovať” proces vytvárania zápisníc a exportov úloh z Jiry. Začali sme pracovať na implementačných úlohách, ktoré sme vybrali do šprintu.

Na konci šprintu sme spätne počas retrospektívy identifikovali niekoľko vecí, ktoré je potrebné zlepšiť do ďalšieho šprintu. A to najmä: prácu, pridelovanie a ohodnotenie úloh, lepšie automatizovať základnú rutinnú prácu (zápisnice), nedostatočná viditeľnosť projektu pre product ownera, zlepšiť dekompozíciu úloh, veľa úloh ostalo v stave in-progress.

### **3.2 Šprint 2**

Úspešne sme dokončili všetky neimplementačné úlohy a podarilo sa nám automatizovať rutinnú prácu. Stále sa málo implementačných úloh posunulo do stavu Done a ostali v stave ready-to-review. Hlavnou príčinou bola zlá aplikácia metodiky pre prehliadky kódu. Znamená to, že vetvy boli dokončené a bol vytvorený pull-request, ale nikto nespravil prehliadku kódu a spojenie do develop vetvy vo vzdialenom repozitári.

Zisťujeme nedostatočnú komunikáciu v tíme a tiež s vlastníkom produktu. Navrhujeme riešenie tohto problému, ktorý by mohol mať zásadný dopad na celý projekt. Dôsledky nedostatočnej komunikácie boli: nikto neurobil prehliadku dokončenej funkcionality, otázky súvisiace s úlohy sme komunikovali príliš neskoro s vlastníkom produktu.

Do budúceho šprintu by sme mali zlepšiť komunikáciu a prácu na pridelených úlohách, aktívne komunikovať stav projektu vlastníkovi produktu.

### **3.3 Šprint 3**

Počas priebehu tretieho šprintu sa nášmu tímu podarilo úspešne dokončiť veľkú časť implementačných úloh, za čo vďačíme zlepšeniu organizácii tímu z pohľadu komunikácie.

V predchádzajúcom šprinte bolo poukázané na to, že mnohé implementačné úlohy boli schopné sa dostať do revidáčnej sekcie, ale neboli nikdy patrične revidované. Tento nedostatok bol odstránený politikou, ktorú sme si určili pre tretí šprint. Vyžadovali sme aby boli jednotlivé kódy patrične zrevidované, zostavené a otestované na funkcionality ostatnými členmi tímu.

Tento jednoduchý cieľ sme dokázali dosiahnuť pomocou hromadného informovania jednotlivých členov o dostupnosti úlohy na testovanie. Následne sa očakávalo verejné prihlásenie dobrovoľníkov, ktorý by danú implementáciu vyskúšali podľa štandardných

metodik. Týmto jednoduchým prístupom sme boli schopní otestovať veľké množstvo navrhovaných zmien v štruktúre existujúceho softvéru, pričom sme odhalili nemálo nedostatkov a následne ich úspešne odstránili.

Bolo poukázané na to, že jednotliví členovia boli schopní úspešne sledovať svoje vlastné komunikačné kanály a verejné kanály. Ale najmä bolo poukázané na to, že sa naučili a osvojili si prístup sledovania cudzích komunikačných kanálov ostatných členov a ich doterajšiu produkciu v ich pridelených vetvách.

### ***3.4 Šprint 4***

Počas tohto šprintu sme sa venovali najmä príprave na prezentáciu projektu resp. riadenia v našom tíme. Väčšinu času sme venovali analýze toho, ako sa náš tím riadi a kde sú medzery. Snažili sme sa na všetko pozrieť objektívne a nedostatky zhodnotiť v pripravovanej prezentácii.

Na konci šprintu, po prezentácii sme zozbierali spätnú väzbu a pripomienku k riadeniu v našom tíme. Opäť sme zhodnotili to, ako sa riadime v našom tíme s ohľadom na získané pripomienky. Navrhli sme potrebné kroky aby sme zlepšili nedostatky, ktoré boli identifikované.

### ***3.5 Šprint 5***

Piaty šprint bol prvý šprint letného semestra. Zhodnotili sme prácu počas skúškového semestra a stanovili sme si ciele na letný semester.

### ***3.6 Šprint 6***

Na začiatku šprintu sme spájali projekty diplomantov s našim projektom aby sme mali zachovaný aktuálny projekt. Počas tohto šprintu sa nám podarilo pripraviť projekt na kompilovnaie MSVC 2013 kompilátorom na WIN32 platformách. Začali sme testovať Leap a Kinect senzor na OSX.

Úspešne sme odprezentovali prvú funkčnú implementáciu Vuzix okuliarií pre obohatenú realitu. tieto okuliare sú funkčné len na WIN32 platforme. Aktualizovali sme inštalačnú príručku o platformovo špecifické detaily.

### ***3.7 Šprint 7***

Vyriešili sme nefunkčný Leap senzor na OSX. V tomto šprinte sme sa sústredili na implementáciu 3D myši a riešenie mnohých problémov spojených s ňou. Spojili sme dokumentácie ostatných tímov a diplomantov do jednej dokumentácie, ktorá sa dá navyše generovať ako HTML dokument v štandardnom formáte pre softvérové projekty.

### ***3.8 Šprint 8***

Podarilo sa nám implementovať dynamickú zmenu pozadia v scéne programu za jeho behu. Začali sme pracovať na novej implementácii Leap senzoru s novým SDK, ktoré podporuje lepšie sledovanie rúk v priestore.

### ***3.9 Šprint 9***

Počas deviate šprintu sme sa sústredili len na dokončenie otvorených úloh do stavu Done, tak aby sme mohli na konci semestra odovzdať hotový produkt.

### ***3.10 Šprint 10***

V poslednom šprinte sme sa sústredili na finalizáciu dema projektu a najmä na dokončenie dokumentácie inžinierskeho diela ako aj dokumentácie riadenia.

## **4 Používané metodiky**

Súčasťou projektu sú metodiky, ktoré aplikujeme pri riadení a manažmente v príslušných oblastiach. Potreby a benefity metodík, ktoré aplikujeme v tomto projekte sú stručne opísané v tejto kapitole. Celé znenie každej metodiky je uvedené v prílohe A.

### ***4.1 Metodika písania a komentovania zdrojového kódu***

Metodika je určená pre všetkých členov tímu. Čiastočne pomáha aj pri procese prehliadky kódu.

Hovorí o základných konvenciách pri písaní a dokumentovaní zdrojového kódu. Cieľom je aby každý člen produkoval konzistentný a čitateľný kód pre ostatných členov, prípadne pre iných programátorov, ktorí sa s projektom predtým nestretli. Hovorí tiež o správnom spôsobe komentovania, ktoré umožní automatické generovanie dokumentácie príslušným nástrojom, v

našom prípade Doxygen. Častým problémom je rozdielne nastavenie vývojového prostredia, je nutné aby mal každý člen nastavené rovnaké odsadzovanie bielymi znakmi. Nekonzistencia spôsobuje problémy pri spájaní vetiev do vzdialeného repozitára a neprehľadnosť zmien.

#### **4.2 Metodika vytvárania úloh v nástroji na evidenciu úloh**

Metodika je určená hlavne pre administrátora nástroja Jira. Admin práva majú pridelení všetci členovia vrátane produktového vlastníka. Pridávanie je sprístupnené všetkým, avšak hlavné zásahy – mazanie úloh, spustenie a ukončenie šprintu, môže vykonávať iba hlavný administrátor.

Dokument jasne definuje postupy pri vytváraní nových úloh. Pred jednotlivými šprintami je produktový backlog rozširovaný o nové úlohy. Úlohy sú následne kategorizované na *implementačné* a *ostatné*. Úlohy s podobnými vlastnosťami sú zorganizované do skupín – documents, bugs, warnings, epics. Každá úloha je ohodnotená story pointami. Nástroj Jira je nám nápomocná pri vytváraní retrospektívy a exporty pridelených úloh sa nemusia manuálne písať.

#### **4.3 Metodika komunikácie**

Metodika opisuje jednotlivé spôsoby komunikácie medzi jednotlivými členmi tímu a existujúce komunikačné siete a kanály, ktoré sa za daných situácií používajú, tiež opisuje prioritu sledovaných kanálov s odôvodneniami.

Tento dokument bol vypracovaný na základe štandardného členenia komunikácie na hovorovú a písanú, aktívnu a pasívnu. Detailne opisuje najmä písanú komunikáciu, pričom poukazuje na dôležitosť sledovania pasívnych komunikačných kanálov, ktoré si musia ustrážiť jednotliví členovia tímu osamote (napr. pridelený branch na GitHub-e, pridelená úloha v Jire). Členovia tímu sú tiež povinný mať všeobecný prehľad, ktorý je poskytnutý skupinovou emailovou komunikáciou, pričom sa odporúča nepoužívať súkromné spôsoby komunikácie, aby mal informačný prínos celý tím. Aktívne spôsoby komunikácie sú používané v prípade nutnej spolupráce nad zjednotenou úlohou, kde je vyžiadaná prítomnosť každého člena tímu.

Metodika tiež hovorí o povinnostiach jednotlivých členov, ktoré musia dodržiavať pre jednoduchšie a priamočiarejšie vykonávanie všetkých procesov riadenia tímového projektu.

#### ***4.4 Metodika manažmentu verzíí***

Tento projekt využíva Gitflow metodiku manažmentu verzíí. Projekt má vytvorený Git repozitár, do ktorého sa ukladajú jednotlivé verzie aplikácie. Gitflow metodika sa zaoberá manažmentom Git vetiev a má celkom striktné pravidlá ich tvorby a zlučovania. Táto metodika popisuje aké vetvy využívame, ako vytvárať nové vetvy a akým spôsobom ich zlučovať naspäť do Master vetvy.

#### ***4.5 Metodika testovania***

Cieľom metodiky testovania je popísať členom tímu ucelené postupy, ako počas vývoja testy vytvárať a vykonávať. Metodika popisuje unit testy, ich vytváranie a vykonávanie v QtCreator-e, regresné testovanie a akceptačné testovanie.

#### ***4.6 Metodika prehliadky kódu***

Pre dosiahnutie väčšej kvality, po napísaní kód prejde prehliadkou. Code review metodika popisuje jednotlivé kroky, ktorých treba dodržiavať sa pri prehliadke kódu, na čo treba pamätať, aké nástroje použiť.

#### ***4.7 Metodika pre logovanie***

Pre jednoduchšie debugovanie projektu sme sa rozhodli zaradiť do projektu metodiku pre písanie logov, ktoré môžu veľmi uľahčiť debugovanie projektu. Touto metodikou by sa mal riadiť každý, kto vytvára novú funkcionálnu alebo refaktoruje pôvodnú funkcionálnu v projekte.

## **5 Globálna retrospektíva**

V tejto podkapitole sumarizujeme všetky procesy, ktoré vznikli v našom tíme a projekte. Identifikujeme tri kategórie procesov, tie ktoré potrebujeme vytvoriť a aplikovať, procesy ktoré sú nefunkčné alebo zbytočné a procesy, ktoré potrebujeme zaviesť.

Vytvoriť nové procesy:

- Riadenie a tvorba dokumentácia na priebežnej báze.
- Commitovať zmeny na menšej granularite.

- Dodržiavať striktné metodiky, ktoré boli vytvorené pre podporu vývoja a riadenia v tíme.
- Neoficiálne stretnutia tímu v laboratóriu.

Eliminovať nefunkčné procesy:

- Všetky úlohy musia byť dokončené v šprinte, v ktorom boli pridelené, okrem blokujúcich a výnimočných stavov.
- Priebežne pracovať na úlohách.

Pokračovať v procese:

- Dekomponovať úlohy.
- Prehliadky kódu aspoň dvomi členmi pred spojením do vzdialeného repozitára.
- Komunikovať problémy a nejasnosti s vlastníkom produktu.
- „Automatizovaná“ rutinná práca (zápisnice, retrospektíva)

Na konci semestra môžeme zhodnotiť, že väčšina z vyššie vymenovaných vecí bola značne posunutá k lepšiemu. Určite stále ostal priestor na zlepšenie. Nášmu tímu sa podarilo dostať produkt do stavu v akom sme ho požadoval náš zákazník. Do budúca stojí za zváženie riadenie tímu metódou SCRUM namiesto použitia napríklad Kanban, či inej metódy pre viac dynamické tímy, ktoré sú viac výskumne zamerané a granularita a náročnosť úloh vopred často nie je známa alebo sa môže zmeniť v čase.



## **6 Zoznam kompetencií tímu**

### **Matúš Cimerman**

- vedúci tímu,
- kontrola správneho písania a komentovania zdrojového kódu

### **Denis Illés**

- Plánovanie úloh v Jire
- Identifikácia a riešenie rizík

### **Dalibor Mészáros**

- Komunikácia v tíme
- Hlavný architekt projektu

### **Irina Dyomina**

- Prehliadky kódu

### **Michal Fašánek**

- Verziovanie projektu

### **Jaroslav Gazdík**

- dokumentácia

### **Filip Jurčacko**

- testovanie

## Príloha A    Metodiky

### *A.a    Metodika prehliadky kódu*

- urobiť prehľad kódu, snažiť sa ho pochopiť
  - v prípade akýchkoľvek otázok, spýtať sa autora kódu ako to myslel
  - opraviť zjavné chyby
- skontrolovať či je kód okomentovaný
  - či je kód okomentovaný v dostatočnej miere
  - doplniť komentáre keď treba, dopísať nové
- skontrolovať formátovanie
  - či sú dodržané pravidlá napísania kódu v jazyku c++
- preveriť kód nástrojmi cppcheck, cpplint
- urobiť kontrolu warningov pri kompilácii
- overiť funkčnosť nového kódu v programe
- keď všetko bude overené, treba vystaviť pull request

## ***A.b Metodika manažmentu verzii***

Používame vetvy: Master, Develop, Feature, Hotfix

**Master** - hlavný projekt

**Develop** - branchnutá z mastra, každý ťprint ma vlastnú Develop vetvu, na konci šprintu sa mergne späť do mastra

**Feature** - branchnutá z developu, každý nový kus funkcionality (task v Jire), ktorý sa kódi musí mať vlastnú Feature vetvu, po dokončení a validácii kódu sa mergne späť do Developu, NEINTERAGUJE S MASTER VETVOU

**Hotfix** - vetva na rýchly fix priamo z mastera, merguje sa do mastera AJ developu, navyšuje aktuálnu verziu

Vždy mergujeme cez Shell a s prepínačom „--no-ff“

Cheat sheet so vsetkymi základnými commandmi:

<https://training.github.com/kit/downloads/github-git-cheat-sheet.pdf>

### **Pull requesty:**

Po dokončení práce, keď sme *ready to review* sa dáva *pull request* na vetvu, do ktorej sa bude mergovať. Pull request sa robí z GUI GitHubu (pravý horný roh), alebo „\$ git request-pull {meno\_commitu} {URL}“. Pozn. doporučujem robiť cez GUI

Po odsúhlasení Pull requestu sa potom pristúpi k mergu.

### **Commit messages:**

v Commit messages používame tagy na začiatok:

[FIX] - fixli sme nejakú chybu z minula, bugfix, hotfix a podobne

[ADD] - pridali sme novú funkcionality, súbor, ...

[DOC] - pridali sme dokumentáciu, komentý...

[REF] - pre refaktoring

[FMT] - formátovanie textu, úprava

[TEST] - pre testy

Za tým veľmi stručne (a výstižne) opíšeme, aké zmeny sme spravili. Message by mali byť krátke, no pokrývať všetko, čo sme v commite spravili.

### **Tvorba feature branchu:**

```
$ git checkout -b "feature/meno-feature" develop
//Switched to a new branch "feature/meno-feature"
```

### **Mergovanie hotového feature:**

```
$ git checkout develop
//Switched to branch 'develop'
$ git merge --no-ff meno-feature
$ git branch -d "feature/meno-feature"
//Deleted branch meno_feature (was 05e9557).
$ git push origin develop
```

### **Tvorba hotfix branch-u:**

```
$ git checkout -b "hotfix/nazov-co-fixujem" master
//Switched to a new branch "hotfix-{cislo_verzie}"
$ ./bump-version.sh {cislo_verzie}
//Navýšenie verzie
$ git commit -a -m "Bumped version number to {cislo_verzie}"
```

### **Uzatvorenie Hotfix branchu:**

```
//-----MERGE DO MASTRA-----
$ git checkout master
//Switched to branch 'master'
$ git merge --no-ff hotfix/nazov-co-fixujem
$ git tag -a {cislo_verzie}
//-----MERGE DO DEVELOP-----
$ git checkout develop
//Switched to branch 'develop'
$ git merge --no-ff "hotfix/nazov-co-fixujem"
$ git branch -d hotfix/nazov-co-fixujem
//Deleted branch hotfix/nazov-co-fixujem (was abbe5d6).
```

## A.c Metodika písania a komentovania zdrojového kódu

Metodika je určená pre všetkých členov tímu. Čiastočne pomáha aj pri procese prehliadky kódu. Tento dokument hovorí o základných konvenciách pri písaní a dokumentovaní zdrojového kódu. Cieľom je aby každý člen produkoval konzistentný a čitateľný kód pre ostatných členov, prípadne pre iných programátorov, ktorý sa s projektom predtým nestretli. Hovorí tiež o správnom spôsobe komentovania, ktoré umožní automatické generovanie dokumentácie príslušným nástrojom, v našom prípade Doxygen1. Častým problémom je rozdielne nastavenie vývojového prostredia, je nutné aby mal každý člen nastavené rovnaké odsadzovanie bielymi znakmi. Nekonzistencia spôsobuje problémy pri spájaní vetiev do vzdialeného repozitára a neprehľadnosť zmien.

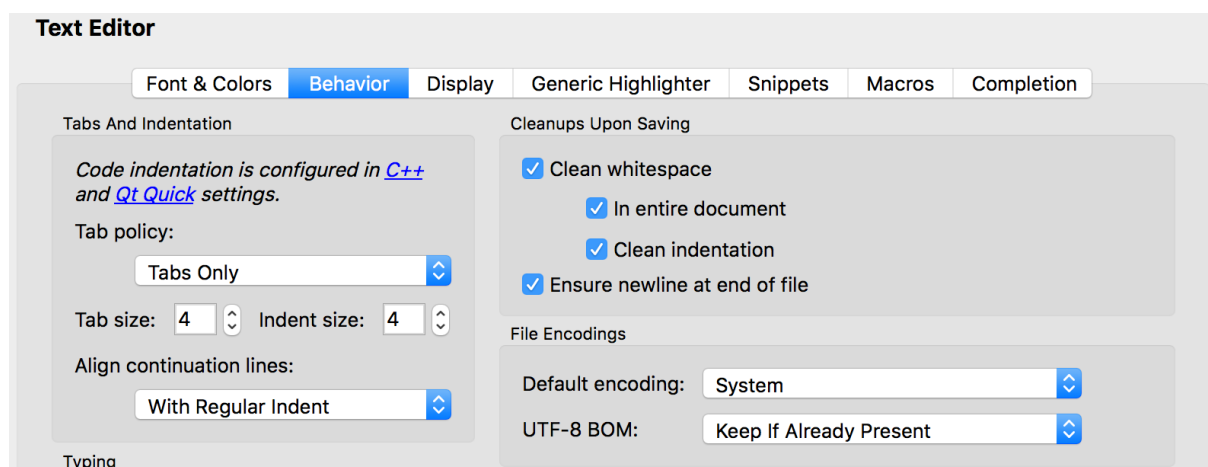
Komentovanie zdrojového kódu Používame dva typy komentárov:

- jednoriadkový, na jeden riadok do 80 znakov, stručne opisuje časť kódu,
- blokový komentár, opisuje väčšiu časť kódu, zväčša metódu alebo triedu.

### Nastavenie editora

Biele znaky, ktoré vznikajú pri odsadzovaní kódu spôsobujú problémy pri verziovaní projektu nástrojom Git (generuje to umelé zmeny). Preto je potrebné konzistentné nastavenie vývojové prostredia všetkými členmi. V našom projekte používame vývojové prostredie QT Creator.

Výber v hornej lište Tools -> Options, následne v ľavom menu položku Text Editor a záložka Behavior. Nastaviť podľa obrázku nižšie.



Obrázok 1 - nastavenie editora

### Zlé praktiky:

- nikdy nepoužívať using namespace a tiež keyword using.
- nepoužívať 0 (nulu) ako NULL pointer, vždy použiť nullptr (ak kompilátor podporuje C++11), inak NULL.
- otvorenie pull-requestu bez predchádzajúcej aplikácie metodiky pre prehliadku kódu, warningy počas kompilácie.
- pri testovaní pointrov, použite (!myPtr) alebo (myPtr); nepoužívajte myPtr != nullptr.
- neporovnávajte premenné spôsobom `x == true`, alebo `x == false`; použite `(x)` alebo `(!x)`.

### Dobré praktiky:

- inicializovať všetky atribúty v konštruktore cez initialization list,
  - ak je v konštruktore keyword new, potom v deštruktore musí byť delete,
  - inializovať v takom poradí v akom sú zapísané v triede, resp. usporiadať od najväčších (pointer, trieda, double, int, char, ...)
- hlavičkové súbory sú rozdelené do troch blokov a mali by byť zoradené podľa abecedy:
  - hlavný hlavičkový súbor, Foo.h v Foo.cpp,
  - súbory štandardnej knižnice: #include <vector>
  - ostatné súbory externých projektov: #include <osg/PolygonMode>
- prefixy premenných by mali mať výpovednú hodnotu pre čitateľa kódu:
  - k = konštanta (napr. kLimit)
  - g = globálna premenná (napr. gConfig)
  - a = argument (napr. aCnt)
  - C++ špecifické prefixy:
    - ! s = static member (napr. sPrefChecked)
    - ! m = member (napr. mLength)

## ***A.d Metodika vytvárania úloh v nástroji na evidenciu úloh***

Metodika je určená predovšetkým hlavnému administrátorovi nástroja AJira, ktorý je zodpovedný za správne nastavenie, zaradenie a ohodnotenie úloh. Cieľom dokumentu je poskytnúť jednotný spôsob zakladania úloh, ktorým zabezpečíme konzistenciu a prehľad priebehu riešení problémov týkajúcich sa projektu.

### **Pojmy:**

JIRA – nástroj na sledovanie úloh

ISSUE – úloha

SPRINT – definovaný časový úsek (u nás 2 týždne)

PRODUCT BACKLOG – zoznam úloh, ktoré sa neriešia v aktuálnom šprinte

SPRINT BACKLOG – zoznam úloh, ktoré sme zaradili do šprintu a majú byť splnené do konca šprintu

### **Vytvorenie úlohy:**

- Create issue
- Zadať jednoznačný názov
- Zaradiť do správnej kategórie – LABEL: *implementaion* alebo *other*
- Pripojiť príslušný epicLink – warning, bug, document, AR glasses,...
- Opísať úlohu v časti Description
- Kliknúť create

Následne vytvorenú úlohu môžeme ohodnotiť story pointami. Na číse sa zhodneme použitím scrum pokeru, kde sa zoberie priemerná hodnota s tým, že člen s najvyšším a najnižším hodnotením okomentuje svoju voľbu.

Pri vytvorení šprintu sú úlohy presunuté z produktového backlogu do backlogu šprintu. Množstvo súvisí od predošlých šprintov. Ak sme nestihli vyriešiť všetky úlohy z predošlého šprintu, tak nasledujúci šprint bude obsahovať menší počet úloh (= suma story pointov úloh bude menšia). V opačnom prípade je počet úloh pre šprint zvýšený.

Stavy úloh sa môžu nachádzať pre *other*: TODO, INPROGRESS, DONE a pre *implementation*: TODO, INPROGRESS, READYTOREVIEW, DONE.

## ***A.e Metodika testovania***

Cieľom metodiky testovania je popísať členom tímu ucelené postupy, ako počas vývoja testy vytvárať a vykonávať. Metodika popisuje Unit testy, ich vytváranie a vykonávanie v QtCreator-e, regresné testovanie a akceptačné testovanie.

### **Pojmy**

QtCreator - vývojové prostredie

JIRA - nástroj na sledovanie úloh

### **Unit testovanie**

Unit testovanie prebieha počas celého šprintu, unit testy vytvárajú a vykonávajú autori na vlastnom kóde. Na vytváranie unit testov používame framework QTestLib.

### **Vytvorenie unit testov v QtCreator-e**

Vstup: funkcie projektu

Výstup: testovacie funkcie

Zodpovedný: programátor

Najprv vytvoríme testovaciu triedu, ktorá dedí od QObject. Jednotlivé testovacie funkcie musia byť deklarované pod private slots.

```
#include <QtTest/QtTest>
class TestQString: public QObject
{
    Q_OBJECT
private slots:
    void toUpper();
};
```

Následne implementujeme jednotlivé testovacie funkcie.

```
void TestQString::toUpper()
{
    QString str = "Hello";
    QVERIFY(str.toUpper() == "HELLO");
}
```

### **Vykonanie unit testu:**

Vstup: testovacie funkcie



Výstup: výsledky testov

Zodpovedný: programátor

### **Príkaz na spustenie testu má formát:**

```
testname [options] [testfunctions[:testdata]]...
```

príklad:

```
/myTestDirectory$ testQString toUpper
```

### **Vyhodnotenie unit testu:**

Vstup: výsledky testov

Výstup: oprava prípadných chýb

Zodpovedný: programátor

### **Výsledky testu sú v nasledovnom formáte:**

```
***** Start testing of TestQString *****  
Config: Using QTest library 4.8.5, Qt 4.8.5  
PASS : TestQString::toUpper()  
Totals: 1 passed, 0 failed, 0 skipped  
***** Finished testing of TestQString *****
```

Výstup obsahuje zoznam všetkých vykonaných testovacích funkcií, kde každý test skončí buď úspechom alebo zlyhaním. Programátor identifikuje chyby na základe testov ktoré zlyhali, následne ich opraví.

Framework QTestLib umožňuje simulovať udalosti na GUI ako napríklad kliky na menu a tlačidlá, preto je možné pre našu aplikáciu vytvoriť aj komplexné unit testy.

### **Regresné testovanie:**

Pred ukončením šprintu sa vykonajú všetky doposiaľ vytvorené unit testy. Ich výstup ukáže zmeny správania existujúceho kódu, ktoré sú pravdepodobne zapríčinené novšími zmenami.

### **Akceptačné testovanie:**

Ako posledné pred ukončením šprintu prebieha akceptačné testovanie. Akceptačné testovanie je robené manuálne, overuje funkcionality používateľských scenárov, pridanej funkcionality.

### **Príprava testu:**

Vstup: požiadavka na funkcionality, používateľský scenár

Výstup: testovací scenár

Zodpovedný: Product Owner

Akceptačný test sa najprv vytvorí, v podobe scenára ktorý naplňa požiadavku na funkcionality. Obsahuje nasledovné:

**Názov** – obsahuje názov testovacieho scenára

**Účel** – obsahuje čo je daným testovaním overované

**Vstupné podmienky** – podmienky, ktoré musia byť splnené pred vykonávaním jednotlivých akcií

**Výstupné podmienky** – podmienky, ktoré musia byť splnené po vykonaní jednotlivých akcií

**Akcia** – akcia v systéme, ktorú používateľ vykoná

**Očakávaná reakcia** – reakcia, ktorá je očakávaná po vykonaní danej akcie

**Skutočná reakcia** – reakcia, ktorá po vykonaní danej akcie skutočne nastala

Príklad testu:

<b>ID</b>	1	<b>Názov</b>	Navigácia v 3D priestore prostredníctvom 3d myši	
<b>Účel</b>	Overenie funkčnosti 3D myši			
<b>Vstupné podmienky</b>		3D myš je pripojená k počítaču		
<b>Výstupné podmienky</b>				
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná reakcia</b>	<b>Skutočná reakcia</b>	
1	Pohyb do strán	Posunutie kamery doľava a doprava		
2	Pohyb hore a dole	Posunutie kamery hore a dole		
3	Pohyb otáčanie a naklápanie	Rotácia		

**Vykonanie testu:**

Vstup: testovaný softvér, testovacie scenáre

Výstup: doplnené testovacie scenáre o skutočné výsledky

Zodpovedný: Product Owner

Testy vykonáva Product Owner podľa predpísaných krokov v testovacích scenároch.

**Vyhodnotenie testu:**

Vstup: doplnené testovacie scenáre o skutočné výsledky

Výstup: oprava prípadných chýb

Zodpovedný: programátori

V prípade že odhalené chyby nevieme opraviť ihneď, sú k nim vytvorené úlohy v JIRE, ktoré sa následné riešia v ďalšom šprinte.

## A.f Metodika pre logovanie

Na logovanie používame zvyčajne predvolený logger *LOG*. Ak by bolo potrebné použiť logger pre špecifický účel, treba použiť *CLOG*. Použitie je nasledovné:

```
LOG(INFO) << "This is info log";  
CLOG(ERROR, "performance") << "This is info log using performance logger";
```

V každej triede kde chceme použiť logovanie je potrebné pripojiť na začiatku `easylogging.h` nasledovne: `#include "easylogging++.h"`

Logovací nástroj poskytuje niekoľko úrovní logovanie, výber úrovne závisí od časti kódu a kontextu, ktorý bude logovaný:

Level	Popis
Global	Generická úroveň pre všetky úrovne.
Trace	Sledovanie konkrétnych udalostí, ktoré nastali v programe.
Debug	Debugovanie aplikácie.
Fatal	Kritická chyba, ktorá nastala v programe a vedie k zastaveniu programu.
Error	Chyba, ktorá nastala v programe, ale nevedie k zastaveniu programu.
Warning	Upozornenie na chybu, ktorá nemá dopad na funkcionality programu v úzkom kontexte.
Info	Informačné logy.
Verbose	Podrobné logovanie programu.
Unknown	Pokiaľ nevieme nikam zaradiť úroveň logu, neodporúča sa používať.

# Príloha B Zázpisnice

## B.a Prvý týždeň

**Autor:** Matúš Cimerman

**Dátum strenutia:** 30.9.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.  
Členovia tímu: Bc. Matúš Cimerman  
Bc. Irina Dyomina  
Bc. Michal Fašánek  
Bc. Jaroslav Gazdík  
Bc. Denis Illés  
Bc. Filip Jurčacko  
Bc. Dalibor Mészáros

**Priebeh strenutia:**

- Diskusia o projekte,
  - Historia,
  - Na čo sluzi a čo momentálne dokaze,
  - Predstavenie vizie.
- Diskusia o:
  - SCRUMe,
  - O riadení a nástrojoch riadenia,
  - O tom ako budeme komunikovať mimo strenutí.

**Úlohy na budúce strenutie:**

ID	Riešiteľ	Opis úlohy
-	Volba "sefa" tímu	Tím si zvolí zodpovedného člena za tím, ktorý bude riešiť prípadne problémy.
-	Vyber hlavného architekta	Hlavný architekt projektu, mal by mať najlepší prehľad o všetkých častiach systému.
-	Vyber nástroja na task mgmt	Trello, Jira, Redmine alebo niečo iné?
-	Navrhnut web tímu	Webová stránka projektu

## ***B.b Druhý týždeň***

**Autor:** Matúš Cimerman

**Dátum strenutia:** 30.9.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.  
Členovia tímu: Bc. Matúš Cimerman  
Bc. Irina Dyomina  
Bc. Michal Fašánek  
Bc. Jaroslav Gazdík  
Bc. Denis Illés  
Bc. Filip Jurčacko  
Bc. Dalibor Mészáros

**Stav úloh z minulého strenutia:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>	<b>Stav</b>
-	Tím	Voľba "šéfa tímu"	done
-	Tím	Voľba hlavného architekta	done
-	Tím	Výber nástroja na task mgmt	done, Jira
-	Fašánek	Web projektu	in-progress

**Priebeh strenutia:**

- oboznamovanie sa s projektom a jeho časťami,
- diskusia o BDD/TDD frameworku pre cpp,
- diskusia o potrebe logovania udalostí v projekte a výbere knižnice pre logovanie,

**Úlohy na budúce strenutie:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>
-	Tím	Výber frameworku na BDD/TDD development.
-	Tím	Výber knižnice pre logovanie v cpp programoch.

## ***B.c Tretí týždeň***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 7.10.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.  
Členovia tímu: Bc. Matúš Cimerman  
Bc. Irina Dyomina  
Bc. Michal Fašánek  
Bc. Jaroslav Gazdík  
Bc. Denis Illés  
Bc. Filip Jurčacko  
Bc. Dalibor Mészáros

**Stav úloh z minulého stretnutia:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>	<b>Stav</b>
-	Tím	Výber frameworku na BDD/TDD development	done, vybraný framework Igloo
-	Tím	Výber frameworku pre logovanie v cpp programoch	done, vybraná knižnica Pantheios
-	Fašánek	Web projektu	in-progress

**Priebeh stretnutia:**

- oboznamovanie sa so systémom,
- príprava backlogu,
- riešenie inštalačných problémov, kompilácia projektu.

**Úlohy na budúce stretnutie:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>
-	Cimerman	Gitflow - naštudovať
-	Illés	Jira - vytvorenie projektu, nastavenie
-	Illés, Mészáros	Aktualizácia inštalačného manuálu

## ***B.d Štvrtý týždeň***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 14.10.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

**Stav úloh z minulého stretnutia:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>	<b>Stav</b>
-	Cimerman	Gitflow - naštudovať	done
-	Illés	Jira - vytvorenie projektu, nastavenie	done
-	Illés, Mészáros	Aktualizácia inštalačného manuálu	done

**Priebeh stretnutia:**

- oboznamovanie sa so systémom,
- plnenie backlogu,
- riešenie inštalačných problémov, kompilácia projektu.

**Úlohy na budúce stretnutie:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>
-	Cimerman	gitflow - vytvoriť metodiku
-	Gazdik	dokumentácia - vytvoriť metodiku
-		Zjednotiť metodiky: ako programovať v cpp, gitflow, formátovanie kódu, ...
23	Všetci	[Task] Nastaviť konzistentne editor



## ***B.e Piaty týždeň***

**Autor:** Filip Jurčacko

**Dátum strenutia:** 21.10.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.  
Členovia tímu: Bc. Matúš Cimerman  
Bc. Irina Dyomina  
Bc. Michal Fašánek  
Bc. Jaroslav Gazdík  
Bc. Denis Illés  
Bc. Filip Jurčacko  
Bc. Dalibor Mészáros

**Stav úloh z minulého strenutia:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>	<b>Stav</b>
78	Cimerman	Vytvoriť GitFlow metodiku	in--progress
1	Fašánek	Web projektu	done
7	Gazdík	Zápisnice + šablóna	done
34	Mészáros + Illés	Aktualizácia OSG	done
10	Illés	Spustenie bez cotire, doplnenie includov	done
9	Mészáros	Opravovanie warningov	done
12	Jurčacko	"&Multi-select mode" bug	in progress
23	Všetci	Nastaviť konzistentne editor	done

**Priebeh strenutia:**

- oboznamovanie sa so systémom,
- plnenie backlogu + poker pre druhý šprint
- riešenie inštalčných problémov, kompilácia projektu.

**Úlohy na budúce stretnutie:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>
50	Jurčacko	Vytvoriť Zápisnicu w5
78	Cimerman	Vytvoriť GitFlow metodiku
56	Mészáros	Opraviť warningy - cppCheck
58		Opraviť warningy - MSVC
57	Cimerman	Opraviť warningy - CppLint
12	Jurčacko	"&Multi-select mode" bug
80	Illés	Aktualizovať inštalačný návod
41	Cimerman	Vytvoriť dok. s retrospektívou
49	Cimerman	Generovanie dokumentácie a diagramov
2	Fašánek	Nahrať dokumenty na webstránku

## ***B.f Šiesty týždeň***

**Autor:** Dalibor Mészáros

**Dátum stretnutia:** 28.10.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

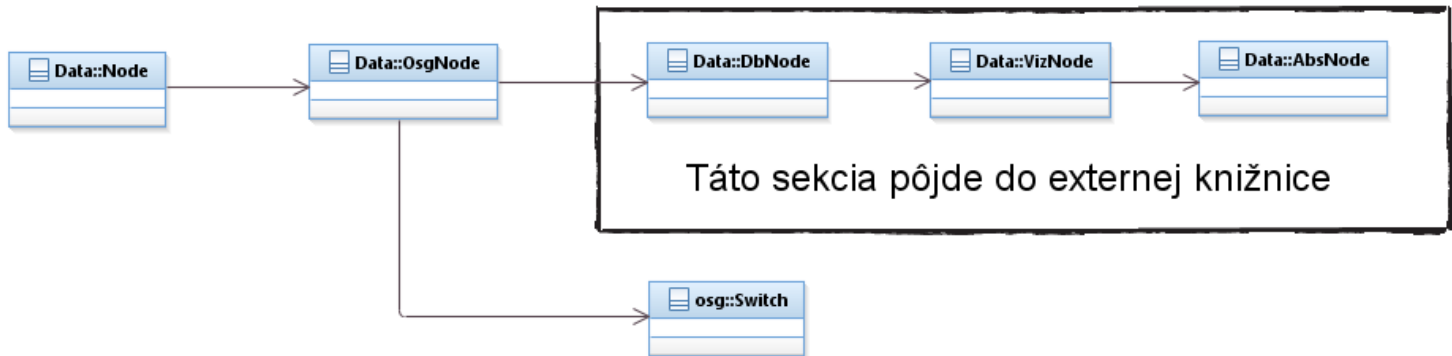
**Stav úloh z minulého stretnutia:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Opis úlohy</b>	<b>Stav</b>
50	Jurčacko	Vytvoriť Zápisnicu w5	done
78	Gazdík, Cimerman	Vytvoriť GitFlow metodiku	done
56	Mészáros, Jurčacko, Gazdík	Opraviť warningy - cppCheck	in progress
58	Mészáros	Opraviť warningy - MSVC	in progress
57	Cimerman	Opraviť warningy - CppLint	in progress
12	Jurčacko	"&Multi-select mode" bug	in progress
80	Illés	Aktualizovať inštalačný návod	in progress
41	Cimerman	Vytvoriť dok. s retrospektívou	done
49	Cimerman	Generovanie dokumentácie a diagramov	in progress
2	Fašánek	Nahrať dokumenty na webstránku	done

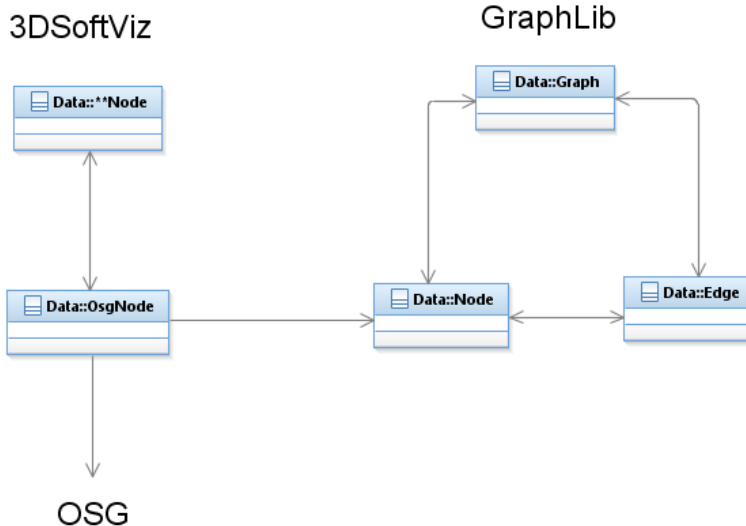
**Priebeh stretnutia:**

- Diskutovali sme zmeny, problémy a riešenia vo vetve "update-aruco"
- Prediskutovali sme našu zverejnenú GitFlow metodiku a dohodli sme sa na dodržovaní opísaných metodík
  - Ohodnotili sme všetky implementačné problémy story pointami, ktorým chýbali zadané hodnoty
  - Popridelovali sme zvyšné nepriradené úlohy jednotlivým členom

- Zaoberali sme sa problematikou refaktoriácie pre vzťahy tried "Node", "OsgNode" a "DbNode", "VizNode", "AbsNode"
- Úlohou refaktoriácie je vyňať spojenie "DbNode", "VizNode", "AbsNode" a zabezpečiť pripojenie k projektu vo forme externej knižnice



- Refaktoriáciu treba aplikovať na sekcie "Node", "Edge" a čiastočne "Graph"



- Odhadovaný čas takejto komplikovanej refaktoriácie vychádza približne na jeden mesiac až celý semester
- Zaoberali sme sa problematikou 3D myšky a jej integrácie do projektu
  - Treba zistiť aké header súbory treba pridať z ponúkaného SDK
  - Treba modifikovať Cmake a cmake/Find\*, aby bolo možné pridať náš modul
  - Je potrebné vytvoriť novú triedu - interface a core
  - Interface - minimálne množstvo tried, aby sme mohli vymeniť core v prípade potreby
  - Interface - mal by obsahovať v najhoršom prípade zvlášť triedy pre zachytávanie pohybu, rotácie a stlačenia
  - Core - trochu viac tried, ako v prípade interface, majú obsahovať priame ovládanie cez SDK

- Namapovať jednotlivé pohyby podľa dizajnu Kinect tried
  - Mali by sme zmazať "ransac" a "vector3" v Kinecte
- Mali by sme si rozmyslieť ako namapujeme jednotlivé pohyby pre "orbitálny fixný" a "voľný letecký" pohyb.
- Na ďalšom stretnutí zozbierame výsledky, dokumenty a pokračujem ďalším šprintom
- Mali by sme si pozrieť informácie k "logForCpp", alebo "logForCxx", ktoré by potenciálne mohli vymeniť QDebug
- V rámci TDD (alebo BDD) by sme si mohli pozrieť čo je "Igloo", ako sa s tým pracuje
- V namespace "OsgModeling" je problém, že "Cube", "Sphere", "SpikySphere" sú mimo svojho namespace
- "HistoryBuffer" by sa mal nachádzať v namespace "Kinect"
- Mali by sme preskúmať adresár 3DSoftVizu a odstrániť nepotrebné súbory:
  - zbytočné .bat súbory
  - staré databázy
  - upratať obrázky v resources/
  - resources/img/ obsahujú staré fotky

### Úlohy na budúce stretnutie:

ID	Riešiteľ	Opis úlohy
49	Cimerman	Generovanie dokumentácie a diagramov
80	Illes	Aktualizovať inštalačný návod
63	Gazdik, Fašánek	3Dconnexion Mouse - preštudovať SDK
83	Cimerman	Vytvoriť metodiku - ako programovať v C++
79	Dyomina	Vytvoriť CodeReview metodiku
41	Meszaros	Vytvoriť dok. s retrospektívou
13	Illes	Tlačidlo "Load function calls" po otvorení priečinka zhodí program
12	Jurcacko	Tlačidlo "&Multi-select mode" vracia cez "getId" NULL, čo zhodí program
57	Cimerman	Opraviť warningy - CppLint
56	Meszaros	Opraviť warningy - cppCheck
58	Meszaros	Opraviť warningy - MSVC
69	Gazdik, Fašánek	3Dconnexion Mouse - vytvoriť diagramy
70	Gazdik, Fašánek	3Dconnexion Mouse - vytvoriť modul v projekte
82	Gazdik, Fašánek	3Dconnexion Mouse - vytvoriť core class / interface

## ***B.g Siedmy týždeň***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 4.11.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

**Stav úloh z minulého stretnutia:**

<b>ID</b>	<b>Riešiteľ</b>	<b>Stav</b>	<b>Opis úlohy</b>
49	Cimerman	Done	Generovanie dokumentácie a diagramov
80	Illes	Done	Aktualizovať inštalačný návod
63	Gazdik, Fašánek	Done	3Dconnexion Mouse - preštudovať SDK
83	Cimerman	in-progress	Vytvoriť metodiku - ako programovať v C++
79	Dyomina	Done	Vytvoriť CodeReview metodiku
41	Meszaros	Done	Vytvoriť dok. s retrospektívou
13	Illes	Ready to re-view	Tlačidlo "Load function calls" po otvorení priečinka zhodí program
12	Jurcacko	Merge od Garaja	Tlačidlo "&Multi-select mode" vracia cez "getId" NULL, čo zhodí program
57	Cimerman	Ready to re-view	Opraviť warningy - CppLint
56	Meszaros	Ready to re-view	Opraviť warningy - cppCheck
58	Meszaros	Ready to re-view	Opraviť warningy - MSVC
69	Gazdik, Fašánek	To-Do	3Dconnexion Mouse - vytvoriť diagramy
70	Gazdik, Fašánek	Ready to re-view	3Dconnexion Mouse - vytvoriť modul v projekte
82	Gazdik, Fašánek	To-Do	3Dconnexion Mouse - vytvoriť core class / interface

### Priebeh stretnutia:

- Diskusia o tom ako správne robiť code review:
  - Mergovať aspoň deň pred stretnutím, resp. pred koncom šprintu.
  - Na stretnutí by už malo byť všetko mergnuté a prejdené stavom ready to review.
  - Každý člen tímu by mal sledovať Jiru, a ktoré tasky sú ready to review.
  - Držať sa code review metodiky.
- Diskusia o taskoch:
  - Prečo nič z implementačných taskov nebolo v stave done? Nikto nesledoval Jiru a nerobil code review.
  - Pracovať na taskoch priebežne, nie jeden/dva dni pred stretnutím.
  - Na konci šprintu by mali byť VŠETKY úlohy v stave *done*, s výnimkou blockerov. Každý člen by teda mal mať svoju vetvu - mergnutú v develop vetve.
  - Na konci šprintu ešte celý kód vždy "upratať", aplikovať metodiky - toto by mal robiť každý ešte pred pull-requestom.
- Každý by už mal dodržiavať gitflow metodiku. Zhodnotili sme, že aktuálna gitflow metodika potrebuje nejaké úpravy, ktoré budú hovoriť o ukončení života vetvy a pull-requestoch.
  - Codereview metodika - upraviť, bola napísana veľmi všeobecne.
  - Zhodnotili sme ako sa nám darilo, či nedarilo počas druhého šprintu.
  - Ako hlavný problém v našom tíme sme identifikovali komunikáciu.
  - Za šprint by sme mali zvládnuť vyriešiť aspoň 30SP.
  - Otvorenie diskusie o dokumentácii, čo všetko chceme dokumentovať?
  - Zápisnice, retrospektívy a percentuálny podiel na šprinte by mal byť odoslaný najneskôr 1 deň po stretnutí alebo ukončení šprintu.





## B.h Ôsmy týždeň

**Autor:** Michal Fašánek

**Dátum stretnutia:** 11.11.2015

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

**Stav úloh z minulého stretnutia:**

ID	Riešiteľ	Opis úlohy	Stav
57	Cimerman	Opraviť warningy CppLint	done
84	Meszároš	Zmazať zbytočné triedy	done
70	Gazdík	3DConnexion mouse Vytvoriť modul v projekte	done
88	Dyomina	Nájsť cudzie kódy	done
13	Illés	Tlačidlo "Load function calls" zhodí program	ready to review
96	Jurčacko	Opraviť error handling	ready to review
56	Meszároš	Opraviť CppCheck warningy	ready to review
58	Meszároš	Opraviť MSCV warningy	ready to review
17	Kapec	Linkovať sa na OSG modeling	ready to review
12	Jurčacko	Tlačidlo "Multi-select Node" vracia NULL, čo zhodí program	in progress
85	Illés	Začleniť triedy do príslušných namespace-ov	in progress
86	Cimerman	Opraviť warningy - OSX	in progress
69	Dyomina	3DConnexion mouse vytvoriť diagramy	in progress
82	Gazdík	3DConnexion mouse vytvoriť core class/interface	in progress
72	Fašánek	3DConnexion mouse nama- povať na ovládanie virtuálnej	to do

		kamery	
--	--	--------	--

#### Priebeh stretnutia:

- Diskutovali sme o úspešnosti prvého týždňa šprintu
- Analyzovali sme stav úloh a pridávali nové úlohy
- Diskutovali sme o dokumentácií a spôsoboch podieľania sa jednotlivých členov
  - Ako vyhodnocovať úlohy
  - Rozpracovanie implementačných úloh
  - Možnosti písania univerzálnej dokumentácie v ktorej by mohli pokračovať aj ďalšie tímy
  - Zmeny v predošlých implementáciách
  - Diagramy
  - Big Picture obsah
- Prístup do miestnosti 1.27
- Referenčný model GitLib
  - namapovanie na 3dConnexion myš
- Súbory v Resource – čo sa ešte používa?

#### Úlohy na budúce stretnutie:

ID	Riešiteľ	Opis úlohy
13	Illés	Tlačidlo "Load function calls" zhodí program
96	Jurčacko	Opraviť error handling
72	Fašánek	3dConnexion mouse namapovať na ovládanie virtuálnej kamery
56	Meszároš	Opraviť CppCheck warningy
58	Meszároš	Opraviť MSCV warningy
17	Kapec	Linkovať sa na OSG modeling
12	Jurčacko	Tlačidlo "Multi-select Node" vracia NULL, čo zhodí program
83	Dyomina	Metodika programovania v C++
11	Illés	Aktualizovať používateľskú príručku
92	Jurčacko	Dokumentácia k produktu – 3 šprinty

## ***B.i Deviaty týždeň***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 18.11.2015

**Prítomní:** Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Diskusia o OSG Modelingu,
- Pripravit demo 3D myši – najlepšie tak, aby bola funkčná do DOD na FIIT.
- Refaktorovať diluculum mimo namespace
  - Lua, LuaGraph a LuaInterface by sa mali vyňať z projetku .
  - Do LuaGraphu a LuaInterface pridať metódy aby Lua nepoužíval Diluculum priamo.
  - Nachádza sa na 5-6 miestach v kóde.
- Diskusia o stereoskopickom zobrazení
  - OSG podpora
- Diskusia o multiview zobrazení - viac monitorov
  - OSG podpora,
  - Existujú example na to – jedno OSG okno rozťahnuté cez 3 monitory.
- Treba otestovať kalibrovanie kamery:
  - Každá kamera ma iné skreslenie,
  - Diplomovka Juríka.
- Zhodli sme sa na tom, že pripojíme výstup z PlantUML do doxygenu

- Otvorili sme diskusiu o udržiavaní robustnej a hodnotnej dokumentácie pre tento projekt, ktorú by tvorili aj nasledujúce tímove projekty po nás.
  - Nájsť možnosti generovania dokumentácie z textových súborov (niečo ako markdown štýlovanie)
  - Predpokladom sú napríklad:
    - dobre okomentovaný zdrojový kód,
    - unit testy,
    - identifikované požiadavky a analýza,
  - Nové implementované moduly do projektu by mali byť ukážkou.
  -

Stav	úloh	z	minulého	stretnutia:
Zapisnica w8		Michal Fasanek	Done	V prílohe export taskov z Jiry. w8-1_status - status uloh z minuleho tyzdna w8_todo-inprogress - zvyadne ulohy na buduci
Opraviť error handling		Filip Jurcacko	Done	Na viacerých miestach v kode je do funkcií posielaný argument bool* error ktory sa vnútri funkcie nastaví v prípade chyby  Avsak často po vykonaní funkcií buď nieje skontrolovaný vôbec alebo je viacnásobne
Dokumentácia k produktu - riadenie		Filip Jurcacko	Done	BIELIKOVA <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html">http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html</a>
Dokumentácia k produktu - BigPicture		Filip Jurcacko	Done	BIELIKOVÁ <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-">http://www2.fiit.stuba.sk/~bielik/courses/tp-</a>
Dokumentácia k produktu - 3 šprinty		Filip Jurcacko	Done	do 18.11. v AIS
Nájsť všetky cudzie kódy		Irina Dyomina	Done	identifikovať všetky cudzie kódy, a nájsť odkiaľ boli zobrať  napr: foldre: src > Noise (hlavička zobrazuje že kód je
Opraviť warningy - OS X		Matus Cimerman	Done	
Začleniť triedy do príslušných namespaceov		Denis Illes	Done	Triedy ako "Cube", "Sphere", "SpikySphere" sa nachádzajú mimo svojho namespaceu (Clustering). Treba ich začleniť. Rovnako ako "HistoryBuffer" (malo by to byť v kinect namespace)
Vytvoriť metodiku - ako programovať v C++		Matus Cimerman	Done	Vytvoriť dokument, v ktorom sa opíše postup pri programovaní v jazyku cpp. Zaznamenajú sa základné princípy, ktorých sa budeme držať pri vývoji softvéru. * astyle,
3Dconnexion Mouse - vytvoriť core class / interface		Jaroslav Gazdik	In Progress	Vytvoriť triedy na spracovanie a konvertovanie údajov získaných z myšky, napísať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvoriť interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSoftViz.
3Dconnexion Mouse - namapovať na ovládanie virtuálnej kamery		Michal Fasanek	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovládanie v oboch režimoch - voľný a orbitálny pohyb.
3Dconnexion Mouse - vytvoriť diagramy		Irina Dyomina	Done	Vytvoriť diagram s triedami znázorňujúce začlenenie modulu do projektu. Inšpirovať sa s modulom Kinect.
Opraviť warningy - MSVC		Dalibor Meszaros	Done	Warningy detekované nástrojom MSVC
Opraviť warningy - cppCheck		Dalibor Meszaros	Done	Warningy detekované nástrojom cppCheck
linkovať sa na osgModelling		Peter Kapec	Done	treba odstrániť z projektu súbory v Math/ - Bezier - Curve - Export
Tlačidlo "Load function calls" po otvorení priečinka zhodí program		Denis Illes	Done	Pri otvorení zložky s .lua súbormi program crashne. Zrejme Windows related issue, keďže na iOS funguje bez akýchkoľvek problémov.  Riešenie: zakomentovať ...\\3dsoftviz\\resources\\scripts\\luadb\\extraction\\extractor.lua line 182 (assert) alebo zmazať druhú podmienku > assert(path, "wrong path
Tlačidlo "&Multi-select mode" vracia cez "getIid" NULL, čo zhodí program		Filip Jurcacko	Done	Pri označovaní viacerých uzlov program crashne.
Aktualizovať používateľskú príručku		Denis Illes	Done	

## Úlohy na budúce stretnutie:

Upratať nesprávne umiestnené veci z 3dsodtviz	Denis Illes	In Progress	doc - doxygen resources
Aruco example testy	<i>Unassigned</i>	To Do	vysúšať 3dsodtviz\install\bin\aruco_test_gl.exe a ostatné *_gl.exe  arg. live camera.yml 0.025  freeglut.dll prekopirovať z
Refaktoring časti: Kinect, Leap, OpenCV	Filip Jurcacko	In Progress	inšpirácia - Garaj
OSG Jedno okno - rozťahnuté cez 3 monitory	Matus Cimerman	In Progress	
OSG Multiview	Denis Illes	In Progress	OSQ multiview/window - projektory virtualCave equalizer - skúsiť googlit
Refaktoring nahradiť cudzie kódy originalmi	Irina Dyomina	To Do	
Refaktoring - diluculum	Dalibor Meszaros	In Progress	diluculum použitý mimo namespace Lua (triedy LuaGraph, LuaInterface) riešenie: LuaGraph/Interface - pridať metódy
3Dconnexion Mouse - vytvoriť core class / interface	Jaroslav Gazdik	In Progress	Vytvoriť triedy na spracovanie a konvertovanie údajov získaných z myšky, napísať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvoriť interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSofViz.
3Dconnexion Mouse - namapovať na ovládanie virtuálnej kamery	Michal Fasanek	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovládanie v oboch režimoch - voľný a orbitálny pohyb.
AUGREALITY-18 Opraviť CameraManipulator.cpp	<i>Unassigned</i>	To Do	
AUGREALITY-18 Opraviť QtColorPicker.cpp	<i>Unassigned</i>	Ready to review	
AUGREALITY-18 Opraviť NoiseUtils.cpp	<i>Unassigned</i>	Ready to review	
Presunúť "cudzi" kód do dependencies	Peter Kapec	To Do	Zdrojové súbory: NoiseUtils.cpp, QtColorPicker.cpp, CameraManipulator.cpp

## ***B.j Desiaty týždeň***

**Autor:** Dalibor Mészáros

**Dátum stretnutia:** 25.11.2015

**Prítomní:** Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Diskusia ku dňu otvorených dverí a otázke prezentovania projektu 3dSoftViz:
  - Podpora zobrazenia Cave
  - Podpora zobrazenia na viacerých obrazovkách (rozšírenie)
  - Zhotoviť fotografickú dokumentáciu na prezentovanie tímu a projektu v akcií
- Boli vykonaný merge pre gitlib a osgmodeling
- Boli diskutované problémy, s ktorými mali problémy jednotliví členovia tímu s ich pridelenými úlohami

## Stav úloh z minulého stretnutia:

Upratať nesprávne umiestnené veci z 3dsodtviz	Denis Illes	In Progress	doc - doxygen resources
Aruco example testy	<i>Unassigned</i>	To Do	vysúšať 3dsodtviz_install\bin\aruco_test_gl.exe a ostatné *_gl.exe  arg. live camera.yml 0.025  freeglut.dll prekopirovať z
Refaktoring časti: Kinect, Leap, OpenCV	Filip Jurcacko	In Progress	inšpirácia - Garaj
OSG Jedno okno - rozťahnuté cez 3 monitory	Matus Cimerman	In Progress	
OSG Multiview	Denis Illes	In Progress	OSQ multiview/window - projektory virtualCave equalizer - skúsiť googlit
Refaktoring nahradiť cudzie kódy originalmi	Irina Dyomina	To Do	
Refaktoring - diluculum	Dalibor Meszaros	In Progress	diluculum použitý mimo namespace Lua (triedy LuaGraph, LuaInterface) riešenie: LuaGraph/Interface - pridať metódy
3Dconnexion Mouse - vytvoriť core class / interface	Jaroslav Gazdik	In Progress	Vytvoriť triedy na spracovanie a konvertovanie údajov získaných z myšky, napísať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvoriť interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSofViz.
3Dconnexion Mouse - namapovať na ovládanie virtuálnej kamery	Michal Fasanek	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovládanie v oboch režimoch - voľný a orbitálny pohyb.
AUGREALITY-18 Opraviť CameraManipulator.cpp	<i>Unassigned</i>	To Do	
AUGREALITY-18 Opraviť QtColorPicker.cpp	<i>Unassigned</i>	Ready to review	
AUGREALITY-18 Opraviť NoiseUtils.cpp	<i>Unassigned</i>	Ready to review	
Presunúť "cudzi" kód do dependencies	Peter Kapec	To Do	Zdrojové súbory: NoiseUtils.cpp, QtColorPicker.cpp, CameraManipulator.cpp



## Úlohy na budúce stretnutie:

Upratať nesprávne umiestnené veci z 3dsodtviz	Denis Illes	In Progress	doc - doxygen resources .bat delete logging ako modul - <a href="https://github.com/LuaDist/lualogging">https://github.com/LuaDist/lualogging</a>
Aruco example testy	Unassigned	To Do	vysúšať 3dsodtviz\install\bin\aruco_test_gl.exe a ostatné *_gl.exe arg. live camera.yml 0.025 freeglut.dll prekopiovať z "OSG\ThirdParty\VC10\86\bin" a premenovať na camera.yml prekopiovať z "_install\share\3dsodtviz\config"
Refaktoring časti: Kinect, Leap, OpenCV	Filip Jurcacko	In Progress	inšpirácia - Garaj - gitlib spraviť ako osobitne externé libky da sa samostatne skompilovať, netreba mu 3dsodtviz pozret aj CMakeLists.txt GitGraphImporter a GitGraphUpdater su interface, používajú 3dsodtviz a lib
OSG Jedno okno - rozťahnuté cez 3 monitory	Matus Cimerman	In Progress	
OSG Multiview	Denis Illes	In Progress	OSQ multiview/window - projekty virtualCave equalizer - skusit googlit tasky - analýza 3 okná - iná kamera, pre každé fullscreen.. pozrieť si example
Refaktoring nahradiť cudzie kody originalmi	Irina Dyomina	To Do	
Refaktoring - diluculum	Dalibor Meszaros	In Progress	diluculum použitý mimo namespace Lua (triedy LuaGraph, LuaInterface) riešenie: LuaGraph/Interface - pridať metódy
3Dconnexion Mouse - vytvoriť core class / interface	Michal Fasanek	In Progress	Vytvoriť triedy na spracovanie a konvertovanie údajov získaných z myšky, n funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvoriť interface, cez ktoré bude komunikovať modul (myš) s existujúcim p 3DSoftViz.
3Dconnexion Mouse - namapovať na ovládanie virtuálnej kamery	Michal Fasanek	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovlá režimoch - voľný a orbitálny pohyb.
Presunúť "cudzi" kod do dependencies	Peter Kapec	To Do	Zdrojové súbory: NoiseUtils.cpp, QColorPicker.cpp, CameraManipulator.cp
linkovať sa na osgModelling	Peter Kapec	Done	treba odstrániť z projektu súbory v Math/ Bezier, Curve, Export, Model, Utilities a linkovať sa na osgModeling knižnicu, ktorú treba pridať do dependencies a
Tlačidlo "&Multi-select mode" vracia cez "getid" NULL, čo zhodí program	Filip Jurcacko	Done	Pri označovaní viacerých uzlov program crashne.

## ***B.k Jedenásty týždeň***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 2.12.2015

**Prítomní:** Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Diskusia k prezentácii RTP,
  - Ako funguje riadenie tímu?
  - Ake problémy nastali s aplikovaním metodík v tíme?
  - Ako si plnia svoje manažerské roly členovia tímu?
- V tomto šprinte sa treba zamerať na dokočenie úloh a ich adekvátnu dokumentáciu v inžinierskom diele.
  - Dokončiť minimálne 3D myš a otvorený refactoring, ktorý bude možné dokončiť.
- Zhodli sme sa na tom, že budeme pracovať na projekte aj cez sviatky a skúškové obdobie aby sme projekt posunuli. Je to z viacerých dôvodov:
  - Projekt je veľký a trvalo nám dlho kým sme sa zoznámili s jeho všetkými súčasťami.
  - Chceme projekt niekam posunúť – implementovať obohatenú realitu s použitím 3D okuliarov, implementovanie CAVE zobrazovania.
- Diskusia k dokumentácii:
  - Dokumentácia by mala pre náš projekt predstavovať skôr technickú správu.
  - Mali by sme ju transformovať do stavu „in-progress“ tak, aby mohli v dokumente pokračovať aj ďalšie tímové projekty
  - Zhodli sme sa, že má zmysel dokumentovať aj rozumnú časť analýzy problému, napríklad pri implementácii nového modulu do projektu.
- Treba otestovať webkameru a sledovanie tváre.

## Stav úloh z minulého stretnutia:

Upratať nesprávne umiestnené veci z 3dsodtviz	Denis Illes	In Progress	doc - doxygen resources .bat delete logging ako modul - <a href="https://github.com/LuaDist/lualogging">https://github.com/LuaDist/lualogging</a>
Aruco example testy	Unassigned	To Do	vysúšať 3dsodtviz\_install\bin\aruco_test_gl.exe a ostatné *_gl.exe arg. live camera.yml 0.025 freeglut.dll prekopirovať z "OSG\ThirdParty\VC10\x86\bin" a premenovať na freeglut.dll camera.yml prekopirovať z "_install\share\3dsodtviz\config"
Refaktoring časti: Kinect, Leap, OpenCV	Filip Jurcacko	In Progress	inšpirácia - Garaj - gitlib spraviť ako osobitne externé libky da sa samostatne skompilovať, netreba mu 3dsodtviz pozrieť aj CMakeLists.txt GitGraphImporter a GitGraphUpdater su interface, používajú 3dsodtviz a lib
OSG Jedno okno - rozťahnuté cez 3 monitory	Matus Cimerman	In Progress	
OSG Multiview	Denis Illes	In Progress	OSQ multiview/window - projektor virtualCave equalizer - skusit googlit tasky - analýza 3 okná - iná kamera, pre každé fullscreen.. pozrieť si example
Refaktoring nahradit' cudzie kody originalmi	Irina Dyomina	To Do	
Refaktoring - diluculum	Dailbor Meszaros	In Progress	diluculum použitý mimo namespace Lua (triedy LuaGraph, LuaInterface) riešenie: LuaGraph/Interface - pridať metódy
3Dconnexion Mouse - vytvorit' core class / interface	Michal Fasaneck	In Progress	Vytvorit' triedy na spracovanie a konvertovanie údajov získaných z myšky, napisať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvorit' interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSodtviz.
3Dconnexion Mouse - namapovať na ovládanie virtuálnej kamery	Michal Fasaneck	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovládanie v oboch režimoch - voľný a orbitálny pohyb.
Presunúť "cudzí" kód do dependencies	Peter Kapec	To Do	Zdrojové súbory: NoiseUtils.cpp, QColorPicker.cpp, CameraManipulator.cpp
linkovať sa na osgModelling	Peter Kapec	Done	treba odstrániť z projektu súbory v Math/ Bezier, Curve, Export, Model, Utilities a linkovať sa na osgModeling knižnicu, ktorú treba pridať do dependencies ako submodule
Tlačidlo "&Multi-select mode" vracia cez "getid" NULL, čo zhodí program	Filip Jurcacko	Done	Pri označovaní viacerých uzlov program crashne.

## ***B.1 Dvanásty týždeň***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 9.12.2015

**Prítomní:** Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Posledné stretnutie v zimnom semestri.
- Zhodli sme sa, že budeme na projekte pracovať aj cez sviatky a skúškové obdobie aby sme projekt posunuli a dosiahli všetky ciele, ktoré sme si stanovili na začiatku projektu.
- Diskusia najmä o tom, prečo sa nám nepodarilo dokončiť posledné úlohy:
  - Riešili sme hlavne dokumentáciu,
  - Príprava na deň otvorených dverí na FIIT kde sme prezentovali 3Dsoftviz spolu s Kinect-om
  - Posledný šprint trval o týždeň menej ako ostatné šprinty – nedostatok času
  - Nedokončené úlohy ostávajú s cieľom dokončiť ich do Vianoc resp. cez vianočné sviatky

## Stav úloh z minulého stretnutia:

Upratať nesprávne umiestnené veci z 3dssoftviz	Denis Illes	In Progress	doc - doxygen resources .bat delete logging ako modul - <a href="https://github.com/LuaDist/lualogging">https://github.com/LuaDist/lualogging</a>
Aruco example testy	Unassigned	To Do	vysúšať 3dssoftviz\_install\bin\aruco_test_gl.exe a ostatné *_gl.exe arg. live camera.yml 0.025 freeglut.dll prekopiovať z "OSG\ThirdParty\VC10\x86\bin" a premenovať na freeglut.dll camera.yml prekopiovať z "_install\share\3dssoftviz\config"
Refaktoring časti: Kinect, Leap, OpenCV	Filip Jurcacko	In Progress	inšpirácia - Garaj - gitlib spraviť ako osobitne externé libky da sa samostatne skompilovať, netreba mu 3dssoftviz pozret aj CMakeLists.txt GitGraphImporter a GitGraphUpdater su interface, používajú 3dssoftviz a lib
OSG Jedno okno - rozťahnuté cez 3 monitory	Matus Cimerman	In Progress	
OSG Multiview	Denis Illes	In Progress	OSQ multiview/window - projektor virtualCave equalizer - skusit googlit tasky - analýza 3 okná - iná kamera, pre každé fullscreen.. pozrieť si example
Refaktoring nahradiť cudzie kody originalmi	Irina Dyomina	To Do	
Refaktoring - diluculum	Dalibor Meszaros	In Progress	diluculum použitý mimo namespace Lua (triedy LuaGraph, LuaInterface) riešenie: LuaGraph/Interface - pridať metódy
3Dconnexion Mouse - vytvoriť core class / interface	Michal Fasanek	In Progress	Vytvoriť triedy na spracovanie a konvertovanie údajov získaných z myšky, napísať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvoriť interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSoftViz.
3Dconnexion Mouse - namapovať na ovládanie virtuálnej kamery	Michal Fasanek	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovládanie v oboch režimoch - voľný a orbitálny pohyb.
Presunúť "cudzí" kod do dependencies	Peter Kapec	To Do	Zdrojové súbory: NoiseUtils.cpp, QColorPicker.cpp, CameraManipulator.cpp
linkovať sa na osgModelling	Peter Kapec	Done	treba odstrániť z projektu súbory v Math/ Bezier, Curve, Export, Model, Utilities a linkovať sa na osgModeling knižnicu, ktorú treba pridať do dependencies ako submodule
Tlačidlo "&Multi-select mode" vracia cez "getfd" NULL, čo zhodí program	Filip Jurcacko	Done	Pri označovaní viacerých uzlov program crashne.

## Úlohy na budúce stretnutie:

Upratať nesprávne umiestnené veci z 3dssoftviz	Denis Illes	In Progress	doc - doxygen resources .bat delete logging ako modul - <a href="https://github.com/LuaDist/lualogging">https://github.com/LuaDist/lualogging</a>
Aruco example testy	Unassigned	To Do	vysúšať 3dssoftviz\_install\bin\aruco_test_gl.exe a ostatné *_gl.exe arg. live camera.yml 0.025 freeglut.dll prekopiovať z "OSG\ThirdParty\VC10\x86\bin" a premenovať na freeglut.dll camera.yml prekopiovať z "_install\share\3dssoftviz\config"
Refaktoring časti: Kinect, Leap, OpenCV	Filip Jurcacko	In Progress	inšpirácia - Garaj - gitlib spraviť ako osobitne externé libky da sa samostatne skompilovať, netreba mu 3dssoftviz pozret aj CMakeLists.txt GitGraphImporter a GitGraphUpdater su interface, používajú 3dssoftviz a lib
OSG Jedno okno - rozťahnuté cez 3 monitory	Matus Cimerman	In Progress	
OSG Multiview	Denis Illes	In Progress	OSQ multiview/window - projekory virtualCave equalizer - skusit googlit tasky - analýza 3 okná - iná kamera, pre každé fullscreen.. pozrieť si example
Refaktoring nahraďiť cudzie kody originalmi	Irina Dyomina	To Do	
Refaktoring - diluculum	Dalibor Meszaros	In Progress	diluculum použitý mimo namespace Lua (triedy LuaGraph, LuaInterface) riešenie: LuaGraph/Interface - pridať metódy
3Dconnexion Mouse - vytvoriť core class / interface	Michal Fasanek	In Progress	Vytvoriť triedy na spracovanie a konvertovanie údajov získaných z myšky, napísať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvoriť interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSoftViz.
3Dconnexion Mouse - namapovať na ovládanie virtuálnej kamery	Michal Fasanek	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovládanie v oboch režimoch - voľný a orbitálny pohyb.
Presunúť "cudzí" kod do dependencies	Peter Kapec	To Do	Zdrojové súbory: NoiseUtils.cpp, QColorPicker.cpp, CameraManipulator.cpp
linkovať sa na osgModelling	Peter Kapec	Done	treba odstrániť z projektu súbory v Math/ Bezier, Curve, Export, Model, Utilities a linkovať sa na osgModeling knižnicu, ktoru treba pridať do dependencies ako submodule
Tlačidlo "&Multi-select mode" vracia cez "getfd" NULL, čo zhodí program	Filip Jurcacko	Done	Pri označovaní viacerých uzlov program crashne.

## ***B.m Prvý týždeň LS***

**Autor:** Dalibor Mészáros

**Dátum stretnutia:** 18.2.2016

### **Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Diskutované témy:**

- Pripravenie tímového počítača
  - Presun dát pôvodných vlastníkov
  - Skontrolovať ovládače, správnu inštaláciu driverov (win10)
  - Sfunkčniť projekt 3DSoftViz, nainštalovať potrebné komponenty
  - Pripravené Nvidia okuliare
    - Použiteľné pre stereo videnie
- Zčať realizovať projekciu na plátna
- Vuzix
  - Vyskúšať
  - Zakomponovať do projektu 3DSoftViz
- Úlohy s najvyššou prioritou:
  - Sfinalizovať prácu na jednotlivých vetvách a vykonať merge do develop-u
  - Synchronizovať/Mergnúť vetvu od Garaja (Najvyššia priorita)
  - Synchronizovať/Mergnúť vetvu od Juríka
  - Zvoliť a začleniť logovaciu aplikáciu
  - Vyriešiť objavené problémy s výkonom
    - Zvoliť aplikáciu a zabezpečiť kontrolu výkonu ((micro)benchmark)
  - Unit testy (gitlib)
  -

- Skompilovať example
- Smartgit – aplikácia pre správu lokálnych repozitárov
- FileSettingsHelp – pridať možnosť otestovať funkcionality jedným tlačidlom, alebo argumentom
  - Jedno kliknutie načíta test graf - veoliu - menší graf
  - Load graf – ./\_install/lib/lua/leg
- Rýchle prepínanie pozadí za behu programu
  - Miesto skyboxu biele, čierne pozadie
  - Nastaviť prepínanie na klávesy
  - Momentálne funguje pomocou konfiguračného súboru
- Či sa dá miesto skyboxu zobrazit' videostream?
  - Live video, napr. z kamery, kinect
- Juríkova vetva
  - Bol pridaný zdroj svetla a tieňovanie
  - Podlaha
- Kanban
  - Ready, Story
  - Komentý, Unit testy, Merged,
  - Zmenit' scrum na kanban



## ***B.n Druhý týždeň LS***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 25.2.2016

### **Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Riesenie problemov s integraciou easylogging do projektu
  - Na WIN problem kvoli MSVC2010 kompilatoru a podpore C++11. Skusit skompilovat Qt a dalsie potrebne moduly/kniznice na MSVC2013
- Diskusia o multiwindow/view zobrazeni
  - Equalizer – velky project, ktory bude pravdepodobne velmi narocne zaradit do 3Dsoftviz-u
  - Preskumat moznosti riesenia priamo v OSG
- Prestudovat igoo testovaci kniznicu pre BDD testy
  - Kazdy novy kod musi mat testy a logovanie
- Testovanie Vuzix okuliarov
  - SDK pravdepodobne nebude fungovat
  - Len ako zobrazovanie, analogia monitoru
- Pozriet sa na Leap na MacOSX
- OSX:
  - Instalacia OpenNI2, OpenCV
- NITE kompilovat zdrojove kody, ktore su google drive

Presun zvysnych veci na google drive

## ***B.o Tretí týždeň LS***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 3.3.2016

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

**Priebeh stretnutia:**

- Riešenie problémov s integráciou easylogging do projektu
  - Na WIN problém kvôli MSVC2010 kompilátoru a podpore C++11. Skúsiť skompilovať Qt a ďalšie potrebné moduly/knižnice na MSVC2013
- MSVC2013 kompilátor nie je prioritou – nie je to blocking point pre ďalšiu prácu a 3D mys ako sme si mysleli.
- Diskutovali sme o tom ako rozťahnúť okno cez tri monitory:
  - QT aplikácia resp. window je ako wrapper nad celou aplikáciou,
  - Vnorený je OpenGL canvas, ktorý ale nevidí svoje okolie,
  - Keď dáme v aplikácii full-screen mode – opengl sa rozťahne,
  - Treba nastaviť správny uhol kamier.
  - Ako by sa toto dalo riešiť s OSG?
- Analýza použitia a začlenenie Leap senzoru pre Mac OSX.
- Overiť implementáciu a následne otestovať OpenCV + Kinect na Mac OSX.
  - Skontrolovať find Cmake súbor.
  - Použiť homebrew.
- Skompilovať zdrojové kódy NiTE pre OSX a Unix.
- Diskusia o 3D myši a jej implementácii na ďalších platformách.

Riešili sme problémy s kompiláciou na OSX.

## ***B.p Štvrtý týždeň LS***

**Autor:** Michal Fašánek

**Dátum stretnutia:** 10.3.2016

### **Prítomní:**

Vedúci:            Ing. Peter Kapec, PhD.

Členovia tímu:        Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Doplnenie nového pozadia do 3dSoftviz
  - Analýza problému
- 3Dmouse
  - SDK pre Windows nekompatibilné s Qt
  - Možnosť použitia reimplementácie SDK pre Qt
  - Diskusia o potrebe SDK pre iné platformy
- Unit testy
  - Skúmanie možností prístupu pri implementácií
- Refaktoring do podoby knižníc
  - Dynamické vs statické knižnice
  - V akom stave sú jednotlivé moduly
- Diskusia o migrácií – Qt5 a VS2013
  - Analýza dostupných riešení pre migrovanie
  - Analýza dopadu na project
- Plánovanie pre merge jednotlivých vetiev aj mimo projektu

Analýza dôležitosti jednotlivých vetiev

## ***B.q Piaty týždeň LS***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 17.3.2016

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

**Priebeh stretnutia:**

- Diskutovali sme o práci ďalších študentov na projekte:
  - Do konca týždňa merge všetkých ostatných projektov do develop-u a neskôr master-a nášho projektu.
- Do inštalačného manuálu pridať *git clone –recursive*, ktorý zabezpečí načítanie všetkých submodulov počas klonovania projektu.
- Otestovať Kinect na OSX a Linuxe s novým Kinect senzorom, ktorý by mal byť nezávislý od platformy.
- Pridať do inštalačnej príručky OSX špecifické detaily, resp. problémy s ktorými sme sa stretli.
- Diskutovali sme o funkčnom MSVC 2013 pre Windows.
- Analýza zlého natáčania grafu v scéne pri ovládaní pohybom hlavy používateľa:
  - Face recognition s Aruco nefunguje správne.
- Pridať hľadanie SDK a ovládačov 3D myši na OSX platforme.
- Vuzix:
  - Prvá implementácia odprezentovaná vlastníkovy produktu.
  - Treba dokončiť a otestovať implementáciu + rozpoznávanie markerov.

Vuzix funkčne len na Windows – neexistuje podpora na ostatne platformy (možno bude použiteľné len ako zobrazovacia jednotka cez HDMI kábel so stereo zobrazením)

## ***B.r Šiesty týždeň LS***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 31.3.2016

### **Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Riešenie a diskusia k problémom, ktoré vznikli merge-ovaním projektov do nášho projektu.
  - Nefunkčný Leap senzor na OSX – potrebné vyriešiť a opraviť.
  - OpenNi2 a NiTE tiež spôsobuje problémy s linkovaním projektu na OSX.
  - Potrebné opravy pre Windows – done.
- Ovládanie scény pohybom hlavy – face recognition funguje správne na všetkých platformách:
  - Potrebné je správne nastaviť aplikáciu v GUI – zaradiť do používateľskej príručky.
- 3D myš:
  - SDK a ovládače sa správne linkujú na OSX. Vyriešiť implementáciu na OSX.
  - SDK a ovládače na linux: linkovať + implementácia.
  - wrapper pre separátne platformy.
- Vyriešiť problémy a nefunkčný Leap senzor na OSX.
- Diskutovali sme problémy, ktoré vznikli pri implementácií pozadia.
- Roztiahnutie aplikácie cez tri monitory funguje správne. Otestované a implementované. Analógia „CAVE“ pravdepodobne nebude možné jednoducho implementovať v tomto projekte.
- Funkčný projekt pre Qt4 aj Qt5.
- Vuzix okuliare:
  - Funkčná a otestovaná implementácia.
  - Funguje tiež marker recognition cez kameru okuliarov – zlá orientácia, ktorú je potrebné opraviť.
- Vygenerovaná dokumentácia v nástroji Sphinx v HTML a ďalších formátoch.
  - Vytvoriť automatické generovanie podľa zdrojového kódu.

Začleniť do Cmake.

## ***B.s Siedmy týždeň LS***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 7.4.2016

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

**Priebeh stretnutia:**

- Riešenie problémov s 3D myšou - implementácia v projekte VTK pre Unix aj pre MacOSX.
- Vyriešili sme problémy s kompiláciou Leap na OSX - problém s includovaním Windows only knižnice na všetkých platformách.
  - Pravdepodobne bude treba zmeniť úplne implementáciu Leap-u, pretože aktuálne nie sú detekované žiadne gestá, ktoré sú podporované SDK.
- Diskutovali sme možnosti implementácie Leap Orion do nášeho projektu.
- Analyzovať možnosti webhooks na develop repozitár na projekt na Github.

Pripraviť metodiku na písanie logov a testov.

## ***B.t Ósmy týždeň LS***

**Autor:** Matúš Cimerman

**Dátum stretnutia:** 14.4.2016

### **Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- Riešenie problémov s 3D myšou - implementácia v projekte VTK pre Unix aj pre MacOSX.
- Vyriešili sme problémy s kompiláciou Leap na OSX - problém s includovaním Windows only knižnice na všetkých platformách.
  - Pravdepodobne bude treba zmeniť úplne implementáciu Leap-u, pretože aktuálne nie sú detekované žiadne gestá, ktoré sú podporované SDK.
- Diskutovali sme možnosti implementácie Leap Orion do nášeho projektu.
- Analyzovať možnosti webhooks na develop repozitár na projekt na Github.
- Pripraviť metodiku na písanie logov a testov.

***B.u Deviaty týždeň LS***



## ***B.v Desiaty týždeň LS***

**Autor:** Michal Fašánek

**Dátum stretnutia:** 21.4.2016

**Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia**

- 3Dmouse
  - Vyhľadanie problému s threadmi a posielaním signálu pre Linux
- Leap
  - Manipulácia s demom pomocou rúk
- Vuzix
  - Integrácia stereoskopického videnia pomocou premietania
- Kinect, Leap
  - Testovanie pre MAC

## ***B.w Jedenásty týždeň LS***

**Autor:** Michal Fašánek

**Dátum stretnutia:** 28.4.2016

### **Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- 3Dmouse
  - Kalibrácia a testovanie pre Linux
  - Testovanie demo pre OSX
- Testovanie pred zlučovaním vetiev
  - Garaj – vizualizácia verziovania
  - Jurík – Obohatená realita
- Vuzix
  - Dokončenie testovania pre mapovanie na značkovač
- Sphinx

zaradenie do Sphinx systému

## ***B.x Dvanásty týždeň LS***

**Autor:** Michal Fašánek

**Dátum stretnutia:** 5.5. 2016

### **Prítomní:**

Vedúci: Ing. Peter Kapec, PhD.

Členovia tímu: Bc. Matúš Cimerman

Bc. Irina Dyomina

Bc. Michal Fašánek

Bc. Jaroslav Gazdík

Bc. Denis Illés

Bc. Filip Jurčacko

Bc. Dalibor Mészáros

### **Priebeh stretnutia:**

- 3Dmouse
  - Vývojové diagramy
  - Technická dokumentácia
  - Finalizácia programového kódu
- Diskusia o dlhodobom smerovaní projektu
- Vyhodnotenie tímového projektu
- Tvorba tímovej dokumentácie
- Doxygen
  - Zaradenie nástroja pre automatickú tvorbu dokumentácie z programovéh kódu do build systému

## Príloha C Exporty úloh z Jiry

### C.a Úlohy z prvého šprintu

Issue Type	Summary	Assignee	Status	Description	Epic Name	Labels
Sub-task	AUGREALITY-5 Aktualizovať OSG 3.4	Denis Illes	Done			implement
Sub-task	AUGREALITY-5 Aktualizovať Aruco	Peter Kapec	Done			implement
Sub-task	AUGREALITY-21 Oboznámiť sa s cppLint	<i>Unassigned</i>	Done			implement
Task	Nastaviť editor v QtCreator	<i>Unassigned</i>	Done	Tab Policy - Tab Only Tab Size - 4, Indent Size - 4 Align Continuation Lines - Not at all		other
Sub-task	AUGREALITY-21 Oboznámiť sa s cppCheck	<i>Unassigned</i>	Done			implement
Task	Vyskúšať nástroje pre C++	<i>Unassigned</i>	Done	Nástroje cppCheck a cppLint		implement

Bug	Tlačidlo "Load function calls" po otvorení priečinka zhodí program	Denis Illes	Done	Pri otváraní zložky s .lua súbormi program crashne. Zrejme Windows related issue, keďže na iOS funguje bez akýchkoľvek problémov. Riešenie: zakomentovať ...\\3dsoftviz\\resources\\scripts\\luadb\\extraction\\extractor.lua line 182 (assert) alebo zmazať druhú podmienku > assert(path, "wrong path passed")		implement
Bug	Tlačidlo "&Multi-select mode" vracia cez "getId" NULL, čo zhodí program	Filip Jurcacko	Ready to review	Pri označovaní viacerých uzlov program crashne.		implement
Task	Includnúť chýbajúce headre pre non-unity build	Denis Illes	Done			implement
Task	Opraviť warningy	Dalibor Meszaros	Done			implement
Sub-task	AUGREALITY-7 Pridať na web zázpisnice z 2-4w	Matus Cimerman	Done			other
Task	Šablóna pre zázpisnice zo strenutí	Jaroslav Gazdik	Done	Vytvorit gdoc sablonu pre zapisnice zo strenuti - pre web.		other
Task	Aktualizovať knižnice v projekte	<i>Unassigned</i>	Done	Knižnice Aruco, OpenSceneGraph 3.4		implement

Task	Nastaviť JIRU podľa potreby	Denis Illes	Done			other
Task	Upraviť inštalačný manuál	<i>Unassigned</i>	Done			other
Task	Nahrať dokumenty na webovú stránku	Michal Fasanek	Done			other
Task	Dokončiť webovú stránku	Michal Fasanek	Done			other

**C.b Úlohy z druhého šprintu**

<b>Issue Type</b>	<b>Summary</b>	<b>Assignee</b>	<b>Status</b>	<b>Description</b>	<b>Epic Name</b>	<b>Labels</b>
Task	Zapisnica w8	Michal Fasanek	Done	V prilohe export taskov z Jiry. w8-1_status - status uloh z minuleho tyzdna w8_todo-inprogress - zvsne ulohy na buduci tyzden		other
Task	Gitflow metodiku upraviť	Jaroslav Gazdik	Done	Pill requesty a nič viac...		other
Task	Dokumentácia k produktu - riadenie	Filip Jurcacko	To Do	BIELIKOVÁ <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html">http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html</a>		other
Task	Dokumentácia k produktu - BigPicture	Filip Jurcacko	To Do	BIELIKOVÁ <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html">http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html</a>		other
Task	Dokumentácia k produktu - 3 šprinty	Filip Jurcacko	To Do	do 18.11. v AIS		other
Task	Úprava CodeReview metodiky	Irina Dymina	Done	do 19.11 do AIS - BIELIKOVÁ		other
Task	Upload dokumentov na stránku	Michal Fasanek	Done			other
Task	Zápisnica W7	Matus Cimerman	Done			other

Task	Vytvorit metodiku - ako programovat v C++	Matus Cimerman	Done	Vytvorit dokument, v ktorom sa opíše postup pri programovaní v jazyku cpp. Zaznamenajú sa základné princípy, ktorých sa budeme držať pri vývoji softvéru. * astyle, * cpplint, * cppcheck, * spojit s txt na drope.		other
Task	3Dconnexion Mouse - vytvorit core class / interface	Jaroslav Gazdik	In Progress	Vytvorit triedy na spracovanie a konvertovanie údajov získaných z myšky, napísať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvorit interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSoftViz.		implement
Task	3Dconnexion Mouse - vytvorit modul v projekte	Jaroslav Gazdik	Done	Pripojiť SDK k projektu - nájsť knižnice, v ktorých sú použité funkcie, začleniť do adresára projektu, nastaviť cesty v CMakeLists.		implement
Task	3Dconnexion Mouse - vytvorit diagramy	Irina Dymina	In Progress	Vytvorit diagram s triedami znázorňujúce začlenenie modulu do projektu. Inšpirovať sa s modulom Kinect.		implement
Bug	Opraviť warningy - MSVC	Dalibor Meszaros	Done	Warningy detekované nástrojom MSVC		implement
Bug	Opraviť warningy - CppLint	Matus Cimerman	Done	Warningy detekované nástrojom cppLint		implement
Bug	Opraviť warningy - cppCheck	Dalibor Meszaros	Done	Warningy detekované nástrojom cppCheck		implement



Task	Vytvorit' dok. s retrospektívou	Dalibor Meszaros	Done	Zanalyzovat' 2. šprint, prekonzultovat' s tímom, vyzdvihnúť kladné/záporné stránky šprintu, navrhnúť vylepšenia, zahodiť nefungujúce spôsoby, spísať do dokumentu a odovzdať na upload.		other
Bug	Tlačidlo "Load function calls" po otvorení priečinka zhodí program	Denis Illes	Done	Pri otváraní zložky s .lua súbormi program crashne. Zrejme Windows related issue, keďže na iOS funguje bez akýchkoľvek problémov.  Riešenie: zakomentovať ...\3dsoftviz\Resources\scripts\luadb\extraction\extractor.lua line 182 (assert) alebo zmazať druhú podmienku > assert(path, "wrong path passed")		implement
Bug	Tlačidlo "&Multi-select mode" vracia cez "getId" NULL, čo zhodí program	Filip Jurcacko	Ready to review	Pri označovaní viacerých uzlov program crashne.		implement
Task	Aktualizovať používateľskú príručku	Denis Illes	To Do			other

*C.c Úlohy z tretieho šprintu*

Issue Type	Summary	Assignee	Status	Description	Epic Name	Labels
Task	Zapisnica w8	Michal Fasanek	Done	V prilohe export taskov z Jiry. w8-1_status - status uloh z minuleho tyzdna w8_todo-inprogress - zvsne ulohy na buduci tyzden		other
Task	Opravit error handling	Filip Jurcacko	Done	Na viacerych miestach v kode je do funkcii posielany argument bool* error ktory sa vnuri funkcie nastavuje v pripade chyby  Avsak casto po vykonani funkcii bud nieje skontrolovany vobec alebo je viacasobne prepisany viacerymi volaniami funkcie nez je skontrolovany.  Fix it.		implement
Task	Gitflow metodiku upraviť	Jaroslav Gazdik	Done	Pill requesty a nič viac...		other
Task	Dokumentácia k produktu - riadenie	Filip Jurcacko	To Do	BIELIKOVÁ <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html">http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html</a>		other
Task	Dokumentácia k produktu - BigPicture	Filip Jurcacko	To Do	BIELIKOVÁ <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html">http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html</a>		other
Task	Dokumentácia k produktu - 3 šprinty	Filip Jurcacko	To Do	do 18.11. v AIS		other
Task	Úprava CodeReview metodiky	Irina Dymina	Done	do 19.11 do AIS - BIELIKOVÁ		other

Task	Upload dokumentov na stránku	Michal Fasanek	Done			other
Task	Nájsť všetky cudzie kódy	Irina Dymina	Done	identifikovať všetky cudzie kódy, a nájsť odkiaľ boli zobrať napr: foldre: src > Noise (hlavička zobrazuje že kód je prevzatý) Viewer > Treeltem, TreeModel QOSG > qtcolorpicker OSGQTBrowser > WebViewImage LuaGraph > LuaGprahTreeltem		implement
Task	Zápisnica W7	Matus Cimerman	Done			other
Bug	Opraviť warningy - OS X	Matus Cimerman	In Progress			implement
Task	Začleniť triedy do príslušných namespaceov	Denis Illes	Done	Triedy ako "Cube", "Sphere", "SpikySphere" sa nachádzajú mimo svojho namespaceu (Clustering). Treba ich začleniť. Rovnako ako "HistoryBuffer" (malo by to byť v kinect namespace)		implement
Task	Zmazať zbytočné triedy a vyskúšať program	Dalibor Meszaros	Done	V projekte sa nachádzajú triedy "ransac" a "vector3", ktoré sú pravdepodobne zbytočné a nepoužívajú sa. Treba ich zmazať a vyskúšať fungovanie projektu bez týchto tried.		implement
Task	Vytvoriť metodiku - ako programovať v C++	Matus Cimerman	Done	Vytvoriť dokument, v ktorom sa opíše postup pri programovaní v jazyku cpp. Zaznamenajú sa základné princípy, ktorých sa budeme držať pri vývoji softvéru. * astyle, * cpplint, * cppcheck, * spojiť s txt na drope.		other

Task	3Dconnexion Mouse - vytvoriť core class / interface	Jaroslav Gazdik	In Progress	Vytvoriť triedy na spracovanie a konvertovanie údajov získaných z myšky, napísať funkcie, ktoré pohyby v jednotlivých smeroch spracuje a poskytne projektu. Príklad Core triedy - Kinect. Vytvoriť interface, cez ktoré bude komunikovať modul (myš) s existujúcim projektom 3DSoftViz.		implement
Task	3Dconnexion Mouse - namaľovať na ovládanie virtuálnej kamery	Michal Fasanek	In Progress	Poskytnuté údaje cez rozhranie použiť na ovládanie virtuálnej kamery. Ovládanie v oboch režimoch - voľný a orbitálny pohyb.		implement
Task	3Dconnexion Mouse - vytvoriť modul v projekte	Jaroslav Gazdik	Done	Pripojiť SDK k projektu - nájsť knižnice, v ktorých sú použité funkcie, začleniť do adresára projektu, nastaviť cesty v CMakeLists.		implement
Task	3Dconnexion Mouse - vytvoriť diagramy	Irina Dymina	In Progress	Vytvoriť diagram s triedami znázorňujúce začlenenie modulu do projektu. Inšpirovať sa s modulom Kinect.		implement
Bug	Opraviť warningy - MSVC	Dalibor Meszaros	Done	Warningy detekované nástrojom MSVC		implement
Bug	Opraviť warningy - CppLint	Matus Cimerman	Done	Warningy detekované nástrojom cppLint		implement
Bug	Opraviť warningy - cppCheck	Dalibor Meszaros	Done	Warningy detekované nástrojom cppCheck		implement

Task	Vytvorit' dok. s retrospektívou	Dalibor Meszaros	Done	Zanalyzovat' 2. šprint, prekonzultovat' s tímom, vyzdvihnúť kladné/záporné stránky šprintu, navrhnúť vylepšenia, zahodiť nefungujúce spôsoby, spísať do dokumentu a odovzdať na upload.		other
Task	linkovat' sa na osgModelling	Peter Kapec	Ready to review	treba odstranit z projektu subory v Math/- Bezier- Curve-Export - Model- Utilitiesa linkovat sa na osgModeling kniznicu, ktoru treba pridať do dependencies ako submodule		implement
Bug	Tlačidlo "Load function calls" po otvorení priečinka zhodí program	Denis Illes	Done	Pri otváraní zložky s .lua súbormi program crashne. Zrejme Windows related issue, keďže na iOS funguje bez akýchkoľvek problémov.  Riešenie: zakomentovať ...\\3dsoftviz\\resources\\scripts\\luadb\\extraction\\extractor.lua line 182 (assert) alebo zmazať druhú podmienku > assert(path, "wrong path passed")		implement
Bug	Tlačidlo "&Multi-select mode" vracia cez "getId" NULL, čo zhodí program	Filip Jurcacko	Ready to review	Pri označovaní viacerých uzlov program crashne.		implement
Task	Aktualizovať používateľskú príručku	Denis Illes	To Do			other

## C.d Úlohy zo štvrtého šprintu

T	Summary	Assignee	Status	Description	Epic Name	Labels
📌	Aktualizovat metodiku "Pisanie zdrojoveho kodu v C++"	Matus Cimerman	Done	▪ do 11.12. pred odovzdanim finalnej dokumentacie Rozsirt metodiku o poznatky z minulych rokov.		other
📌	Nahrat subory na stranku	Michal Fasanek	Done			other
📌	Prezentacia k riadeniu v TP na predmet MSI/MIS	Unassigned	Done	Viac info na - <a href="http://www2.fiit.stuba.sk/~simko/manazment/rtp-intro-2015.pdf">http://www2.fiit.stuba.sk/~simko/manazment/rtp-intro-2015.pdf</a>		other
📌	Dokumentacia k riadeniu	Unassigned	Done			other
📌	Retrospektiva 4. sprint	Matus Cimerman	Done			other
📌	Zapisnica w11	Matus Cimerman	Done			other
🔗	CAVE	Unassigned	To Do		CAVE (using Equalizer)	implement
📌	Inžinierske dielo	Unassigned	Done			other

## C.e Úlohy z piateho šprintu

<input checked="" type="checkbox"/>	Opravit error handling	Filip Jurcacko	<b>DONE</b>	Na viacerých miestach v kode je do funkcií posielaný argument bool* error ktorý sa vnútri funkcie nastavuje v prípade chyby Avšak často po vykonaní funkcie buď nie je skontrolovaný vôbec alebo je viacsobne prepísaný viacerými volaniami funkcie než je skontrolovaný. Fix it.	implement
<input checked="" type="checkbox"/>	Gitflow metodiku upraviť	Jaroslav Gazdik	<b>DONE</b>	Pill requesty a nič viac...	other ...
<input checked="" type="checkbox"/>	Dokumentácia k produktu - riadenie	Filip Jurcacko	<b>DONE</b>	BIELIKOVÁ <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html">http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html</a>	other
<input checked="" type="checkbox"/>	Dokumentácia k produktu - BigPicture	Filip Jurcacko	<b>DONE</b>	BIELIKOVÁ <a href="http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html">http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html</a>	other
<input checked="" type="checkbox"/>	Dokumentácia k produktu - 3 šprinty	Filip Jurcacko	<b>DONE</b>	do 18.11. v AIS	other
<input checked="" type="checkbox"/>	Úprava CodeReview metodiky	Irina Dyomina	<b>DONE</b>	do 19.11 do AIS - BIELIKOVÁ	other
<input checked="" type="checkbox"/>	Upload dokumentov na stránku	Michal Fasanek	<b>DONE</b>		other
<input checked="" type="checkbox"/>	Nájsť všetky cudzie kódy	Irina Dyomina	<b>DONE</b>	identifikovať všetky cudzie kódy, a nájsť odkiaľ boli zobrať napr: foldre: src > Noise (hlavička zobrazuje že kód je prevzatý) Viewer > TreeItem, TreeModel QOSG > qtcolorpicker OSGQTBrowser > WebViewImage LuaGraph > LuaGprahTreeItem	implement
<input checked="" type="checkbox"/>	Zápisnica W7	Matus Cimerman	<b>DONE</b>		other
<input checked="" type="checkbox"/>	Opraviť warningy - OS X	Matus Cimerman	<b>DONE</b>		implement

## C.f Úlohy zo šiesteho šprintu

1-3 of 3

Columns ▾

T	Summary	Assignee	Status	Description	Epic Name	Labels
<input checked="" type="checkbox"/>	Analýza logovacích knižníc	Jaroslav Gazdik	DONE	<p>pozrite:  <a href="https://github.com/SergiusTheBest/plog">https://github.com/SergiusTheBest/plog</a></p> <ul style="list-style-type: none"> <li>obsahuje linky na dalsie knizice - treba aj !!!!  <a href="https://github.com/gabime/spdlog">https://github.com/gabime/spdlog</a>  <a href="https://johnwbyrd.github.io/logog/index.html">https://johnwbyrd.github.io/logog/index.html</a>  <a href="https://github.com/easylogging/easyloggingpp/blob/master/README.md">https://github.com/easylogging/easyloggingpp/blob/master/README.md</a>  <a href="https://github.com/emilk/loguru">https://github.com/emilk/loguru</a></li> </ul> <p>analyzujte:</p> <ul style="list-style-type: none"> <li>sposob pisania logov</li> <li>ake features poskytuje</li> <li>ako je stara / ako casto sa aktualizuje</li> <li>ci je to header-only</li> <li>ako sa builduje (ci ma CMake)</li> <li>atd.</li> </ul> <p>vysledkom analyzy musi byt nejaky vystup (napr. tabulka porovania)            tak aby zostali informacie aj pre dalsich...</p>		implement
<input checked="" type="checkbox"/>	Multiple window view	Unassigned	TO DO		Multiview	implement other
<input checked="" type="checkbox"/>	GIT - Merge branches	Unassigned	DONE	<p>Priority:</p> <ol style="list-style-type: none"> <li>Garaj</li> <li>Jurik</li> </ol> <p>Otestovať a mergnúť podľa metodiky</p>		implement



## C.g Úlohy zo siedmeho šprintu

T	Summary	Assignee	Status	Description	Epic Name	Labels
<input checked="" type="checkbox"/>	MenuButton - Test	Jaroslav Gazdik	DONE	Pridať tlačítko TEST (file, setting, help, test) na rýchle otvorenie základných grafov <ul style="list-style-type: none"> <li>Jedno kliknutie načíta menší test graf (300-500 uzlový)</li> <li>Load graf - načíta cestu "install lib/lua/leg" (function call graph)</li> </ul>		implement ...
<input checked="" type="checkbox"/>	Analýza benchmarkových knižníc	Irina Dyomina	DONE	skuste analyzovat <a href="http://www.bfilipek.com/2016/01/micro-benchmarking-libraries-for-c.html">http://www.bfilipek.com/2016/01/micro-benchmarking-libraries-for-c.html</a> pretoze mame problemy aj s rychlostou... vysledkom analyzy musi byt nejaky vystup (napr. tabulka porovania) tak aby zostali informacie aj pre dalsich...		implement
<input checked="" type="checkbox"/>	Analýza logovacích knižníc	Jaroslav Gazdik	DONE	pozrite: <a href="https://github.com/SergiusTheBest/plog">https://github.com/SergiusTheBest/plog</a> <ul style="list-style-type: none"> <li>obsahuje linky na dalsie knizice - treba aj tie !!!!  <a href="https://github.com/gabime/spdlog">https://github.com/gabime/spdlog</a>  <a href="https://johnwbyrd.github.io/logog/index.html">https://johnwbyrd.github.io/logog/index.html</a>  <a href="https://github.com/easylogging/easyloggingpp/blob/master/README.md">https://github.com/easylogging/easyloggingpp/blob/master/README.md</a>  <a href="https://github.com/emilk/loguru">https://github.com/emilk/loguru</a></li> </ul> analyzujte: <ul style="list-style-type: none"> <li>sposob pisania logov</li> <li>ake features poskytuje</li> <li>ako je stara / ako casto sa aktualizuje</li> <li>ci je to header-only</li> <li>ako sa builduje (ci ma CMake)</li> <li>atd.</li> </ul> vysledkom analyzy musi byt nejaky vystup (napr. tabulka porovania) tak aby zostali informacie aj pre dalsich...		implement ...
<input checked="" type="checkbox"/>	Multiple window view	Unassigned	TO DO		Multiview	implement other
<input checked="" type="checkbox"/>	GIT - Merge branches	Unassigned	DONE	Priority: 1. Garaj 2. Jurik Otestovať a mergnúť podľa metodiky		implement
<input checked="" type="checkbox"/>	Vyskúšať okuliare VUZIX	Unassigned	DONE	Správne zapojiť a vyskúšať okuliare na nejakých examploch		implement

## C.h Úlohy z ôsmeho šprintu

☑ 3D mouse on OSX	Matus Cimerman	JUST STARTED	Working 3D mouse on OSX inspiration: VTK project
☑ 3D mouse SDK find Cmake - OSX	Matus Cimerman	DONE	<ul style="list-style-type: none"> <li>create cmake file to find 3D mouse SDK on OSX platform</li> </ul>
☑ 3D mouse - rework	Michal Fasaneck	ALMOST FINISHED	<ul style="list-style-type: none"> <li>multiplatform rework from win32 specific</li> </ul>
☑ 3D mouse lib	Michal Fasaneck	ALMOST FINISHED	<ul style="list-style-type: none"> <li>platform independent,</li> <li>3dmouse as a lib.</li> </ul>
🔗 Merge - garaj, jurik, open pull-request	Matus Cimerman	DONE	<ul style="list-style-type: none"> <li>open pull-request from 3dsoftviz</li> <li>garaj develop</li> <li>jurik develop</li> </ul>
☑ Face recognition wrong orientation	Matus Cimerman	DONE	<p>Face recognition using Aruco module</p> <ul style="list-style-type: none"> <li>wrong orientation (left - right, up - down) when moving head</li> </ul>
☑ Video stream namiesto skyboxu	Denis Illes	DONE	Zistiť či sa dá nahradiť statický skybox nejakým video streamom. napr. webcam, kinect,...
☑ Prepínanie pozadia	Matus Cimerman	DONE	Momentálne sa dá nastavovať pozadie (čierne, biele, skybox) iba v config súbore (..\3dsoftviz\resources\config)cc Umožniť prepínanie pomocou kláves.
☑ MenuButton - Test	Jaroslav Gazdik	DONE	<p>Pridať tlačítko TEST (file, setting, help, test) na rýchle otvorenie základných grafov</p> <ul style="list-style-type: none"> <li>Jedno kliknutie načíta menší test graf (300-500 uzlový)</li> <li>Load graf - načíta cestu "install lib/lua/leg" (function call graph)</li> </ul>
☑ Analýza benchmarkových knižníc	Irina Dyomina	DONE	<p>skuste analyzovať</p> <p><a href="http://www.bfilipek.com/2016/01/micro-benchmarking-libraries-for-c.html">http://www.bfilipek.com/2016/01/micro-benchmarking-libraries-for-c.html</a></p> <p>pretože máme problémy aj s rýchlosťou...</p> <p>výsledkom analýzy musí byť nejaký výstup (napr. tabuľka porovnaní)</p> <p>tak aby zostali informácie aj pre ďalších...</p>

## C.i Úlohy z deviateho šprintu

T	Summary	Assignee	Status	Description
<input checked="" type="checkbox"/>	Kinect - gestures doesn't work on OSX	Matus Cimerman	ANALYZING	Kinect is recognized but gestures doesn't appear to work.
<input checked="" type="checkbox"/>	Webhooks	Matus Cimerman	ANALYZING	Analyza ako by sme mohli pouzit webhooky na generovanie dokumentacie z githubu
<input checked="" type="checkbox"/>	<a href="#">Doxygen in CmakeLists</a>	Matus Cimerman	REVIEWING	build doxygen docu using cmake in project
<input checked="" type="checkbox"/>	Metodika - logovanie (easylogging)	Matus Cimerman	ANALYZING	Vytvorit metodiku pre pouzitie easylogging v nasom projekte - opis vsetkych moznosti kniznice a kedy ich sprav
<input checked="" type="checkbox"/>	Zapisnice weeks 1-2, 3-8	Matus Cimerman	DONE	
<input checked="" type="checkbox"/>	Crash pri load graph	Filip Jurcacko	DONE	Ak je load graph okno zavrete bez vybratia suboru, alebo je vybrany neexistujuci subor (manualnym zadanim ce Treba pridat kontrolu vstupneho suboru
<input checked="" type="checkbox"/>	Akceptacne testy	Irina Dyomina	DONE	
<input checked="" type="checkbox"/>	Sphinx dokumentacia - metodika	Irina Dyomina	DONE	
<input checked="" type="checkbox"/>	Dokumentacia - Excel import	Irina Dyomina	DONE	
<input checked="" type="checkbox"/>	Sphinx dokumentacia - PlantUML	Irina Dyomina	DONE	

## C.j Úlohy z desiateho šprintu

# Príloha D Retrospektívy šprintov

## *D.a Po prvom šprinte*

### **Retrospektíva 1. Šprintu**

*Tím #4 - alphaReach*

---

#### **Čo sa nám podarilo počas šprintu?**

- Zlepšiť komunikáciu v tíme.

#### **Čo nešlo podľa našich predstáv?**

- Práca na úlohách.
- Pridelovanie úloh riešiteľom.
- Ohodnocovanie úloh story pointami.
- Automatizácia najzákladnejších úloh, napríklad zápisnice na webe ihneď po stretnutí.
- Viditeľnosť projektu – nedostatočná komunikácia s product ownerom o stave projektu.
- Na konci šprintu nebol dodaný fungujúci inkrement.
- Práca s Jirou – vykazovanie stavu v akom sú úlohy.
- Identifikované úlohy boli nedostatočne popísané a dekomponované.

#### **Čo môžeme zmeniť a zlepšiť do budúceho šprintu?**

- Automatizovať najzákladnejšie úlohy.
- Zlepšiť prácu s Jirou a tým aj vykazovanie stavu v akom je projekt a šprint.
- Špecifickejšie zapisovanie úloh do Jiry.
- Ešte zlepšiť komunikáciu v tíme a s product ownerom.

## ***D.b Po druhom šprinte***

### **Retrospektíva 2. Šprintu**

*Tím #4 - alphaReach*

---

#### **Čo sa nám podarilo počas šprintu**

- Dokončili sme všetky dokumentačné úlohy a dokázali sme posunúť väčšinu implementačných úloh do revidáčnej fázy

#### **Čo nešlo podľa našich predstáv**

- Ani jedna implementačná úloha neprešla revidovaním kódu a nebola dokončená
- V tíme bola nedostatočná komunikácia medzi jednotlivými členmi, skupinou a vlastníkom produktu
- Nedostatočná komunikácia zapríčinila, že:
  - Nedohodli sme sa na tom, kto bude vykonávať revidovanie kódu, preto úlohy neprešli revidovaním do sekcie dokončených úloh
  - Neskoro sme sa začali pýtať vlastníka produktu a vedúceho na rady, ako postupovať pri riešení úloh

#### **Čo môžeme zmeniť a zlepšiť do budúceho šprintu**

- Uvedomili sme si, že by sme sa mali aktívne pýtať vlastníka produktu a vedúceho, kedykoľvek nevieme jednoznačne určiť, ako postupovať v riešení úloh
- Nebudeme si odkladať riešenie úloh na poslednú chvíľu, aby sa stihli zrevidovať

## *D.c Po tret'om šprinte*

### **Retrospektíva 3. Šprintu**

*Tím #4 - alphaReach*

---

#### **Čo sa nám podarilo počas šprintu?**

- Úspešne dokončiť takmer všetky implementačné úlohy a teda robiť prehliadky kódu ostatnými členmi tímu.
- Automatizovať rutinné procesy (zápisnice, retrospektíva).
- Zlepšili sme komunikáciu v tíme a tiež s vlastníkom produktu.
- Včas sme komunikovali nejasnosti k úlohám a stavu produktu s vlastníkom produktu.

#### **Čo nešlo podľa našich predstáv?**

- Procesy, ktoré sme až teraz začali používať aktívne:
  - Tvorba dokumentácie nebola zvládnutá z pohľadu riadenia a manažmentu.
- Burndown chart:
  - Skokové zmeny, ktoré implikujú, že sa nepracuje priebežne.
  - Nízka, resp. pomalá „rýchlosť“ (velocity) tímu.

#### **Čo môžeme zmeniť a zlepšiť do budúceho šprintu?**

- Zlepšiť proces tvorby dokumentácie:
  - Priebežná tvorba dokumentácie.
- Zefektívniť prácu na úlohách.