

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

# 3D UML

Dokumentácia k riadeniu projektu

**Vedúci tímu:** Ing. Ivan Polášek Phd.

**Členovia tímu:** Bc. Adam Kulíšek, Bc. Matej Jenis, Bc. Rami Mtier, Bc. Tomas Hnojčík, Bc. Patrik Kolek, Bc. Boris Buček

**Akademický rok:** 2015/2016

- [1 Úvod](#)
- [2 Podiel na vytváraní dokumentu](#)
- [3 Role členov projektového tímu](#)
- [4 Aplikácie manažmentov](#)
  - [4.1 Manažment kvality](#)
    - [4.1.1 Verifikácia a validácia procesov](#)
  - [4.2 Manažment plánovania](#)
    - [4.2.1 Priebeh plánovania v našom projekte](#)
  - [4.3 Manažment monitorovania](#)
  - [4.4 Manažér podpory vývoja a integrácie](#)
    - [4.4.1 Vetvenie zdrojového kódu](#)
  - [4.5 Manažment komunikácie](#)
  - [4.6 Manažment tvorby dokumentácie](#)
    - [4.6.1 Vytvárané dokumenty](#)
  - [4.7 Manažment rizík](#)
- [5 Sumarizácia šprintov](#)
  - [5.1 Úvodná fáza](#)
  - [5.2 Prvý šprint](#)
  - [5.3 Druhý šprint](#)
  - [5.4 Tretí šprint](#)
- [6 Používané metodiky](#)
  - [6.1 Manažment kvality](#)
    - [6.1.1 Metodika písania zdrojového kódu](#)
    - [6.1.2 Metodika tvorby tried a premenných](#)
    - [6.1.3 Zapuzdovanie a modifikátory prístupu](#)
    - [6.1.4 Metodika komentovania kódu](#)
    - [6.1.5 Metodika refaktorovania kódu](#)
    - [6.1.6 Metodika lexikálneho refaktorovania](#)
  - [6.2 Manažment plánovania](#)
    - [6.2.1 Plánovanie úloh na stretnutiach](#)
    - [6.2.2 Definition of Done](#)
    - [6.2.3 Metodiky v praxi](#)
  - [6.3 Manažment monitorovania](#)
    - [6.3.1 Organizácia stavu úloh v nástroji pre manažment úloh](#)
    - [6.3.2 Identifikácia zložitosti úloh](#)
    - [6.3.3 Postup pri vytváraní/pridelovaní úloh na začiatku šprintu](#)
    - [6.3.4 Postup pri vytváraní/pridelovaní úloh počas šprintu](#)
    - [6.3.5 Zhodnotenie použitej metodiky](#)
  - [6.4 Manažment podpory a vývoja a integrácie](#)
    - [6.4.1 Metodika pridávania novej funkcionality](#)
    - [6.4.2 Zásady pomenovania commitu](#)
  - [6.5 Manažment komunikácie](#)

- [6.5.1 Komunikačný kanál pre každodennú komunikáciu](#)
- [6.5.2 Zdieľanie dát](#)
- [6.5.3 Zhodnotenie dodržiavania metodík](#)
- [6.6 Manažment dokumentácie](#)
  - [6.6.1 Dokumentácia k riadeniu projektu a dokumentácia k produktu \(tri šprinty, big picture\)](#)
  - [6.6.2 Zápisnice zo stretnutí a dokumenty o retrospektíve](#)
  - [6.6.3 Inštalačná príručka a Dokument úloh priradený jednotlivým rolám](#)
  - [6.6.4 Zdieľanie dokumentov](#)
  - [6.6.5 Dostupnosť a zálohovanie dokumentov](#)
  - [6.6.6 Pomocné dokumenty](#)
- [6.7 Manažment rizík](#)
  - [6.7.1 Identifikácia rizík](#)
  - [6.7.2 Analýza rizík](#)
  - [6.7.3 Navrhovanie riešení](#)
  - [6.7.4 Riešenie neočakávaných rizík](#)
  - [6.7.5 Zhodnotenie metodiky](#)
- [7 Globálna retrospektíva zimného semestra](#)
  - [7.1 Retrospektíva šprintov](#)
    - [7.1.1 Retrospektíva druhého šprintu](#)
- [8 Export úloh počas šprintov](#)
  - [8.1 Úvodné úlohy](#)
  - [8.2 Šprint 1](#)
  - [8.3 Šprint 2](#)
  - [8.4 Šprint 3](#)
- [9 Tvorba stránky 3D Lab-u](#)
- [10 Prílohy](#)
  - [10.1 Zápisnica 1](#)
  - [10.2 Zápisnica 2](#)
  - [10.3 Zápisnica 3](#)
  - [10.4 Zápisnica 4](#)
  - [10.5 Zápisnica 5](#)
  - [10.6 Zápisnica 6](#)
  - [10.7 Zápisnica 7](#)
  - [10.8 Zápisnica 8](#)

# 1 Úvod

Tento dokument je vytvorený za účelom zdokumentovania riadenia nášho projektu na predmete tímový projekt. V dokumente je zosumarizované, ktorý člen sa podieľal na ktorej časti dokumentácie.

Každý člen nášho tímu má pridelenú svoju manažérsku rolu ktorú sa snaží v tomto dokumente popísať. Opisuje čomu sa daná rola venuje, aké metodiky sa skrývajú za touto rolou a ako sa darilo tieto metodiky uplatňovať počas práce na projekte.

V ďalšej časti tohoto dokumentu sa nachádza sumarizácia jednotlivých šprintov na projekte. Opisuje sa tu spôsob identifikácie jednotlivých šprintov a opis každého šprintu ktorý prebehol počas práce na projekte.

Používané metodiky sú zhrnuté na jednom mieste v dokumentu. Tieto metodiky sú výsledkom plnenia si manažérskych rolí. Opisované metodiky sa nám vo veľkej miere podarilo úspešne začleniť do projektu.

V závere dokumentu je vypracovaná globálna retrospektíva, ktorá sumarizuje jednotlivé šprinty a celkovú prácu na projekte.

## 2 Podiel na vytváraní dokumentu

Názov kapitoly	Autor
Úvod	Bc. Patrik Kolek
Role členov tímu	všetci členovia tímu
Aplikácie manažmentov	všetci členovia tímu
Sumarizácia šprintov	všetci členovia tímu
Používané metodiky: Manažment kvality	Bc. Matej Jenis
Používané metodiky: Manažment plánovania	Bc. Patrik Kolek
Používané metodiky: Manažment monitorovania	Bc. Tomáš Hnojčík
Používané metodiky: Manažment podpory vývoja a integrácie	Bc. Matej Jenis, Bc. Adam Kulíšek
Používané metodiky: Manažment komunikácie	Bc. Adam Kulíšek
Používané metodiky: Manažment tvorby dokumentácie	Bc. Rami Mtier
Používané metodiky: Manažment rizík	Bc. Boris Buček
Globálna retrospektíva zimného semestra	Bc. Adam Kulíšek
Export úloh počas šprintov	všetci členovia tímu
Tvorba stránky 3D Lab-u	Bc. Adam Kulíšek
Preberacie protokoly	celý tím

Tab. 1: Autori jednotlivých kapitol dokumentu

### 3 Role členov projektového tímu

Manažerska činnosť	Pridelená osoba
Vedúci tímu a manažér komunikácie	Bc. Adam Kulíšek
Manažér dokumentácie	Bc. Rami Mtier
Manažér podpory vývoja a integrácie	Bc. Matej Jenis
Manažér kvality	Bc. Matej Jenis
Manažér rizík	Bc. Boris Buček
Manažér monitorovania	Bc. Tomáš Hnojčík
Manažér plánovania	Bc. Patrik Kolek

Tab. 2: Manažérske úlohy členov v tíme

## 4 Aplikácie manažmentov

V tejto kapitole dokumentu rozoberieme kompetencie, zodpovednosti a povinnosti jednotlivých manažérsky rolí.

### 4.1 Manažment kvality

Manažér kvality kontroluje správnosť procesov. Overuje, či vybrané procesy vykonávajú vybranú činnosť (smerujú k stanovenému cieľu) a či ju vykonávajú správne. Manažér kvality je teda zodpovedný za verifikáciu a validáciu procesov. Taktiež je zodpovedný za tvorbu konvencií pri písaní zdrojového kódu (konvencie programovania). Tieto konvencie obnášajú opis techník napríklad pri vytváraní tried, pomenovaní premenných a metód, spôsoboch zapuzdovania premenných a metód a podobne, písaní komentárov. Tieto techniky sú stanovené za účelom udržovania kódu prehľadného a čitateľného. V našom projekte je dôležitým cieľom manažéra kvality dohliadať, aby bol kód ľahko pochopiteľný pre našich nasledovníkov, ktorý s ním budú pracovať. Za účelom zjednodušenia zrozumiteľnosti kódu spravuje manažér kvality v našom projekte aj proces refaktoringu existujúceho kódu. V tomto procese sa jedná hlavne o úpravu nevhodných názvov premenných a metód, odstránenie nepotrebných tried, prípadne rozdelenie zložitých tried do viacerých jednoduchších modulov.

#### 4.1.1 Verifikácia a validácia procesov

Základnou úlohou manažera kvality je rozdeliť úlohy daného šprintu jednotlivým členom tímu. V našom projekte sme sa snažili pridelovať úlohy skupinám nie jednotlivcom. Úlohou manažera kvality bolo následne overiť prácu na taskoch. Manažér kvality vytvorený kód preveril a zastili nakoľko dodržiava stanovenú štruktúru a konvencie popísané v nasledujúcich odstavcoch. Samozrejme overil aj výstup implementovanej časti, čím overil nakoľko sú požiadavky na úlohu splnené. Manažér kvality teda vykonáva určitý druh verifikácie a validácie. Na túto úlohu mohol manažér kvality poveriť akéhokoľvek člena tímu. Prevažne to však boli členovia tímu, ktorý rozumeli existujúcemu prototypu v tom čase najlepšie a vedeli posúdiť, nakoľko je nová implementácia vhodná a zodpovedá štruktúre existujúceho kódu.

### 4.2 Manažment plánovania

Plánovanie predstavuje jednu z etáp životného cyklu projektu. Pri plánovaní určujeme aké ciele chceme dosiahnuť na konci projektu a v jeho jednotlivých fázach. Určujeme aké nástroje na to použiť, koľko času a prostriedkov na to budeme potrebovať. Najintenzívnejšie býva zväčša v úvodných častiach projektu kde sa

rozhodujeme, ako bude vyzerat' plán projektu. V ďalších fázach projektu, keď sa projekt vykonáva a riadi sa plány zväčša prispôsobujú aktuálnym požiadavkám. Medzi hlavné dôvody prečo sme manažment plánovania zaviedli do nášho projektu je zredukovanie neurčitostí v projekte (v našom projekte sa ich ukázalo pomerne dosť), zväčšenie výkonnosti a efektívnosti tímu, aby každý člen vedel čo má robiť, aby sa dali úlohy paralelizovať a nevznikala duplicitná práca, aby sme vedeli rozumne odhadnúť čas na dosiahnutie stanoveného cieľa. Pomocou plánovania sme si vybudovali dobrý základ pre monitorovanie a riadenie projektu, čo sme následne využili v oboch manažmentoch.

Hlavnou úlohou manažéra plánovania je určiť metodiky ktoré sa budú dodržiavať pri plánovaní projektu. Jeho úlohou je taktiež evidovať požiadavky zákazníka a snažiť sa ich čo najlepšie preniesť do plánov projektu. Pri plánovaní sa snaží rozložiť prácu tak, aby boli všetci členovia tímu mali približne rovnako veľa práce.

#### 4.2.1 Priebeh plánovania v našom projekte

V úvodnej fáze sme plánovali ako budeme komunikovať medzi sebou v roli členov tímu a ako budeme komunikovať ako tím s vedúcim projektu. Nasledovalo plánovanie času a miesta stretnutia.

Ďalším krokom bola voľba metodík na plánovanie projektu. Zvolili sme si nástroj Trello<sup>1</sup> do ktorého sa dajú pridávať a upravovať jednotlivé úlohy prislúchajúce nášmu projektu. Výhodou toho nástroja je, že jednotlivé úlohy sa dajú jednoducho rozdeľovať do ľubovoľných skupín, evaluovať, určovať termíny pre dokončenie jednotlivých úloh a pridelovať jednotlivým členom tímu.

Ďalšia dôležitá časť, kde prebiehala výrazná časť plánovania boli naše stretnutia v škole, ktoré sa odohrávali každý utorok od 11:00 do 14:00. Na konci každého stretnutia sme si naplánovali jednotlivé úlohy do ďalšieho týždňa a väčšinu úloh sme popridelovali jednotlivým členom tímu.

V priebehu projektu sa naše plány viac krát zmenili v dôsledku nadobudnutia nových informácií pri analýze projektu. Keďže projekt, ktorý bol pridelený nášmu tímu má pomerne dlhú históriu a pracovalo na ňom veľa ľudí, či už v rámci tímových projektov alebo diplomových a bakalárskych prác, bolo neľahkou úlohou zanalyzovať aktuálny stav projektu, vybrať najdôležitejšie časti a na základe toho naplánovať ďalšie smerovanie projektu.

V posledných šprintoch sme začali plánovať implementáciu jednotlivých častí projektu. Pri plánovaní sme sa stretli s problémom paralelizovania úloh, pretože

---

<sup>1</sup> <https://trello.com/>



jednotlivé implementačné časti na seba nadväzovali. Tento problém sme čiastočne vyriešili snahou o spoločné implementovanie novej funkcionality (zišli sme sa v škole a na jednom počítači sme sa snažili riešiť problémy ktoré vznikali pri pridávaní novej funkcionality do projektu).

### 4.3 Manažment monitorovania

Manažér monitorovania má na starosti pridávanie a aktualizáciu úloh v nástroji pre manažment úloh na začiatku, prípadne počas šprintu. Rovnako sa musí starať o to, aby každý člen tímu mal v tomto nástroji pridelené úlohy, ktoré sa majú vykonať pre konkrétny šprint. Každá úloha je ohodnotená zložitou jej vypracovania a je stanovený dátum a čas, dokedy sa musí daná úloha dokončiť. Čím viac je problém zložitejší, tým viac členov tímu sa môže podieľať na jeho riešení.

Hlavnou úlohou je teda organizácia úloh v nástroji Trello, ktorý sme si vybrali pre manažment úloh. Ďalšou dôležitou funkciou je sledovanie včasného ukončenia úlohy a po jej ukončení priradenie manažéra kvality k ukončenej úlohe, ktorý spraví kontrolu jej kvality.

### 4.4 Manažér podpory vývoja a integrácie

Úlohou manažéra podpory vývoja a integrácie je v našom projekte hlavne navrhnúť metodiku pre udržiavanie prostredia pre vývoj a testovanie produktu. Dbal na dodržiavanie pravidiel verziovania a práce s Gitom<sup>2</sup>. Pri práci s týmto nástrojom sme spoločne s tímom navrhli vlastný model s ktorým pracujeme. Takisto manažér podpory vývoja a integrácie spravuje webového klienta Gitu, ktorým je v našom prípade Bitbucket<sup>3</sup>.

#### 4.4.1 Vetvenie zdrojového kódu

Vytvorený model pozostáva z hlavnej vetvy (master), ktorá obsahuje funkčný, otestovaný kód pripravený na použitie. Tieto tri základné požiadavky musia byť splnené, aby mohla byť vetva pripojená do master vetvy. Z hlavnej vetvy je odvodená deployment vetva, ktorá zahŕňa kód vo vývoji, s ktorým sa už ale dá pomerne dobre pracovať. Ostatné vetvy nesú názov feature, ktorá sa v nich implementuje a v týchto vetvách sa vykonáva väčšina projektu. Po otestovaní kód postupuje do vyšších vrstiev.

### 4.5 Manažment komunikácie

Manažér komunikácie je osoba zodpovedná za určenie a vytvorenie komunikačných kanálov pre členov tímu. Tie je potrebné vhodne nakonfigurovať a vytvoriť k nim

---

<sup>2</sup> <https://git-scm.com/>

<sup>3</sup> <https://bitbucket.org/>

prislúchajúce metodiky využívania. Metodiky určujú povolené a nepovolené postupy práce v kanáloch, špecifikujú využitia rôznych technických funkcionalít, ktoré stanovené kanály poskytujú a štandardizujú spôsoby komunikácie.

## 4.6 Manažment tvorby dokumentácie

Tvorba dokumentácie je dôležitou súčasťou výstupu vytváraného projektu. Dokumentácia zhrňuje dosiahnuté výsledky a približuje projekt budúcim vývojárom projektu, manažerom. Agilný vývoj metódou SCRUM vyžaduje väčšie množstvo dokumentácie. Manažér dokumentácie zodpovedá za vytváranie šablón pre výstupné dokumenty, kontrolu kvality a včasné vytváranie dokumentov.

Vytvorené boli tri základné

### 4.6.1 Vytvárané dokumenty

Projekt vyžaduje používanie istých typov dokumentov:

- **zápisnice zo stretnutí** - zápisnice zo stretnutí zhrňujú informácie o priebehu stretnutia s projektovým vlastníkom.
- **dokumenty o retrospektíve** - po šprinte sa vykonáva záznam zhrnutia retrospektívny
- **inštalčná príručka** - príručka s informáciami o rozbehnutí spájaných projektov s riešeniami vyskytnutých problémov.
- **dokumentácia riadenia projektu** - obsahuje informácie o rozdelení manažérskych úloh, metodikách použitých v projekte a postupoch a zmenách ktoré počas prvých mesiacoch vykonali
- **dokumentácia k produktu(tri šprinty, big picture)** - je technická dokumentácia o produkte, ktorý vyvíjame a vykonaných zmenách, ktoré v projekte nastali.
- **Dokument úloh priradený jednotlivým rolám** - dokument, ktorý obsahuje jednotlivé manažérske úlohy a opis činností ktoré vykonávajú konkrétne v našom tíme.

Informácie o štruktúrach dokumentov a zmenách ktoré sa vykonali počas práce na projekte sa nachádzajú v kapitole Metodiky, podkapitole Metodika tvorby dokumentácií.

## 4.7 Manažment rizík

Manažéri rizík sú v tímoch zodpovední za identifikovanie potenciálnych rizík a navrhovanie efektívnych riešení, čím sa môže predísť možným škodám. Predchádzanie takýmto hrozbám má za následok jednoduchšie dosiahnutie stanoveného cieľa bez zbytočnej straty času alebo zdrojov.

Hlavnou úlohou manažéra rizík v našom tíme je vytvorenie metodiky pre identifikáciu, analýzu a riešenie rizík. Manažér rizík je tiež zodpovedný za určovanie závažnosti identifikovaných rizík. Je tiež potrebné, aby o navrhnutých riešeniach na jednotlivé riziká vedel každý člen tímu a aj sa nimi riadil. Neskorá identifikácia rizík môže mať za následok väčšie spôsobené škody. Riziká môžu navyše nastať v akejkol'vek fáze projektu a každé môže mať iný dopad. Zníženie efektivity členov tímu alebo tímu celkovo, nesplnenie úloh v stanovenom čase alebo aj zlyhanie celého projektu - každý z týchto negatívnych dopadov môže byť spôsobený neidentifikovaným alebo ignorovaným rizikom. Na niektoré riziká existujú jednoduché riešenia, iné je možné len zmierniť, ale nikdy úplne odstrániť. Na všetky tieto veci musí manažér rizík pri vytváraní metodiky prihliadať. Pri návrhu riešení musí manažér rizík riešenia analyzovať, aby ich aplikovaním nevznikli ďalšie problémy.

## 5 Sumarizácia šprintov

V tejto kapitole je zhrnutá naša práca v jednotlivých šprintoch. Pri identifikácii nových šprintov sme sa riadili podľa harmonogramu predmetu. V tomto harmonograme bolo stanovené, kedy sa má pracovať na ktorom šprinte a toho sme sa snažili držať. Každý šprint trval 2 týždne.

### 5.1 Úvodná fáza

V úvodnej časti projektu (pred prvým šprintom) sa členovia tímu navzájom predstavili a ako tím sme sa zoznámili aj s vlastníkom produktu. V tejto časti sme sa hlavne oboznámili s témou projektu, dohodli sa na spôsobe komunikácie v rámci tímu (HipChat<sup>4</sup>), vybrali sme si nástroj pre manažment úloh (Trello) a ako repozitár pre náš projekt sme si zvolili BitBucket. Pre zlepšenie fungovania tímu a jasné definovanie toho, kto má čo na starosti, bola každému členovi tímu priradená rola, za ktorú je zodpovedný. Rovnako sa nám podarilo vytvoriť stránku tímu a aj kostra stránky pre 3D UML Lab. Pre rýchlejšie oboznámenie sa s témou 3D UML sme absolvovali stretnutie s ľuďmi, ktorí majú skúsenosti s prácou na tomto projekte. Na tomto stretnutí nám boli prezentované doterajšie riešenia, ich stav a aké sú možnosti v pokračovaní riešenia tohto projektu.

Počas tejto fázy sme si osvojili používanie komunikačného kanálu HipChat, prácu s nástrojom Trello a zamýšľali sa nad možnosťami pokračovania riešenia projektu.

### 5.2 Prvý šprint

Po úvodnej analýze projektu sme sa dohodli aj s vlastníkom produktu na integrácii existujúcich riešení do jednotného funkčného riešenia. V rámci tohto šprintu sme sa snažili získať všetky existujúce riešenia (bakalárske práce, výstup práce z minuloročného tímu). Každý člen si nainštaloval vývojárske prostredie pre 3D UML a snažil sa rozbehať existujúce projekty na svojom počítači. Nakoľko tento postup nebol triviálny, vytvorili sme tutoriál k rozbehaniu projektu pre uľahčenie tohto procesu v budúcnosti. Po inštalácii a spustení projektov sme sa s nimi zoznamovali viac z technického hľadiska a snažili sme sa ich pochopiť.

V rámci tohto šprintu sme rozšírili aj funkcionality pre manažovanie našich úloh. Stiahli sme si rozšírenie ScrumForTrello<sup>5</sup>, vďaka čomu sme boli schopní ohodnocovať jednotlivé úlohy v šprinte.

---

<sup>4</sup> <https://www.hipchat.com/>

<sup>5</sup> <http://scrumfortrello.com/>

## 5.3 Druhý šprint

V druhom šprinte sme sa obmedzili len na štúdium projektu aktivity diagramu, ktorý bol spravený podľa novo postavenej architektúry. Nakoľko dokumentácia k tejto architektúre bola slabšia, museli sme analyzovať každú jednu triedu v diagrame tried a hlbšie skúmať jej metódy.

Na začiatku tohto šprintu sme si definitívne stanovili kompetencie rolí každého člena tímu, nakoľko v predchádzajúcom šprinte to bolo ešte mierne chaotické. Po jasnom stanovení úloh si každý člen tímu začal plniť svoje úlohy, za ktoré bol zodpovedný v rámci organizácie tímu.

## 5.4 Tretí šprint

Vzhľadom k tomu, že prvé 2 šprinty boli viac teoretické a viac zamerané na analýzu už existujúcich riešení, v treťom šprinte sme už začali pracovať na integrácii existujúceho riešenia sekvenčného diagramu do projektu aktivity diagramu. Projekt pre aktivity diagram je vytvorený už v novej architektúre, ktorú chceme využiť aj pri implementácii sekvenčného diagramu. Nevýhodou tohto projektu bolo, že neobsahuje dostatočnú technickú dokumentáciu a tak sme si pre celý projekt vytvorili diagramy tried pre lepšie pochopenie závislostí medzi jednotlivými komponentami. Tieto poznatky sme potom vedeli aplikovať pri implementácii sekvenčného diagramu.

## 6 Používané metodiky

V tejto kapitole dokumentu sa nachádzajú nami navrhnuté metodiky s ich stručným opisom. Kapitola je rozdelená do častí podľa jednotlivých druhov manažmentu.

### 6.1 Manažment kvality

Pri manažmente kvality bolo úlohou manažéra kvality navrhnuť niekoľko metodík za účelom dodržania stanovenej prehľadnosti a kvality našej práce. Metodiky, ktorými sme sa riadili sú zapísané nižšie.

#### 6.1.1 Metodika písania zdrojového kódu

Pri tvorbe metodiky písania zdrojového kódu sme sa snažili dodržiavať a vylepšiť metodiku stanovenú predchádzajúcim tímom, na ktorého prácu nadväzujeme. Ide predovšetkým o tvorbu a pomenovanie tried a premenných, zapuzdrovanie a komentovanie.

#### 6.1.2 Metodika tvorby tried a premenných

Hlavná konvencia tvorby tried vychádza zo samoného jazyka C++<sup>6</sup>. Je to konvencia delenia tried na hlavičkové (header) a zdrojové (.cpp) súbory. V hlavičkových súboroch je potom uvedená deklarácia metód a premenných. V zdrojových súboroch sa nachádza definícia týchto metód. Táto konvencia sa ukázala praktickou, nakoľko zvyšuje čitateľnosť rozsiahleho kódu. Hlavičkové súbory poskytujú programátorom prehľad o činnosti danej triedy, bez rozsiahleho kódu samotnej implementácie metód. Pomenovanie tried sme v našom projekte zvolili spôsobom CamelCase<sup>7</sup>. Ide o štýl písania v programovaní, kde je celý názov napísaný pohromade s malými písmenami okrem prvých písmen každého slova v názve. Tie sú písané veľkými písmenami. Názvy našich tried pozostávajú z viacerých slov, preto nám CamelCase prišiel ako správna voľba. CamelCase sme taktiež uplatnili pri pomenovaní metód a premenných, keďže to je zároveň štandardná konvencia mnohých jazykov vrátane C++.

#### 6.1.3 Zapuzdrovanie a modifikátory prístupu

V našom projekte sa predovšetkým snažíme o čo najväčšiu uzavretosť kódu. Verejnými metódami sa stávajú len základné a najdôležitejšie metódy danej triedy. Pomocné metódy cudzím triedam neposkytujeme. Pri premenných rozlišujeme nakoľko sa jedná o hlavnú triedu alebo triedu zdedenú, od ktorej sa už nebude dediť.

---

<sup>6</sup> <http://www.cplusplus.com/>

<sup>7</sup> <https://msdn.microsoft.com/en-us/library/x2dbyw72%28v=vs.71%29.aspx>

V prípade hlavnej triedy poskytujeme protected premenné, ktoré môžu byť využité podtriedami. Pri zdedených triedach používame private premenné. Prístup k premenným zvonku riešime definovaním tzv. “geterov” a “seterov”. Spomínané konvencie sú rovnako dôležitou časťou refaktoringu popísaného v samostatnej sekcii nižšie.

#### 6.1.4 Metodika komentovania kódu

Pri komentovaní kódu vychádzame zo štandardov knižnice Ogre<sup>8</sup>. Štúdium jej funkcionality nám veľmi uľahčil opis jednotlivých metód v samotných hlavičkových súboroch knižnice. Preto sme sa rozhodli pre netriviálne metódy uviesť stručné komentáre o jej činnosti. V princípe sa jedná o jednoriadkové komentáre typu “*Funkcia vyvolaná pri stlačení tlačidla v Menu.*” Rovnako pre uľahčenie komunikácie a hľadania nedostatkov musí byť v hlavičke každej triedy uvedené kto a kedy danú triedu vytvoril.

#### 6.1.5 Metodika refaktorovania kódu

Refaktorovanie je v našom projekte pokladané za user story, keďže ho vyžaduje náš zákazník. Je teda dôležitou súčasťou projektu, nie len jeho akýmsi obohatením a vykonáva sa počas celého trvania predmetu Tímový projekt. Refaktorovanie sme rozdelili na dve logické časti. Sú to lexikálne refaktorovanie a štrukturálne refaktorovanie kódu. V štrukturálnom refaktorovaní sa zamieňajú nevhodné návrhové vzory a usporiadanie tried. Ide teda o významný zásah do existujúceho prototypu. Preto sme sa rozhodli tento druh refaktorovania uskutočniť v druhom semestri štúdia, nakoľko momentálne by sme mohli doplatiť na nedokonalé znalosti prototypu.

#### 6.1.6 Metodika lexikálneho refaktorovania

V lexikálnom refaktorovaní sa snažíme uplatniť nami zvolené konvencie písania kódu na existujúci kód našich predchodcov. Pri samotnej tvorbe týchto konvencií sme sa v prvom rade pozerali na nedostatky existujúceho kódu a jeho možné vylepšenia. Z toho nám vyplynuli hlavne povinnosti premenovania premenných a okomentovania kódu. Názvy premenných boli často vytvorené bez rozmyslenia s nevhodným názvom a bez použitia CamelCasu. Komentovanie kódu chýbalo pri väčšine tried a bude ho treba doplniť. Taktiež bolo v kóde našich predchodcov množstvo nepoužitého “*odkomentovaného*” kódu, ktorý bolo treba odstrániť.

---

<sup>8</sup> <http://www.ogre3d.org/>

## 6.2 Manažment plánovania

Za účelom zvýšenia efektivity a lepšieho sústredenia sa na cieľ, boli v projekte použité nasledovné metodiky pre plánovanie úloh.

### 6.2.1 Plánovanie úloh v nástroji Trello

Táto metodika slúži na pridávanie nových úloh a editovanie už existujúcich. Nástroj Trello sme zvolili kvôli jeho jednoduchosti a predošlým skúsenostiam niektorých členov tímu.

Kvôli prehľadnosti v plánovaní máme úlohy evidované v dvoch typoch skupín(tabúl):

- Šprinty
- Product Backlog

Pre každý šprint máme vytvorenú samostatnú tabuľu v ktorej sú evidované zoznamy úloh. Do každého zoznamu sú pridelené úlohy na základe ich aktuálneho stavu. V každom šprinte udržiavame tieto zoznamy:

- nápady("IDEAS")
- úlohy ktoré sa majú v aktuálnom šprinte vykonať("TODO")
- úlohy na ktorých dokončení sa pracuje ("IN PROGRESS")
- splnené úlohy ("COMPLETED")
- úlohy ktorých ukončenie je akceptované podľa *Definition of Done* ("DONE")

Úlohy sa presúvajú na základe metodík stanovených v časti monitorovanie projektu. Pri plánovaní sa kladie dôraz na úlohy na ktorých sa ešte nezačalo pracovať alebo tie, ktoré sú už ukončené.

Na tabuli product backlogu sa udržiavajú používateľské príbehy vo forme big picture. Rozloženie zoznamov je podobné ako v šprint backlogu, navyše je tam len zoznam pre používateľské príbehy. Na začiatku každého šprintu sa vyberajú úlohy ktoré by sa mali v danom šprinte vykonávať. Vyberanie úloh je podmienené konzultáciou s vedúcim projektu.

K tejto metodike patrí ešte pridávanie požadovaného času ukončenia každej úlohy v šprinte, na základne jej náročnosti a potreby ukončenia úlohy v istom termíne.

Pridelovanie úloh je podmienené používateľskými rolami v tíme a dohodou medzi jednotlivými členmi tímu, to znamená, že ak má niekto záujem pracovať na nejakej úlohe a nespadá pod jeho rolu, spýta sa ostatných či môže danú úlohu riešiť a po schválení všetkými členmi tímu, je mu úloha pridelená v nástroji trello. Nepridelené



úlohy prideluje manažér plánovania, ktorý sa snaží o spravodlivé rozdelenie úloh

### 6.2.1 Plánovanie úloh na stretnutiach

Na každom stretnutí spravíme retrospektívu ktorej cieľom je odhaliť problémy na ktoré sme narazili v priebehu týždňa alebo tie, ktoré sa podarilo vyriešiť.

Nad nedoriešenými úlohami za ukončený šprint a nad úlohami ktoré máme v produktovom backlogu spravíme planning poker. Metóda plannig poker spočíva v tom, že hodnotíme náročnosť jednotlivých úloh pomocou kartičiek s rôznymi číselnými hodnotami. Každý člen tímu má svoje kartičky a pre každú úlohu vyberie kartičku s hodnotou, ktorú považuje za adekvátnu k náročnosti danej úlohy. Každý člen tímu položí kartičku na stôl tak, aby nikto nevidel hodnotu ktorá sa nachádza na kartičke. Následne všetci členovia tímu ukážu všetci naraz, akú hodnotu zvolili. Členovia ktorí zvolili najmenšie a najväčšie čísla vysvetlia svoju voľbu a o úlohe sa diskutuje. Tento postup sa opakuje až pokiaľ sa všetci členovia tímu nezhodnú na rovnakej hodnote. Ak sa členovia zhodnú, že náročnosť úlohy je väčšia ako 13, tak sa úloha rozloží na menšie podúlohy.

Po pridelení náročnosti jednotlivým úlohám sa členovia dohodnú na tých úlohách, ktoré sa budú vykonávať v najbližšom šprinte. Veľký vplyv na výbere úloh má vedúci projektu, pretože určuje, ktoré úlohy sú aktuálne najprioritnejšie. Vybrané úlohy sa zapíšu do Trelly do tabule pre aktuálny šprint.

### 6.2.2 Definition of Done

Metodika “definícia ukončenia úlohy” slúži na to, aby nám pomohla určiť, kedy je daná úloha hotová a už sa jej netreba viac venovať.

Úloha je ukončená ak spĺňa nasledovné podmienky:

- k úlohe je stručne a výstižne okomentovaný zdrojový kód.
- program je ošetrený voči chybám.
- kód je nasadený vo vývojovej vetve vo verziovacom systéme.
- funkcionálna je otestovaná kontrolórom úlohy z pohľadu požiadaviek.

### 6.2.3 Metodiky v praxi

Dodržiavanie metodík týkajúcich sa plánovania bolo pre členov tímu osožné. Mali sme lepší pohľad na aktuálny stav projektu a pomerne ľahko sa dohľadával problém, ak sa niečo nestihlo spraviť na čas alebo sa na niečo zabudlo. Ďalšia prínosná vec dodržiavania metodiky bola tá, že v každom týždni mal každý člen nejakú úlohu ktorú mal vypracovať. Týmto spôsobom sme sa snažili zabezpečiť rovnomerné rozdelenie práce medzi členov tímu. Bohužiaľ nie vždy bolo možné spravodlivo

rozdeliť množstvo práce najmä v úvodných fázach projektu, pretože v tíme bolo nerovnomerná znalosť problematiky medzi jednotlivými členmi v dôsledku predošlých skúseností na projekte. Kvôli snahe o rýchli štart sa úlohy nerozdeľovali úplne rovnomerne, ale priradovali sa tým členom tímu u ktorých sa predpokladalo, že zvládnu danú úlohu výrazne rýchlejšie ako členovia ktorý sa v danej problematike predtým nepohybovali (prehĺbenie vedomostí v problematike nebola triviálna záležitosť a vyžadovala pomerne veľa času). Tento spôsob bol uplatnený iba v raných fázach projektu. Po dosiahnutí lepšieho pohľadu na projekt sa úlohy začali rozdeľovať rovnomernejšie čo veľmi zvýšilo progres a efektivitu vykonávanej práce.

## 6.3 Manažment monitorovania

### 6.3.1 Organizácia stavu úloh v nástroji pre manažment úloh

Pre monitorovanie a manažment úloh sme si v našom tíme zvolili nástroj Trello. V začiatkovej fáze riešenia projektu bola vytvorená jedna tabuľa (board), do ktorej sa pridávali všetky úlohy spolu s dátumom ich ukončenia. Úlohy sú kategorizované na základe ich stavu riešenia:

- “IDEAS” - všetky úlohy, ktoré sú navrhnuté členmi tímu. Následne po ich analýze a schválení sú úlohy pridané do kategórie „TODO“
- “TODO” - zoznam úloh, ktoré boli určené pre daný šprint
- “IN PROGRESS” - úlohy, na ktorých členovia tímu práve pracujú
- “COMPLETED” - úlohy, ktoré boli riešiteľom označené ako splnené, ale nebolo ešte overené, či spĺňajú *Definition of Done*.
- “DONE” - zoznam úloh, ktoré spĺňajú *Definition of Done*.

Manažér monitorovania vytvára tieto úlohy na začiatku a počas šprintu. Každá úloha obsahuje aj farebné označenie, na základe ktorého vieme identifikovať fázu projektu, ku ktorej bola úloha priradená. Po pridelení úloh jednotlivým členom tímu si následne každý člen sám aktualizuje stav riešenia úlohy. Na každom stretnutí manažér monitorovania prejde zoznam úloh, ktoré sa mali vykonať a ktoré sa reálne vykonali. Následne prebieha vyhodnotenie vykonania všetkých úloh a ich aktualizácia v nástroji Trello. Tento postup sa využíval v úvodnej fáze projektu a počas prvého šprintu.

V druhom šprinte sa pre monitorovanie úloh vykonalo viacero zmien. Vzhľadom k tomu, že v nástroji pre manažment úloh existovala len 1 tabuľa a zoznam úloh rapídne narastal, začínala byť táto tabuľa neprehľadná. Prehľadnosť sa znižovala hlavne v tom, že hoci každá úloha mala priradené farebné označenie, ku ktorému šprintu patrí, začalo byť toto označenie nepostačujúce. Preto sa pre každý šprint vytvorila nová tabuľa, v ktorej sa identifikovali úlohy pre daný šprint. Rovnako sa vytvorila aj nová tabuľa pre produktový backlog, ktorá nám poskytuje jednoduchý

prehľad o tom, na ktorých user-stories momentálne pracujeme a ktoré sa nám podarilo dokončiť podľa stanoveného plánu.

Pre zvýšenie kvality monitorovania ukončenia úloh bola v tabuli vytvorená aj nová kategória "COMPLETED". Každý člen, ktorý ukončil pridelenú úlohu ju pridá do zoznamu ukončených úloh ("COMPLETED"). Následne manažér monitorovania priradí manažéra kvality k tejto úlohe, aby ju skontroloval. Ak je všetko v poriadku, úloha sa pridá do zoznamu "DONE". Tento stav reprezentuje, že úloha bola nielen ukončená, ale aj skontrolovaná z hľadiska kvality a spĺňa stanovenú definíciu splnenia úlohy (*Definition of Done*). Týmto procesom sme zvýšili kvalitu vytvorených úloh. Výhodou je aj jednoduchý prehľad o úlohách, ktoré je potrebné ešte skontrolovať a ktoré je už možné pridať do výsledného medziproduktu.

### 6.3.2 Identifikácia zložitosti úloh

Problémom v úvodnej fáze bolo zaznamenávanie zložitosti úloh v tomto nástroji. Aby sme boli schopní zaznamenať pre každú úlohu jej zložitosť, stiahli sme si rozšírenie „ScrumForTrello“, pomocou ktorého sme vedeli každej úlohe priradiť tzv. storypoints. V prvom šprinte sme zaznamenávali zložitosť úlohy v človeko-hodinách (každý jeden storypoint v Trelle predstavoval 1 človeko-hodinu). Pri identifikácii novej úlohy sa odhadoval približný čas, ktorý je potrebný pre jej vykonanie. Následne každý člen tímu po ukončení úlohy zaznamenal, koľko času reálne strávil pri tejto úlohe. Vďaka tomu bolo možné vyhodnotiť, či naše plánovanie a odhady boli správne.

V druhom šprinte sa odhadovanie náročnosti úloh v človeko-hodinách nahradilo za čísla z Fibonacciho postupnosti, tzv. Planning poker. Tento odhad náročnosti je zadaný ku každej úlohe v nástroji Trello pomocou storypoints, vďaka ktorým máme prehľad o zložitosti každej úlohy v hlavnej tabuli.

### 6.3.3 Postup pri vytváraní/pridelovaní úloh na začiatku šprintu

Na začiatku šprintu sa skontroluje tabuľa pre produktový backlog v nástroji Trello a zhodnotia sa používateľské príbehy (user-stories), ktoré sa podarilo dokončiť. Pri identifikácii úloh pre nový šprint prebieha komunikácia s vlastníkom produktu. Vlastník produktu určí, ktoré používateľské príbehy by sa mali dodať v novom šprinte. Tieto úlohy sa v tabuli pre produktový backlog presunú do stavu „IN PROGRESS“ (v nástroji Trello). To znamená, že v novom šprinte budeme pracovať na používateľských príbehoch identifikovaných v tomto zozname. V ďalšej fáze prebieha rozdeľovanie používateľského príbehu na menšie časti (úlohy), ktoré je potrebné vykonať na splnenie cieľa. Úlohy sú ohodnotené podľa jej náročnosti pomocou metodiky Planning poker. Po úspešnom identifikovaní nových úloh a ich ohodnotení musia byť tieto úlohy zaznamenané v nástroji Trello pre nový šprint. Výsledkom je zoznam úloh, ktoré sú ohodnotené svojou náročnosťou a každý člen má priradené úlohy, ktoré musí vykonať.

#### 6.3.4 Postup pri vytváraní/prideľovaní úloh počas šprintu

Hoci na začiatku šprintu máme definované úlohy, ktoré sa musia vykonať pre celý šprint, môže sa stať a často sa stáva, že potrebujeme niečo zmeniť, resp. pridať. Práve na tento účel v nástroji Trello sme si definovali kategóriu „IDEAS“. V prípade, že má niektorý z členov nejaký nový nápad prípadne požiadavku na zmenu, zaznamená to do tejto kategórie. Následne sa na najbližšom stretnutí tieto zmeny a nápady prediskutujú v rámci členov tímu a aj s vlastníkom produktu.

Ak boli zmeny schválené, nové úlohy sú pridané k existujúcemu zoznamu úloh pre daný šprint. Výsledkom je aktualizovaný zoznam úloh, ktoré je potrebné ukončiť v rámci stanoveného termínu.

#### 6.3.5 Zhodnotenie použitej metodiky

V úvodnej fáze riešenia projektu nám chýbalo hlavne odhadovanie náročnosti úloh, ktoré sme neskôr čiastočne nahradili odhadom v človeko-hodinách. Takéto ohodnotenie však nepredstavovalo reálnu zložitost' problému. Často krát nastal prípad, v ktorom vykonanie rovnakej úlohy trvalo niektorým členom tímu dlhšie. Na základe týchto rôznych časov sme neboli schopní určiť, či náš odhad bol správny alebo nie. Odhad náročnosti bol však neskôr nahradený číslami z Fibbonaciho postupnosti (1,2,3,5,...). Vďaka tejto metodike sme boli schopní reálne určiť zložitost' problému (bez ohľadu na čas, ktorý strávil člen tímu na jeho riešení).

Rovnako môžeme aj pozitívne zhodnotiť zmeny v manažmente úloh v nástroji Trello. Po aplikovaní zmien sa manažment úloh výrazne zjednodušil a je viac prehľadný.

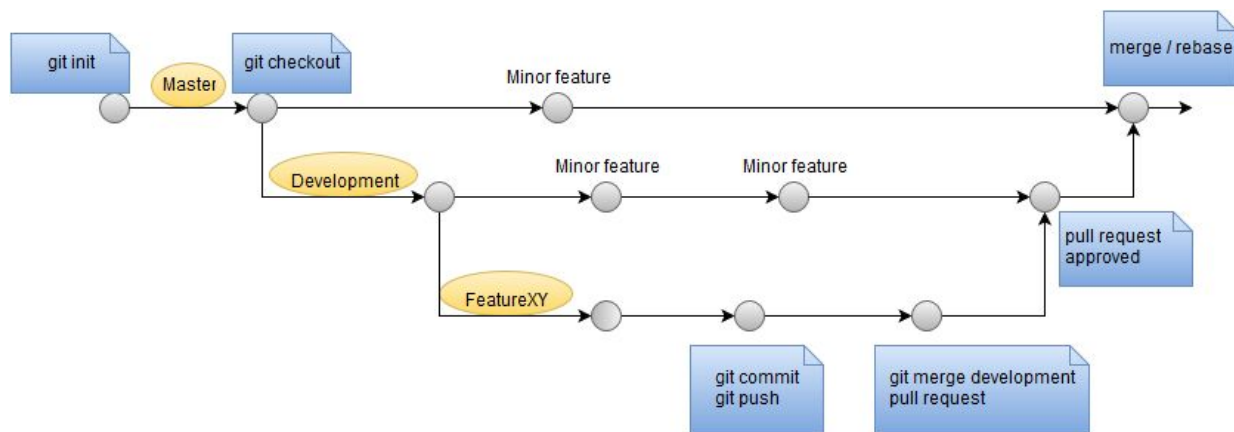
### 6.4 Manažment podpory a vývoja a integrácie

Pri práci s Gitom sme identifikovali metodiku, ktorá nám určovala spôsob pridávania novej funkcionality v Gite. Takisto sme si určili metodiku pre písanie správ odovzdaní (commit message) pri pridávaní novej funkcionality.

#### 6.4.1 Metodika pridávania novej funkcionality

Z hlavnej (master) vetvy je odvodená nová vetva určená pre vývoj s názvom development. Tá reprezentuje všetky zmeny v prototypu, ktoré sa ešte vyvíjajú, prípadne nie sú otestované, alebo si nie sme istí, či sú vykonané úplne správne a podľa našich predstáv. Akékoľvek malé zmeny v zdrojovom kóde sme v našom tíme commitovali priamo do development vetvy. Pod malými zmenami sa rozumie úprava syntaxe, malá zmena metódy jej tvorcom alebo úprava chýb. V prípade rozsiahlejšej zmeny, ktorá môže ohroziť stabilitu development vetvy sa z nej vytvorí nová vetva s názvom funkcionality, ktorá sa v budúcnosti bude vytvárať. Takéto vetvy podporujú prehľadnosť vo verziovaní a vyhýbajú sa konfliktom, keďže členovia tímu môžu pracovať na

rôznych funkcionalitách zároveň. Po dokončení práce na danej vetve sa spraví pull request do development vetvy. Po skontrolovaní kódu manažérom kvality, prípadne inými členmi tímu bude pull request akceptovaný a vykoná sa rebase do development vetvy, pričom do tejto vetvy sa bude pridávať výhradne bezchybný kód, ktorý už nemáme v pláne upravovať a považujeme ho za kompletný. Navrhnutý model je zobrazený na obrázku nižšie a jedná sa o diagram vytvorený na jednom z našich tímových stretnutí.



Obr. 1: Git workflow

### 6.4.2 Zásady pomenovania commitu

Pre každé odovzdanie (commit) je nutné pripojiť správu, ktorá zhrnie, čo daný commit priniesol a zmenil. Našou zásadou je commitovať postupne s menšími zmenami. Rozhodli sme sa preto vytvoriť metodiku pre tvorby názvu commitu:

1. ako prvé určíme v hranatých zátvorkách názov šprintu, v ktorom bol commit vykonaný.
2. nasleduje stručný a výstižný názov zmien v commite.
3. v ďalších riadkoch nasledujú zhrnutia drobnejších funkcionalít, ktoré stoja za zmienku (za predpokladu, že nejaké sú).
4. ako posledný riadok je odkaz na úlohu v softvéri na evidenciu úloh, ktorej sa commit týka.

## 6.5 Manažment komunikácie

Pre potreby tímu stanovil manažér komunikácie komunikačné kanály, určil metodiky ich používania a spôsoby ich prepojenia. Nasledovne ich manažér nakonfiguroval tak, aby boli pripravené k použitiu. Boli navrhnuté dve základné oblasti komunikácie medzi členmi tímu:

- každodenná komunikácia
- zdieľanie dát

Nasleduje zhrnutie metodík pre komunikáciu v tíme, ktoré sme počas šprintov dodržiavali.

### 6.5.1 Komunikačný kanál pre každodennú komunikáciu

Už po úvodnom stretnutí bol pre potreby každodennej komunikácie stanovený skupinový chat HipChat. Jeho výhodou je dostupnosť z mobilných zariadení, z webového prehliadača, či desktopovej aplikácie pre operačné systémy Windows a Mac. Umožňuje tvorbu viacerých skupinových miestností, jednoduchú administráciu a integráciu s rôznymi službami ako napríklad Bitbucket. Všetka komunikácia v tíme sa musí odohrávať v tejto službe, v žiadnom prípade mimo nej (Facebook, email, iný chat).

Vytvorené boli tieto miestnosti:

- Development (vývoj prototypu)
- Git (verziovanie systému)
- Dokumentácia
- 3D UML (iné témy súvisiace s projektom)

Pravidlá používania miestnosti "Development":

1. diskusia sa musí priamo týkať vývoja projektu - nejasnosti ohľadom implementácie, architektúry prototypu, technické otázky a iné.
2. znakom @ označujeme osoby, ktoré s témou súvisia, alebo od ktorých vyžadujeme odpoveď na otázku.
3. je zakázané notifikovať všetkých používateľov pomocou "@all".
4. miestnosť je prepojená s repozitárom projektu na Bitbucket, takže po každom vykonanom commite bude zobrazená informácia o jeho vykonaní spolu so správou a autorom v danej vetve.

Pravidlá používania miestnosti "Git":

1. predmetom diskusie v tejto miestnosti je verziovanie systému, riešenie nejasností ohľadom metodiky verziovania a riešenie konfliktov vzniknutých počas vývoja.
2. nepatria sem pripomienky ku konkrétnym commitom, tie sa riešia priamo v službe Bitbucket pri náhľade na konkrétny commit, či pull request.

Pravidlá používania miestnosti "Dokumentácia":

1. miestnosť je vyhradená pre konverzácie súvisiace s obsahom, štruktúrou a tvorbou akéhokoľvek druhu dokumentov.
2. člen tímu po každom zásahu do dokumentu v tejto miestnosti zhrnie všetky vykonané zmeny.

3. znakom @ označujeme osoby, ktoré s témou súvisia, alebo od ktorých vyžadujeme odpoveď na otázku.

Pravidlá používania miestnosti "3D UML":

4. miestnosť je určená pre komunikáciu k témam ohľadom tímového projektu, ktoré nespádajú do vyššie spomenutých miestností.
5. v prípade nastatia naliehavej situácie je povolené používať skratku pre notifikáciu všetkých členov "@all".
6. pokiaľ je vedená diskusia k dokumentu, stránke, fotke, či inému digitálnemu obsahu na webe, je nutné pripojiť odkaz naň smerujúci.

### 6.5.2 Zdieľanie dát

Pre zhromažďovanie dokumentov, materiálov, fotiek, či iných zdrojov informácií bola zvolená služba Google Drive. V prípade tvorby zápisnice, diagramu, dokumentu, fotky, obrázku a iných výstupov je nutné ich zverejniť po dokončení do patričného priečinka tohto úložiska. Je nutné zachovávať vopred zvolenú štruktúru a dokumenty vhodne organizovať do priečinkov.

### 6.5.3 Zhodnotenie dodržiavania metodík

Od prvého týždňa sme komunikovali pomocou nástroja HipChat. Chvíľu trvalo, kým sa všetci členovia tímu s nástrojom stotožnili, no po istom čase sme ho začali využívať vo veľkej miere. Hlavným problémom bola nedostatočná aktivita a dostupnosť členov tímu v miestnostiach počas prvých týždňov semestra. Pravidelne a vhodne sme používali znak @ na notifikovanie konkrétnych členov v prípade nutnosti pridania ich reakcií. Označenej osobe príde emailom daná správa s odkazom na miestnosť, v ktorej bola napísaná. Zdieľanie dát a dokumentov nebolo spočiatku realizované v dostatočnej miere, no po prvom šprinte sme tento problém začali riešiť. V nasledujúcich šprintoch sme už dôkladne začali zdieľať všetky výstupy do správnych priečinkov.

## 6.6 Manažment dokumentácie

V projekte sme si stanovili metodiku tvorby dokumentácie. Metodika určuje spôsoby vytvárania dokumentov, ich záloh, zdieľania a štruktúry. Rola dohliadajúca na korektné dodržiavanie metodiky je Manažér dokumentácie. Dohliada na včasné vytváranie dokumentov, určuje štruktúru dokumentov a je zodpovedný za ich kvalitu. Opis metodiky sa nachádza v kapitole Metodika.

V tejto časti sa nachádzajú informácie o štruktúrach jednotlivých dokumentov, procese zdieľania dokumentov, princípoch zálohovania dokumentov a vytvárania pomocných dokumentov. Informácie o obsahu dokumentov sú zahrnuté v kapitole Manažment tvorby dokumentácie.

V nasledujúcich sekciách sa nachádzajú informácie o štruktúrach jednotlivých dokumentov. Jednotlivé dokumenty sú kontrolované manažérom dokumentácie, ktorý v prípade nezahody dodáva pripomienky na zmenu

#### 6.6.1 Dokumentácia k riadeniu projektu a dokumentácia k produktu (tri šprinty, big picture)

Dokumenty obsahujú titulnú stranu, obsah a má presne definovanú štruktúru. Tá sa skladá z nasledovných informácií:

- názov vysokej školy
- fakulta vysokej školy
- adresa, poštové smerovacie číslo
- názov dokumentu
- autorov
- vedúceho projektu
- rok vytvárania projektu

#### 6.6.2 Zápisnice zo stretnutí a dokumenty o retrospektíve

Zápisnice sa vytvárajú sa každý týždeň. Majú presne stanovenú štruktúru, ktorá sa len vyplní. Rozsah dĺžky zápisníc je približne 2 strany. Šablóna na dopĺňanie informácií sa nachádza na obrázku číslo 1. Dokumenty o retrospektíve majú tiež stanovenú štruktúru a vytvárajú sa na konci jednotlivých šprintov.

#### 6.6.3 Inštalačná príručka a Dokument úloh priradený jednotlivým rolám

Tieto dokumenty majú voľnú štruktúru. Sú to interné informačné dokumenty, ktoré majú za úlohu hlavne poňať dôležité informácie. Ale dokumenty musia mať istú formu, čo v prípade nášho tímu znamená aspoň titulná strana a obsah.



## Tím 13

### Téma stretnutia

--

Dátum	
Miestnosť	
Vyhotovil	

### Prítomní členovia:

Vedúci projektu	
Členovia tímu	

### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Príklad úlohy	Všetci členovia tímu		OK
2.	Príklad úlohy	Všetci členovia tímu		OK
3.	...			
4.				

### Priebeh stretnutia

### Zoznam úloh

Obr. 2: Šablóna pre tvorbu zápisníc

#### 6.6.4 Zdieľanie dokumentov

Dokumenty sú vypracovávané vo webovom editore Google Docs<sup>9</sup>. Výnimkou sú zápisnice a dokumenty o retrospektíve. Webové prostredie nám umožňuje zdieľať dokumenty a zosynchronizovať prácu, pridávať pripomienky. Zápisnice a dokumenty o retrospektíve sa vypracovávajú lokálne, je zbytočné ich zdieľať pokiaľ majú len jedného autora.

<sup>9</sup> <https://www.google.com/docs/about/>

### 6.6.5 Dostupnosť a zálohovanie dokumentov

Dokumenty sú dostupné na spoločnom úložisku Google Drive<sup>10</sup>. Webové dokumenty sa nachádzajú vo forme odkazov. Záloha jednotlivých dokumentov prebieha pri veľkej zmene dokumentov, tento proces vykonáva manažér dokumentácie.

### 6.6.6 Pomocné dokumenty

Ich tvorba a štruktúra nie je definovaná. Pokiaľ sa člen tímu rozhodne, že je potrebné zdokumentovať istý proces, zdokumentuje to. Pri nasledovnom stretnutí tímu odôvodní svoj úsudok a pokiaľ sa tím zhodne na podobnom názore manažér dokumentácie stanoví štruktúru a bude dohliadať na ich vytváranie a kvalitu spracovania.

## 6.7 Manažment rizík

Ako prevenciu proti možným rizikám počas projektu sme si stanovili metodiku identifikácie a analýzy rizík. Metodika určuje, ako a kedy sa identifikujú možné riziká. Identifikované riziká je potrebné následne zanalyzovať, zoradiť podľa závažnosti a v danom poradí navrhovať riešenia na odstránenie týchto rizík. Za dodržiavanie tejto metodiky celým tímom je zodpovedný manažér rizík.

### 6.7.1 Identifikácia rizík

Aby nedošlo k vzniku veľkých škôd, je potrebné aby boli riziká identifikované tak skoro, ako je to len možné. Riziká sme sa preto rozhodli identifikovať hneď na začiatku každého šprintu pri tvorbe a pridelovaní jednotlivých úloh. Keďže úlohy vytvárame a zadávame do nástroja Trello už počas spoločného stretnutia tímu s našim vedúcim, nad možnými rizikami uvažuje celý tím. Takýmto spôsobom je tiež zaručené, že o každom riziku vedia všetci členovia tímu a nevznikajú tak zbytočné škody. Identifikované riziká môžu byť viazané na jednu konkrétnu úlohu ako aj na celý šprint, alebo celý projekt.

### 6.7.2 Analýza rizík

Analýza rizík prebieha rovno na spoločnom stretnutí po identifikácii. Je potrebné určiť, aký dopad by približne každé riziko malo v prípade, že by sme sa ho rozhodli ignorovať. Riziká môžeme následovne zoradiť podľa závažnosti. Riziká s takmer žiadnym dopadom môžu byť ignorované v prípade, že na ne neexistuje jednoduché riešenie, aby sa tím mohol zamerať na dôležitejšie veci a nestrácal tak efektivitu. Prioritne je potrebné najprv analyzovať a riešiť najzávažnejšie riziká.

---

<sup>10</sup> <https://www.google.com/drive/>

### 6.7.3 Navrhovanie riešení

Táto časť manažmentu rizík prebieha tiež na spoločnom stretnutí celého tímu. Ak má akýkoľvek člen návrh na riešenie daného rizika, povie ho celému tímu a každý člen sa k nemu osobitne vyjadrí. Prijateľné sú aj riešenia, ktoré riziko neodstránia, ale ho len zmiernia. Je však potrebné navrhované riešenia prediskutovať poriadne, aby ich aplikovaním nevznikli ďalšie možné riziká. V prípade ťažko riešiteľných rizík alebo veľkého počtu rizík bude tímová diskusia pokračovať cez komunikačný kanál. Manažér rizík odovzdá následne po stretnutí celý spísaný zoznam s rizikami a ich riešeniami na spoločné úložisko, aby mali k tomuto zoznamu prístup všetci členovia.

### 6.7.4 Riešenie neočakávaných rizík

Nemusí sa vždy podariť, že sa nám na stretnutí podarí identifikovať všetky riziká. Na riziká môže naraziť každý člen tímu počas plnenia úloh v danom šprinte. V takom prípade je člen povinný čo najskôr kontaktovať manažéra rizík. Manažér rizík zvaží závažnosť a rozsah daného rizika a v prípade, že toto riziko je hrozbou pre náš tím, pošle manažér rizík správu všetkým členom tímu. V prípade, že na riziko existuje jednoduché riešenie, je možné ho prediskutovať cez komunikačný kanál tímu. V opačnom prípade môže byť podaný dočasný návrh na zmiernenie rizika a riešenie bude prediskutované na ďalšom spoločnom stretnutí.

### 6.7.5 Zhodnotenie metodiky

Pre identifikáciu, analýzu a riešenie rizík sme si vyhradili určitý čas na spoločnom stretnutí tímu. Možnými rizikami sa zaoberáme hlavne na začiatku šprintu ale aj počas plnenia pridelených úloh. Na spoločnom úložisku sa nachádza zoznam rizík, ktoré sme identifikovali ako aj navrhnuté riešenia. Keďže táto metodika bola zavedená až od začiatku tretieho šprintu, nebola dostatočne otestovaná v praxi a v budúcnosti v nej môžu nastať určité zmeny. Použitím tejto metodiky sa nám zatiaľ podarilo identifikovať nasledovné významné riziká:

- **Nevieme odhadnúť do akej hĺbky bude potrebné meniť prototypy, ktoré budeme spájať** - toto riziko sme sa rozhodli riešiť vytvorením diagramov cez Microsoft Visio<sup>11</sup>, na ktorých je naznačená štruktúra projektu. Členovia tímu majú tak lepšiu predstavu o tom, kde v prototypy sa jednotlivé veci nachádzajú.
- **Dlhé čakanie na existujúce prototypy / dokumenty k projektu** - Toto riziko sa týka aj prvých dvoch šprintov. Keďže 3D UML je projekt, ktorý už dlhšie existuje, potrebujeme existujúce materiály, aby sme mohli v projekte pokračovať. Toto riziko sa nám pravdepodobne nepodarí úplne odstrániť, ale

---

<sup>11</sup> <https://products.office.com/en/visio/flowchart-software>

rozhodli sme sa riziko znížiť zlepšením komunikácie s osobami, ktoré na tomto projekte už v minulosti pracovali.

## 7 Globálna retrospektíva zimného semestra

Zo začiatku semestra nebolo hneď jasné, aký by mal byť výstup našej práce, preto sme s vlastníkom produktu počas prvých týždňov semestra ujasňovali smer, ktorým by sa tento tímový projekt mal uberať. Paralelne sme analyzovali prototypy sekvenčného diagramu, kombinovaných fragmentov a diagramu aktivít. Značný čas trvalo, kým sa všetkým členom tímu podarilo korektne spustiť projekt na vlastnom stroji. Už v tejto fáze sa nám podarilo identifikovať niekoľko problémov s jednotlivými prototypmi projektu 3D UML, hlavne z hľadiska kompatibility, závislostí softvéru a nedostatočne spracovanej inštaláčnej príručky. Počas prvého šprintu sa naplnilo riziko závislosti na tretích osobách, kedy sme čakali na dokumenty alebo informácie, ktoré nám mala tretia osoba poskytnúť. Riziko sa nám nakoniec podarilo eliminovať tak, že sme sa za žiadnych okolností úplne nespoliehali na získanie zdrojov výhradne od tretích osôb, ale hľadali sme ich aj sami (získanie webového sídla predchádzajúceho tímového projektu od Petra Lacka, atď.).

Do konca prvého šprintu sme nepracovali úplne korektne podľa metodiky FIIT SCRUM a robili sme niekoľko chýb, ktoré sa nám po SCRUM paneli v šiestom týždni podarilo napraviť. Tu sú niektoré z chýb, ktorých sme sa dopustili:

- úlohy sme odhadovali v človekohodinách
- neohodnocovali sme pomocou metodiky planning poker
- nemali sme vytvorený produktový backlog, a teda ani používateľské príbehy.

Niektoré chyby boli spôsobené ešte nevyjasnenou budúcnosťou projektu, no hneď po stretnutí s expertami na SCRUM paneli sme všetky chyby identifikovali a rozhodli sa ich napraviť v budúcich šprintoch. Jeden zo základných problémov bola nedostatočná komunikácia a aktivita členov tímu v komunikačnom kanáli HipChat. Retrospektíva bola vykonaná prvýkrát až na konci druhého šprintu. Počas prvých dvoch šprintov sme primárne analyzovali prototypy, spracovali sme detailnejšiu inštaláčnú príručku, vytvorili stránku 3D Lab-u a vykreslili diagramy tried prototypu diagramu aktivít. Implementáciu sme zahájili až v treťom šprinte, kedy sme s vlastníkom produktu dotvorili finálnu verziu produktového backlogu a jednotlivé používateľské príbehy sme zoradili podľa priority. Naše metodiky a spôsob práce v tíme preto nie sú ešte veľmi odladené podľa prístupu agile, no každým týždňom sa snažíme v tejto oblasti zdokonaľiť a identifikovať chybné postupy. Na cvičeniach z MSI sme s cvičiacim analyzovali a konzultovali pracovné postupy v tíme, čo nám do značnej miery pomohlo.

Na každom stretnutí sme vytvárali zápisnice, v ktorých sme zaznamenali všetkých prítomných, úlohy a ich stav, stručný opis priebehu stretnutia a zoznam nových úloh, ktoré sme počas stretnutia vytvorili. Tie nám pomohli vytvoriť si späťne obraz o našich aktivitách v jednotlivých šprintoch.

V nasledovnej tabuľke je zobrazený odpracovaný čas na projekte počas zimného semestra:

Člen tímu	Odpracovaný čas (hodiny)
Bc. Boris Buček	35
Bc. Tomáš Hnojčík	45
Bc. Matej Jenis	75
Bc. Patrik Kolek	40
Bc. Adam Kulíšek	91
Bc. Rami Mtier	75

Tab. 3: Počet odpracovaných hodín členov tímu

## 7.1 Retrospektíva šprintov

V tejto časti sa nachádza retrospektíva, ktorá obsahuje naše úspechy a neúspechy za daný šprint a aj návrhy na zlepšenie od jednotlivých členov tímu. Retrospektívu sme začali riešiť až na konci druhého šprintu. Trvala približne trištvrte hodinu, každý z členov tímu vyjadril svoj názor na dodržiavanie metodík v minulom šprinte. Každému členovi bol poskytnutý priestor na kritiku predošlého šprintu a na predloženie návrhov na zlepšenie postupov. V prípade návrhu na vytvorenie novej metodiky sa vedie diskusia.

### 7.1.1 Retrospektíva druhého šprintu

Dátum	3.11.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Boris Buček

### Úspechy z druhého šprintu

#	Opis úspechu	Zodpovedné osoby
1.	Preštudovanie kompetencií rolí pridelených jednotlivým členom tímu	Všetci členovia tímu

2.	Štúdium activity diagram branche (Výsledok z tímového projektu 2014/2015)	Všetci členovia tímu
3.	Preštudovanie Class diagram branche od Andreja Železnáka	Všetci členovia tímu

### Pripomienky na vylepšenie efektivity od jednotlivých členov tímu

#	Detekované problémy	Navrhujúca osoba
1.	Nejasné stanovenie úloh v Trelle	Rami Mtier
2.	Podobné úlohy v Trelle	Matej Jenis
3.	Neskoré plnenie úloh	Adam Kulíšek
4.	Zlá organizácia úloh v Trelle	Všetci členovia tímu
5.	Žiadne úložisko pre pomocné materiály	Všetci členovia tímu

Pri vytváraní retrospektívy za druhý šprint sme spolu identifikovali vyššie uvedené problémy. Na každý z týchto identifikovaných problémov sme navrhli riešenie, ktoré sme sa rozhodli aplikovať hneď v nasledujúcom šprinte:

- 1. Nejasné stanovenie úloh v Trelle** - Za prvé dva šprinty sa niekoľkokrát vyskytol problém, že u stanovených úloh v Trelle sa nedalo presne určiť, čo všetko je potrebné urobiť, aby sa daná úloha mohla považovať za splnenú. Na tento problém sme navrhli rovno dve riešenia:
  - Pridanie stĺpčeka “splnené” pre úlohy v Trelle. V prípade, že riešiteľ považuje úlohu za hotovú, je potrebné, aby bolo riešenie skontrolované ešte iným členom tímu
  - Pridanie jednotlivých krokov k úlohám. Tento návrh sme sa rozhodli zatiaľ neaplikovať. V prípade, že by bol prvý návrh nedostatočný, môžeme aplikovať aj tento.
- 2. Podobné úlohy v Trelle** – Niektoré úlohy v Trelle sa nám počas predošlého šprintu obsahovo sčasti prekrývali. Tento problém je dosť závažný, keďže každú úlohu môže dostať iný člen tímu. Následkom je zníženie efektivity, pretože časť, ktorú majú úlohy spoločnú, bude vykonaná viackrát. Naším riešením na tento problém je, aby člen, ktorý bude pridávať novú úlohu zároveň aj pozeral na existujúce a porozmýšľal, či je nová úloha unikátna.

3. **Neskoré plnenie úloh** – Niektoré úlohy sa nám nepodarilo splniť v stanovenom čase. Hlavným dôvodom v našom tíme bola slabá komunikácia. U členov tímu je potrebné, aby kontaktovali ostatných v prípade, že si nevedia s niečím poradiť. Je tiež veľmi dôležité, aby každý celý tím vedel o priebežnom stave každého člena.
4. **Zlá organizácia úloh v Trelle** – Počas prvých dvoch šprintov sme úlohy v Trelle po splnení presunuli do stĺpčekov s názvami šprintov. Tento spôsob organizácie sa ukázal byť veľmi neprehľadný a spôsoboval problémy pri vytvorení Burndown chart. Navrhované riešenie bolo vytvoriť samostatné okno v Trelle na každý šprint a backlog.
5. **Žiadne úložisko pre pomocné materiály** - Pomocné materiály a odkazy, ktoré sme si navzájom preposielali, sme mali doteraz uložené iba v jednoduchom zozname na HipChate. S narastajúcim počtom takýchto materiálov sme sa rozhodli ich uložiť na náš spoločný GoogleDrive. Pomocné materiály budú teda zálohované a je možné ich aj kategorizovať podľa zložiek kvôli lepšej prehľadnosti.

## **Zhrnutie retrospektívy**

Celkovo do konca druhého šprintu sme sa neriadili niektorými pravidlami scrumu. Doteraz sme neriešili ani retrospektívu, riziká a ani product backlog. Programovať sme nezačali ešte ani v druhom šprinte hlavne kvôli dlhému čakaniu na existujúci materiál (3D UML prototypy z minulých rokov). Každý člen mal navyše pripomienky k celkovému riadeniu tímu. Všetky tieto pripomienky sme si spoločne prešli na spoločnom stretnutí a na každú sme navrhli riešenie, ktoré by malo zvýšiť celkovú efektivitu tímu. Nami navrhnuté riešenia sme sa rozhodli aplikovať v nasledujúcom šprinte.

## 8 Export úloh počas šprintov

Táto kapitola sumarizuje nami identifikované úlohy, na ktorých sme počas troch šprintov pracovali.

### 8.1 Úvodné úlohy

Prvé dva týždne sa celý náš tím zoznamoval a určoval si komunikačné nástroje. Snažili sme sa hneď od začiatku využívať program na manažovanie činností Trello. Stanovili sme si základné komunikačné nástroje ako HipChat, Gmail skupina, vymenili telefónne čísla. Začiatkové stretnutia boli skôr určené na špecifikáciu cieľov vedúceho práce. Uvažovali sme o stanovení novej architektúry, alebo zdokonalovaní existujúcich prototypov. Prvé stretnutia neobsahovali žiadne ohodnotenie zložitosti jednotlivých úloh, ktoré sme si stanovili. Tabuľka nižšie opisuje jednotlivé úlohy, ktoré sme vykonávali do prvého šprintu. Kolónka status ukazuje, či sme želané úlohy v danom šprinte splnili, alebo nie. Úlohy boli zadávané priebežne na stretnutiach produktovým vlastníkom a naším tímom. V prvých stretnutiach bolo najdôležitejšou súčasťou rozbehnutie prototypu sekvenčného diagramu.

#	Opis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Určenie času stretávania	Všetci členovia tímu	21.9.2015	OK
2.	Vytvorenie emailovej skupiny	Adam Kulišek	25.9.2015	OK
3.	Predbežné určenie funkcií členov tímu	Všetci členovia tímu	29.9.2015	OK
4.	Štúdium stránky predmetu TP	Všetci členovia tímu	25.9.2015	OK
5.	Webová stránka	Adam Kulišek	25.9.2015	OK



6.	Analýza dostupných vývojových prostredí, frameworkov, jazykov, nástrojov pre manažment úloh a komunikačných kanálov.	Všetci členovia tímu	1.10.2015	OK
7.	Určiť názov tímu, logo a plagát	Všetci členovia tímu	25.9.2015	OK
8.	Poslať email vedúcemu	Všetci členovia tímu	27.9.2015	OK
9.	Otestovanie funkčnosti pripojenia do spoločného repozitára na githube	Všetci členovia tímu	4.10.2015	OK
10.	Zápisnica pre prvý týždeň	Rami Mtier	4.10.2015	OK
11.	Prezentácia o 3D UML	Matej Jenis	6.10.2015	OK
12.	Doplniť fotky, zápisnice a info o členoch projektu	Adam Kulíšek	6.10.2015	OK
13.	Analýza Ruby a Python s grafickými knižnicami	Patrik Kolek, Tomáš Hnojčík, Rami Mtier	6.10.2015	OK
14.	Šablóna zápisnice	Adam Kulíšek	4.10.2015	OK
15.	Úspešne spustiť projekt sekvenčného diagramu	Matej Jenis	6.10.2015	Z časti splnené

Tab.4: Úvodné úlohy

## 8.2 Šprint 1

V porovnaní s predošlým určovaním úloh sme pridali ohodnotenie jednotlivých úloh. tento parameter boli človekohodiny. Snažili sme sa napodobniť metódotológiu agilného vývoja SCRUM<sup>12</sup>. Žiaľ spôsob, akým sme používali nástroj Trello, nám neposkytoval dôležitú funkcionalitu pre správne pracovanie v tíme podľa agile metódik. V prvom šprinte sme sa s produktovým vlastníkom dohodli na pretvorení existujúcich prototypov. Našimi primárnymi úlohami sa stalo rozbehnutie bakalárskeho projektu Mateja Jenisa, ktorý sme chceli spojiť s prototypom diagramu aktivít.

#	Opis úlohy	Zodpovedné osoby	Dátum dokončenia	Status	Odhadovaný čas	Celkový čas
1.	Zvoliť si konkrétne zameranie práce	Všetci členovia tímu	13.10.2015	OK	-	-
2.	Sprístupniť Git, rozposlať návod na inštaláciu projektu	Matej Jenis	9.10.2015	OK	-	-
3.	Stretnutie s Michalom Valovičom a Andrejom Železnákom za účelom predstavenia architektúry activity a sequence diagramu	Adam Kulíšek	13.10.2015	OK	-	-
4.	Identifikácia posledného releasu sequence diagramu na bitbuckete	Všetci členovia tímu	12.10.2015	OK	-	-
5.	Navrhnutie webovej stránky laboratória	Adam Kulíšek	13.10.2015	OK	-	-
6.	Zfunkčniť projekt na vlastnom počítači (sekvenčný diagram bez editačných)	Všetci členovia tímu	13.10.2015	OK	-	-

<sup>12</sup> <https://www.scrum.org/>

	funkcií)					
7.	Uloženie bakalárskej práce Mateja Jenisa na Bitbucket (sekvenčný diagram s editačnými funkciami)	Matej Jenis	16.10.2015	OK	1	0.5
8.	Vytvorenie obrázkového návodu pre rozbehanie prototypov	Matej Jenis	16.10.2015	OK	1	1
9.	Sfunkčniť projekt sekvenčného diagramu s editačnými funkciami a animáciu (bakalársky projekt Mateja Jenisa)	Všetci členovia tímu	20.10.2015	OK	13	13
10.	Dvojazyčná verzia stránky 3D Labu	Adam Kulíšek	19.10.2015	OK	1	1
11.	Nahodiť stránku tímu na virtuálny stroj	Adam Kulíšek, Rami Mtier	19.10.2015	OK	1	3
12.	Zverejnenie stránky 3D Labu	Adam Kulíšek	19.10.2015	OK	1	1

Tab.5: Šprint č. 1

### 8.3 Šprint 2

Podarilo sa nám identifikovať posledný funkčný prototyp diagramu aktivít z minuloročného tímového projektu. Šprint sme začali s pochopením a analýzou funkčného prototypu diagramu aktivít. Úlohy v stanovenom šprinte sú opísané v tabuľke nižšie.

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status	Odhadovaný čas	Celkový čas
1.	Rozbehanie prototypu minuloročného tímového projektu	Všetci členovia tímu	25.10.2015	OK	13	13

2:	Vytvoriť dvojice, ktoré budú spolupracovať na pridelených úlohách	Rami Mtier	24.10.2015	OK	0.5	0.5
3.	Prečítanie dokumentácie Andreja Železnáka ohľadom architektúry activity diagramu	Všetci členovia tímu	26.10.2015	OK	5	7
4.	Určiť tasky po prečítaní dokumentácie	Všetci členovia tímu	26:10.2015	OK	1	1
5.	Stanovanie manažérskych pozícií	Adam Kulíšek	23.10.2015	OK	0.5	0.5
6.	Rozpoznať čo sa deje v metódach aktivity diagramu	Všetci členovia tímu	3.11.2015	OK	-	-
7.	Preštudovať kompetencie rolí, ktoré sú každému pridelené	Všetci členovia tímu	30.10.2015	OK	1.5	2
8.	Štúdium Activity Diagram branche	Všetci členovia tímu	3.11.2015	OK	21	21
9.	Rozhodnúť kde pridať wrapper (Ogre, potom WebGL,...)	Všetci členovia tímu	25.9.2015	OK	-	-
10.	Vytvoriť Dokumentáciu k riadeniu vývoja	Rami Mtier	3.11.2015	OK	2	2

Tab.6: Šprint č. 2

## 8.4 Šprint 3

Vytvorili sme s produktovým vlastníkom produktový backlog. Určili sme si aké úlohy máme splniť prvý semester a aké druhý semester. Po scrum paneli sme už vedeli, že nesprávne odhadujeme zložitosť úloh a zmenili sme odhadovanie podľa story

pointov. Naša referenčná hodnota bola 5, tá značila zložitosť rozbehnutia prvého prototypu. Myslíme si, že sa v ďalšom šprinte zmení, pretože sme zatiaľ nemali dostatok relevantných úloh pre stanovenie referenčnej hodnoty.

#	Opis úlohy	Zodpovedné osoby	Dátum dokončenia	Status	User stories
1.	Prepísať class diagramy do MS Visia	Boris Buček, Tomáš Hnojčík, Patrik Kolek	10.11.2015	OK	-
2.	Pridať do menu elementy pre sekvenčný diagram	Matej Jenis	6.11.2015	OK	-
3.	Integrácia BitBucket repozitára s HipChatom	Adam Kulíšek	6.11.2015	OK	-
4.	Vytvorenie nového repozitára	Adam Kulíšek	6.11.2015	OK	2
5.	Pridať nefukčné funkcie (save load, fragment,...) do menu	Matej Jenis	9.11.2015	OK	3
6.	Zobrazenie prázdnej scény sekvenčného diagramu	Matej Jenis	9.11.2015	OK	2
7.	Používateľské menu pre sekvenčný diagram	Matej Jenis	9.11.2015	OK	3
8.	Modifikácia postupov pre lepšie potreby SCRUM-u	Tomáš Hnojčík	7.11.2015	OK	-
9.	Vloženie lifeline do sekvenčného diagramu v projekte				13
10.	Vloženie message do sekvenčného diagramu v projekte				13

Tab.7: Šprint č. 3

## 9 Tvorba stránky 3D Lab-u

V priebehu druhého týždňa semestra vznikla požiadavka na vytvorenie prezentačnej webovej stránky pre 3D Lab. Navrhli a vytvorili sme preto jednoduchú stránku so statickým obsahom, ktorej štruktúra sa zakladá na stránke webového sídla nášho tímového projektu.

Stránka bola vytvorená pomocou viacerých knižníc na prácu s kaskádovými štýlmi (CSS<sup>13</sup>), skriptovacím jazykom Javascript<sup>14</sup> a značkovacím jazykom HTML<sup>15</sup> ako napríklad Bootstrap<sup>16</sup> a Trianglify<sup>17</sup>. V úvode sme rozmýšľali túto stránku zverejniť na platforme GitHub Pages<sup>18</sup>, rovnako ako stránku nášho tímového projektu v druhom týždni semestra, no neskôr sme sa ju rozhodli nasadiť na známu PaaS<sup>19</sup> (platform as a service) Heroku<sup>20</sup>. Predmetom tejto stránky je prezentácia projektov, na ktorých členovia laboratória pracujú, predstavenie jednotlivých členov spolu so základnými informáciami o nich a zviditeľnenie laboratória verejnosti.

---

<sup>13</sup> <http://www.w3schools.com/css/>

<sup>14</sup> <http://www.w3schools.com/js/>

<sup>15</sup> <http://www.w3schools.com/html/>

<sup>16</sup> <http://getbootstrap.com/>

<sup>17</sup> <http://qrohlf.com/trianglify/>

<sup>18</sup> <https://pages.github.com/>

<sup>19</sup> <http://searchcloudcomputing.techtarget.com/definition/Platform-as-a-Service-PaaS>

<sup>20</sup> <https://www.heroku.com/>

# 10 Prílohy

## 10.1 Zápisnica 1

### Téma stretnutia

Úvodné stretnutie, určenie úvodných úloh
--

Dátum	21.9.2015
Miestnosť	Konferenčná miestnosť na 4. poschodí - FIIT STU
Vyhotovil	Bc. Adam Kulíšek

### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Patrik Kolek
	Bc. Adam Kulíšek	Bc. Rami Mtier

### Zhrnutie úloh z predchádzajúceho stretnutia:

-

### Priebeh stretnutia

V prvom rade sme sa zoznámili. Vedúci nás stručne oboznámil s projektom a uviedol pár oblastí, na ktorých by sme mohli pracovať. Upozornil nás na dôležité veci, ktoré máme ohľadom tímového projektu dodržiavať a sledovať. S jeho pomocou sme zostavili zoznam základných úloh, ktoré budeme musieť splniť do ďalšej prednášky z TP. Neskôr sme sa presunuli k stolom v hale na štvrtom poschodí, kde sme si určili základný komunikačný kanál (HipChat) a po analýze nástrojov pre manažment úloh sme sa predbežne dohodli na používaní služby Trello.

### Zoznam úloh

1. Určiť čas stretávania.
2. Vytvoriť emailovú skupinu, zistiť, či existuje nejaká konkrétna forma pre jej názov.
3. Predbežné určenie funkcií členov tímu.
4. Štúdium stránky predmetu TP, prejedenie všetkých deadlines.
5. Vytvoriť základnú stránku tímu, popozerať stránky tímov z predošlých rokov.

6. Analýza dostupných vývojových prostredí, frameworkov, jazykov, nástrojov pre manažment úloh a komunikačných kanálov.
7. Určiť názov tímu, vytvoriť logo a plagát.
8. Poslať email vedúcemu s týmito bodmi



## 10.2 Zápisnica 2

### Téma stretnutia

**Prvé oficiálne stretnutie – zoznamovanie sa s problémom**

Dátum	29.9.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Rami Mtier

### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Patrik Kolek
	Bc. Adam Kulišek	Bc. Rami Mtier

### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Určenie času stretávania	Všetci členovia tímu	21.9.2015	OK
2.	Vytvorenie emailovej skupiny	Adam Kulišek	25.9.2015	OK
3.	Predbežné určenie funkcií členov tímu	Všetci členovia tímu	29.9.2015	OK
4.	Štúdium stránky predmetu TP	Všetci členovia tímu	25.9.2015	OK
5.	Webová stránka	Adam Kulišek	25.9.2015	OK
6.	Analýza dostupných vývojových prostredí, frameworkov, jazykov,	Všetci členovia tímu	25.9.2015	Z časti splnené

	nástrojov pre manažment úloh a komunikačných kanálov.			
7.	Určiť názov tímu, logo a plagát	Všetci členovia tímu	25.9.2015	OK
8.	Poslať email vedúcemu	Všetci členovia tímu	27.9.2015	OK

### Priebeh stretnutia

Na druhom stretnutí sme zistili nezhody v komunikácií s vedúcim tímového projektu, ktoré sa rýchlo vyjasnili. Viedli sme spolu dialóg o povinnostiach a zatiaľ uskutočnenej práci. Vedúci práce nám ponúkol riešenia, ktoré by sme mohli vypracovať v rámci tímového projektu. Vyskytli sa viaceré alternatívy:

1. Zveľadenie aplikácie Mateja Jenisa z hľadiska kódu, funkcionality.
2. Implementácia nového 3D UML diagramu do už existujúcej architektúry a vybranie si dvoch funkcionalít z fragmentov, editovacích funkcií a animácií
3. Vytvorenie novej architektúry v jazykoch Python alebo Ruby a jednoduchej 3D UML aplikácie

Zvažovanými diagramami sa stali sekvenčný diagram, activity diagram a class diagram. S vedúcim práce sme sa dohodli na ďalšom postupe vo forme prezentovania bakalárskej práce Mateja Jenisa, ktorý má skúsenosti s 3D UML vývojom. Hlavným cieľom sa stalo delegovanie úloh v projekte a prezentácia, ktorá objasni 3D UML, nakoniec bude umiestnená aj na webovej stránke.

### Zoznam úloh

1. Vytvorenie zápisnice pre prvé stretnutie.
2. Vytvorenie zápisnice pre druhé stretnutie.
3. Vytvorenie prezentácie o 3D UML
4. Vytvorenie šablóny zápisnice
5. Doplnenie fotiek členov a zápisníc na webovú stránku tímu
6. Analyzovať prostredia v Ruby a Python s grafickými knižnicami
7. Vytvoriť dvojice pre jednotlivé roly
8. Spustiť bakalársky projekt Mateja Jenisa
9. Poslať email vedúcemu s týmito bodmi

## 10.3 Zápisnica 3

### Téma stretnutia

<b>3D UML a jeho možnosti</b>
-------------------------------

Dátum	6.10.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Patrik Kolek

### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Patrik Kolek
	Bc. Adam Kulíšek	Bc. Rami Mtier

### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Otestovanie funkčnosti pripojenia do spoločného repozitára na githube (pull, push)	Všetci členovia tímu	4.10.2015	OK
2.	Zápisnica pre prvý týždeň	Rami Mtier	4.10.2015	OK
3.	Prezentácia o 3D UML	Matej Jenis	6.10.2015	OK
4.	oplniť web o fotky, zápisnice a info o členoch projektu	Adam Kulíšek	6.10.2015	OK
5.	Analýza Ruby a Python s grafickými knižnicami	Patrik Kolek, Tomáš Hnojčík, Rami Mtier	6.10.2015	OK

6.	Šablóna zápisnice	Adam Kulíšek	4.10.2015	OK
7.	Úspešne spustiť projekt	Matej Jenis	6.10.2015	epodarilo sa

### **Priebeh stretnutia**

Na treťom stretnutí sme si lepšie predstavili problematiku 3D UML, jej prínosy a ciele (prezentáciou Mateja Janisa a Ivana Polaseka). Asi hlavným prínosom

3D UML je zjednodušenie veľkých komplexných modelov vyobrazených štandardným UML. Predbežne sme sa dohodli akým smerom sa bude uberať tímový projekt a naplánovali sme si úlohy ktoré vykonáme nasledujúci týždeň.

### **Zoznam úloh**

Stretnutie s Michalom Valovičom a Andrejom Zeleznakom za účelom predstavenia architektúry activity a sekvenčného diagramu.

Identifikovať posledný release sekvenčného diagramu na bitbuckete a oživiť ho v tíme (prototyp od Richarda Belana, Mateja Jenisa, ...)

Zamyslieť sa nad animáciou, editčnými funkciami, architektúrou.

Webová stránka Labáku (členovia, výsledky a zamyslieť sa čo by tam mohlo byť)

## 10.4 Zápisnica 4

### Téma stretnutia

**Stretnutie s ďalšími členmi 3D Uml projektu**

Dátum	13.10.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Boris Buček

### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Patrik Kolek
	Bc. Adam Kulíšek	Bc. Rami Mtier

### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Zvoliť si konkrétne zameranie práce (animácia, editačné funkcie, nová architektúra)	Všetci členovia tímu	13.10.2015	OK
2.	Sprístupniť GIT, rozposlať návod na inštaláciu projektu	Matej Jenis	9.10.2015	OK
3.	Stretnutie s Michalom Valovičom a Andrejom Železnákom za účelom predstavenia architektúry activity a sequence diagramov	Adam Kulíšek	13.10.2015	OK

4.	Identifikácia posledného releasu sequence diagramu na bitbuckete	Všetci členovia tímu	12.10.2015	OK
5.	Navrhnutie webovej stránky laboratória	Adam Kulíšek	13.10.2015	OK
6.	Zfunkčniť projekt na vlastnom počítači	Všetci členovia tímu	13.10.2015	Z časti splnené

### Priebeh stretnutia

Na štvrtom stretnutí sme sa v 3D labe stretli s Richardom, Igorom a Andrejom, s ktorými sme sa porozprávali o našich možnostiach a dohodli sme sa na smerovaní projektu. Naším prvým cieľom bude teda spojenie jednotlivých projektov do jedného, ktoré potom upravíme podľa Andrejovej verzie projektu. Richard nám predstavil svoj prototyp 3D UML, ktorý bol vytvorený v HTML5, CSS3 a JavaScripte. Jednotlivé elementy v prototypu boli veľmi dobre viditeľné a ľahko manipulovateľné cez HTML.

Ďalej sme sa s Richardom, Igorom a Andrejom dohodli, že v prípade nových nápadov nás kontaktujú, a že im budeme písať ohľadne informácií pre webovú stránku 3D labu.

### Zoznam úloh

1. Rozbehanie VirtualBoxu
2. Vytvoriť obrázkový tutorial na rozbehanie prototypu a uverejniť ho na google grupe
3. Nahodenie bakalárskej práce Mateja Jenisa na bitbucket
4. Zfunkčniť projekt so sekvenčným diagramom, editovacími funkciami a animáciou (Matejova bakalárska práca)
5. Nahodiť stránku na virtuálny stroj
6. Vytvoriť dvojjazyčnú verziu stránky 3D labu

## 10.5 Zápisnica 5

### Téma stretnutia

<b>Predstavenie novej 3D UML architektúry</b>
---

Dátum	20.10.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Tomáš Hnojčík

### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Patrik Kolek
	Bc. Adam Kulíšek	Bc. Rami Mtier

### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Uloženie bakalárskej práce Mateja Jenisa na Butbucket	Matej Jenis	16.10.2015	OK
2.	Vytvorenie obrázkového návodu pre rozbehanie prototypu	Matej Jenis	16.10.2015	OK
3.	Sfunkčniť projekt sekvenčného diagramu s editačnými funkciami a animáciou (bakalárska práca Mateja Jenisa)	Všetci členovia tímu	20.10.2015	Z časti splnené
4.	Dvojjazyčná verzia stránky 3D Labu	Adam Kulíšek	19.10.2015	OK
5.	Nahodiť stránku tímu na virtuálny stroj	Adam Kulíšek, Rami Mtier	19.10.2015	OK

6.	Zverejnenie stránky 3D Labu	Adam Kulíšek	19.10.2015	OK
----	-----------------------------	--------------	------------	----

### **Priebeh stretnutia**

Na začiatku piateho stretnutia sme si pripomenuli možnosti smerovania projektu ako aj nové technológie, ktoré boli prezentované chalanmi z 3D Labu na poslednom stretnutí. Rovnako nám bola poskytnutá nová architektúra, ktorá bola použitá pri implementácii Activity diagramu. Dôkladne sme si túto architektúru analyzovali a definovali ciele, ktoré je potrebné splniť v rámci predmetu. Dohodli sme sa na spojení sekvenčného diagramu s editačnými funkciami a animáciou spolu s fragmentami do jedného projektu. Následne tento kompletný sekvenčný diagram potom pretransformujeme do novo predstavenej architektúry.

Po vykonaní tejto transformácie je možné pokračovať v implementácii existujúceho projektu (sekvenčného diagramu) v rámci iných technológií napr. HTML5, CSS3 a JavaScript.





**Obr. 1** – Architektúra Activity diagramu

## Zoznam úloh

1. Vytvoriť dvojice, ktoré budú spolu pracovať na pridelených úlohách pri riešení a implementácii projektu.
2. Prečítať dokument ohľadne novej architektúry použitej v activity diagrame.
3. Určiť ďalšie úlohy po prečítaní dokumentu
4. Analyzovať prvé kroky pre migráciu sekvenčného diagramu s editačnými funkciami do novej architektúry
5. Rozbehať projekt TP Activity fragment z minulého roku

## 10.6 Zápisnica 6

### Téma stretnutia

**Activity diagram a jeho štruktúra**

Dátum	27.10.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Matej Jenis

### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Patrik Kolek
	Bc. Adam Kulíšek	Bc. Rami Mtier

### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Rozbehanie prototypu minuloročného tímového projektu (Branch_TP_2015_activity_fragment)	Všetci členovia tímu	25.10.2015	OK
2.	Vytvoriť dvojice, ktoré budú spolupracovať na pridelených úlohách	Rami Mtier	24.10.2015	OK
3.	Prečítanie dokumentácie od Andreja Železnáka ohľadom architektúry aktivity diagramu	Všetci členovia tímu	26.10.2015	OK
4.	Určiť tasky po prečítaní dokumentácie	Všetci členovia tímu	26.10.2015	OK
5.	Stanovenie manažérskych rolí	Adam Kulíšek	23.10.2015	OK

## **Priebeh stretnutia**

Na stretnutí sme riešili predovšetkým štruktúru aktivity diagramu. Preberali sme spôsob akým komunikuje metamodel s controllerom a ako komunikujú metamodel\_controller a 3Dcontroller navzájom. Riešili sme taktiež možnosť pridania wrappera do aktivity diagramu, keďže neskôr chceme nahradiť grafickú knižnicu Ogre knižnicou WebGL. Na záver stretnutia sme si vyjasnili pozície v tíme a začali vytvárať dokumentáciu k riadeniu projektu.

## **Zoznam úloh**

1. Rozpoznať čo sa deje v metódach aktivity diagramu
2. Rozhodnut kde pridať wrapper (Ogre, potom WebGL,...)
3. Vytvoriť Dokumentáciu k riadeniu vývoja
1. Preštudovať kompetencie rolí, ktoré sú každému pridelené
5. Študovať architektúru a princípy Activity Diagramu

## 10.7 Zápisnica 7

### Téma stretnutia

#### Špecifikácia projektu

Dátum	3.11.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Rami Mtier

#### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Adam Kulíšek
	Bc. Rami Mtier	

#### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Rozpoznať čo sa deje v metódach aktivity diagramu	Všetci členovia tímu	3.11.2015	Z časti splnené
2.	Preštudovať kompetencie rolí, ktoré sú každému pridelené	Všetci členovia tímu	30.10.2015	OK
3.	Štúdium Activity Diagram branche	Všetci členovia tímu	3.11.2015	Z časti splnené
4.	Rozhodnúť kde pridať wrapper (Ogre, potom WebGL,...)	Všetci členovia tímu	25.9.2015	Z časti splnené
5.	Vytvoriť Dokumentáciu k riadeniu vývoja	Rami Mtier	3.11.2015	Z časti splnené

#### Priebeh stretnutia

Paťo je tento týždeň chorý, a teda sa nemohol zúčastniť dnešného stretnutia. Na stretnutí sme si s vedúcim projektu ujasnili smer, ktorým sa chceme hýbať a ujasnili si rozdelenie práce do dvoch semestrov. Cieľom dvoch semestrov bude tri prototypy spojiť do funkčného výsledku. Prvý semester sa budeme zaoberať transformáciou bakalárskeho projektu Mateja Jenisa, čo predstavuje prototyp 3D sekvenčného diagramu, do výstupu z tímového projektu 3D UML z roku 2014/2015, čo je projekt s funkčným activity 3D diagramom. Odkonzultovali sme špecifikáciu projektového backlog-u, určili akým štýlom bude stavaný a špecifikovali user stories, Prediskutovali sme postup práce s repositárom, gitom a trelloom, aby správne odrážal priebeh SCRUM-u. Určili sme si tiež ďalší postup pri analýze projektu z konzultácie a celého stretnutia sme sa venovali nasledujúcim bodom:

1. Retrospektíva – vyšpecifikovali sme nejasnosti v určovaní taskov a komunikácií a managementu, zohľadňovali sme v retrospektíve aj riziká.
2. Vytvorili sme produktový backlog.
3. Vytvorili sme jednotlivé user stories a ohodnotili sme ich. Ako referenčnú hodnotu sme použili prácu, ktorú sme vynaložili na rozbehnutie troch prototypov aplikácie.
4. Určili sme si úlohy na jeden šprint – 18 story pointov, ktoré budeme vykonávať nasledujúce dva týždne:
  - Zobrazenie prostredia prázdnej scény – 2 story pointov
  - Dotvorenie používateľského menu - 3 story pointov
  - Vkladanie elementov do scény - 13 story pointov
5. Vkladanie elementov má vysoký počet story pointov, ale nevedeli sme to korektne rozdeliť na menšie časti.
6. Určili si ďalší postup pri analýze projektu.

### **Zoznam úloh**

1. Vytvorenie zápisnice zo stretnutia.
2. Vytvorenie vlastného repositára a jeho nastavenie.
3. Modifikácia postupov v trelle pre lepšie potreby SCRUM-u.
4. Dotvorenie analýzy projektu.
5. Zjednotenie a zdieľanie class diagramov.

## **Príloha A - Product Backlog**

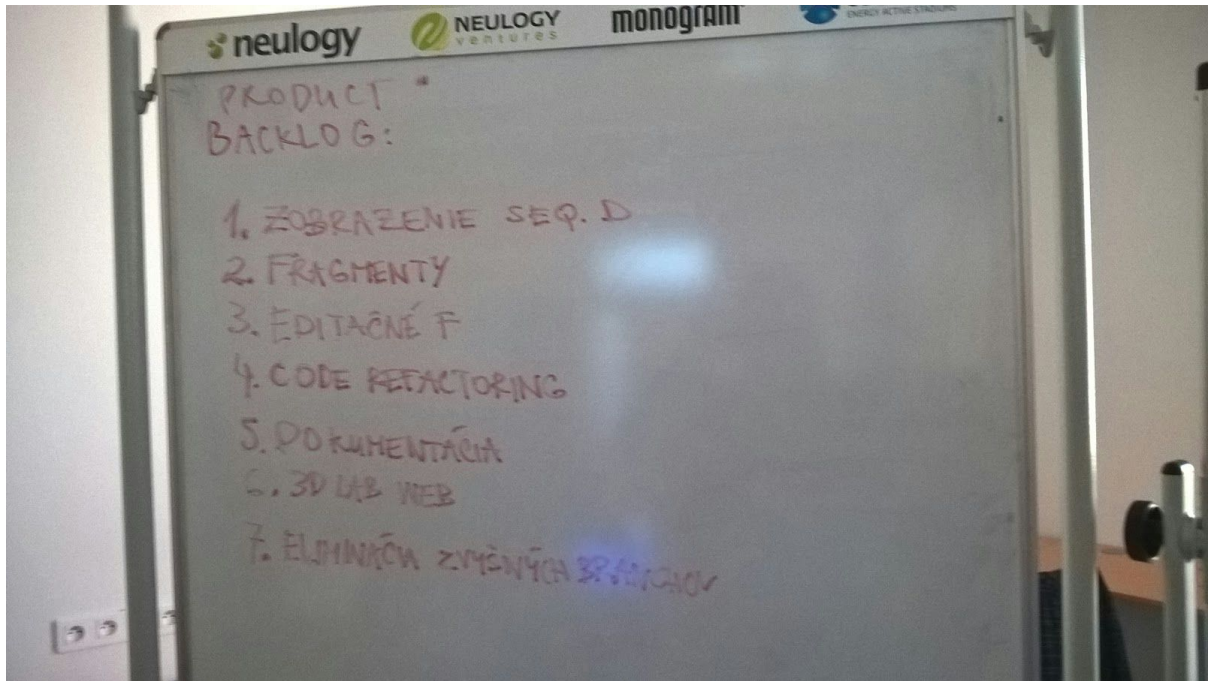
### **Big Picture**

1. Zobrazenie sekvenčného diagramu v prostredí a architektúre aktivity diagramu
2. Dorobenie fragmentov do novej architektúry sekvenčného diagramu

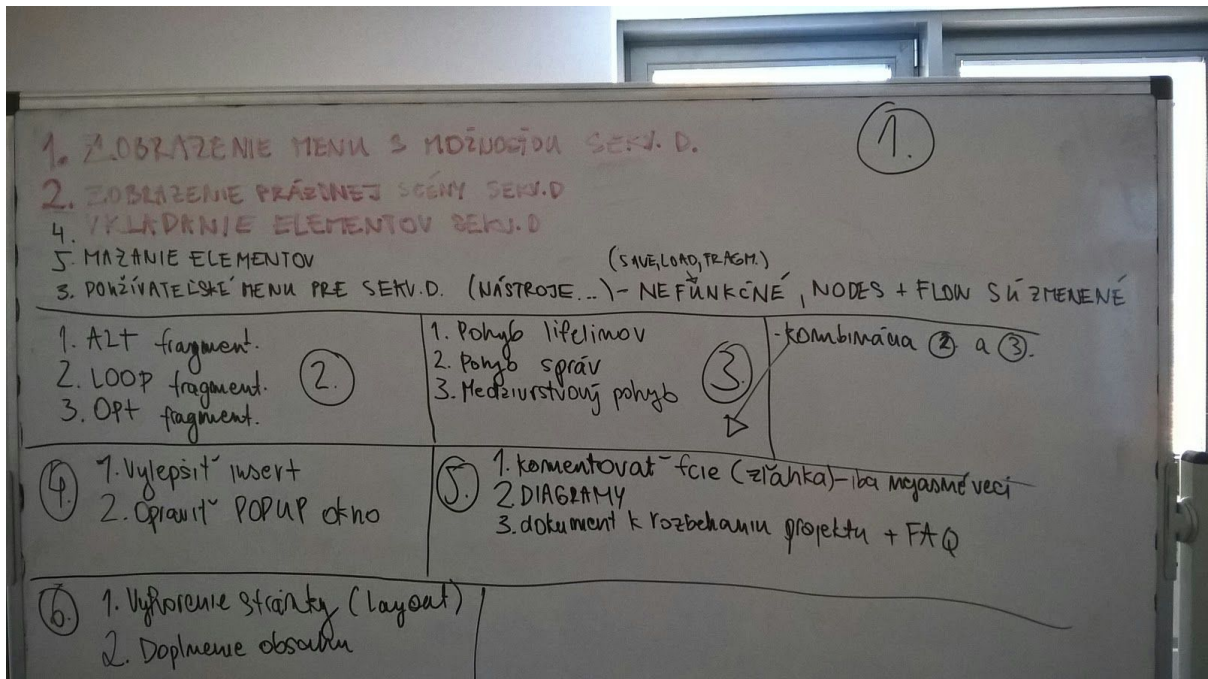
3. Dorobenie editačných funkcií (Letný semester)
4. Code refactoring (Vylepšenie insertu, lepšie komentáre, zrušenie nepotrebných namespaceov,...)
5. Podrobná dokumentácia (Class diagram zachytávajúce štruktúru prototype = keď budú hotové sharnem link)
6. 3D LAB WEB PAGE
7. Eliminácia branchov na bitbuckete (z troch vznikne jeden)

### **User stories jednotlivých položiek**

1. Zobrazenie sekvenčného diagramu v prostredí a architektúre aktivity diagramu
  - Oživenie tlačidla sekvenčného diagramu v úvodnom menu
  - Zobrazenie scény
  - Zobrazenie menu obsahujúce elementy sekvenčného diagram (zatiaľ nefunkčné)
  - Vkladanie elementov
  - Mazanie elementov
2. Dorobenie fragmentov do novej architektúry sekvenčného diagramu
  - Pridanie ALT fragmentu
  - Pridanie OPT fragmentu
  - Pridanie LOOP fragment
3. Dorobenie editačných funkcií (Letný semester)
  - Pohyb lifelineov
  - Pohyb asynchrónnou správou
  - Medzivrstvový pohyb
4. Code refactoring
  - Vylepšiť insert
  - Opraviť Popup okno
  - Sprehľadniť kód
5. Podrobná dokumentácia
  - Stručné, výstižné komentáre k funkciám
  - Class diagram štruktúry prototype
  - Dokument k rozbehaniu prototype + FAQ
6. 3D LAB WEB PAGE
  - Vytvoriť stránku
  - Doplnenie obsahu



Obr. 1 – Big picture



Obr. 2 – Používateľské príbehy

## 10.8 Zápisnica 8

### Téma stretnutia

<b>Diskusia o forme dokumentácie k inžinierskemu dielu</b>
--

Dátum	10.11.2015
Miestnosť	Jobsovo softvérové štúdio (FIIT STU)
Vyhotovil	Bc. Boris Buček

### Prítomní členovia:

Vedúci projektu	Ing. Ivan Polášek, PhD.	
Členovia tímu	Bc. Boris Buček	Bc. Tomáš Hnojčík
	Bc. Matej Jenis	Bc. Patrik Kolek
	Bc. Adam Kulíšek	Bc. Rami Mtier

### Zhrnutie úloh z predchádzajúceho stretnutia:

#	Popis úlohy	Zodpovedné osoby	Dátum dokončenia	Status
1.	Prepísať class diagramy pre Core do MS Visia	Boris Buček, Patrik Kolek, Tomáš Hnojčík	10.11.2015	OK
2.	Pridať do menu elementy pre sekv. Diagram	Matej Jenis	9.11.2015	OK
3.	Integrácia BitBucket repozitára s HipChatom	Adam Kulíšek	6.11.2015	OK
4.	Vytvorenie nového repozitára a master branchu	Adam Kulíšek	6.11.2015	OK
5.	Pridať nefunkčné funkcie (save, load, fragm.) do menu	Matej Jenis	9.11.2015	OK



6.	Zobrazenie prázdnej scény pre sekvenč. diagram	Matej Jenis	9.11.2015	OK
7.	Používateľské menu pre sekvenč. diagram	Matej Jenis	9.11.2015	OK
8.	Modifikácia postupov v Trelle pre lepšie potreby SCRUM-u	Tomáš Hnojčík	7.11.2015	OK

## Priebeh stretnutia

Na stretnutí sme si spolu s vedúcim zrekapitulovali a predviedli sme, čo všetko sa stihlo do tohto stretnutia naprogramovať / nakresliť vo forme diagramov v MS Office Visio. Je potrebné rozumne zrefaktorovať existujúci projekt (výstup tímového projektu 3D UML z 2014/2015), aby bol lepšie čitateľný, keď ho po nás preberie nasledujúci tím.

Vytvorili sme si vlastnú DoD (Definition of Done) pre naše user stories, kedy budú z nášho pohľadu úspešne dokončené.

Ďalej sme diskutovali o forme dokumentácie k inžinierskemu dielu. Dohodli sme sa s vedúcim, ktoré veci z programovacej stránky je potrebné dokončiť do budúceho týždňa, aby sa nachádzali aj v dokumentácii. Mali by sme sa zamerať hlavne na technickú dokumentáciu. V dokumentácii je potrebné, aby boli informácie o stave sekvenčného diagramu. Rozhodli sme sa túto časť rozdeliť na 4 modely :

1. Pôvodný stav diagramu pred našou zmenou
2. Stav po prechode diagramu do novej architektúry
3. Ako by to malo v budúcnosti vyzeráť
4. Návrh na wrappery

K jednotlivým obrázkom bude samozrejme potrebné pridať aj technický opis. V prípade, že by sme chceli, tak môžeme pridať do dokumentácie aj dáke informácie o kombinovaných fragmentoch. Rozsiahle obrázky by mali byť vektorové, aby sa dali lepšie čítať.