

# Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

---



## 3D UML

*Dokumentácia k riadeniu*

*Tím č. 4*

---

Tím: *Bc. Hana Baranovičová, Bc. Francisc Juraš, Bc. Miroslav Kudláč, Bc. Lukáš Markovič, Bc. et Bc, Martin Melis, Bc. Michal Valovič, Bc. Andrej Železnák*

Vedúci tímu: *Ing. Ivan Polášek, PhD.*

Študijný program: *Informačné systémy/Softvérové inžinierstvo*

Predmet: *Tímový projekt I*

Akademický rok: *2014/2015, zimný semester*

# Obsah

1. Úvod .....	5
2. Roly členov tímu a podiel práce .....	6
2.1. Rozdelenie manažérskych činností.....	6
2.2. Ostatné roly a úlohy členov tímu.....	6
2.3. Podiel práce na jednotlivých častiach dokumentácie .....	7
2.3.1. Dokumentácia k riadeniu.....	7
2.3.2. Dokumentácia k inžinierskemu dielu .....	8
3. Aplikácie manažmentov .....	9
3.1. Manažment rozvrhu, plánovania .....	9
3.2. Manažment kvality a monitorovania .....	9
3.3. Manažment komunikácie.....	10
3.3.1. Komunikácia na sociálnej sieti.....	10
3.3.2. Komunikácia v rámci stretnutia.....	11
3.4. Manažment riadenia .....	11
3.5. Manažment dokumentovania.....	11
3.6. Manažment vývoja a integrácie.....	12
3.7. Hlavný architekt .....	12
3.7.1. Manažment testovania .....	13
3.8. Manažment rizík.....	13
4. Sumarizácia šprintov .....	14
4.1. Šprint č. 1 .....	14
4.1.1. Úlohy v šprinte č. 1 .....	15
4.2. Šprint č. 2 .....	15
4.2.1. Úlohy v šprinte č. 2 .....	16
4.3. Šprint č. 3 .....	16

4.3.1. Úlohy pre šprint 3 .....	17
4.4. Šprint č. 4 .....	17
4.4.1. Úlohy v šprinte č. 4 .....	18
4.5 Šprint č.5 .....	18
4.5.1 Úlohy v šprinte č. 5 .....	19
5. Používané metodiky .....	20
5.1. Metodika: Coding standard .....	20
5.2. Metodika: Code review .....	20
5.3. Metodika: Dokumentovanie .....	20
5.4. Metodika: Evidencia úloh.....	20
5.5. Metodika: Komunikácia .....	21
5.6. Metodika: Testovanie .....	21
5.7. Metodika: Verziovanie .....	21
6. Globálna retrospektíva ZS.....	22
6.1. Retrospektíva prvého šprintu.....	22
6.1.1. Priebeh šprintu.....	22
6.1.2. Úlohy šprintu.....	22
6.1.3. Velocity chart .....	23
6.1.4. Burndown chart .....	24
6.2. Retrospektíva druhého šprintu.....	24
6.2.1. Priebeh šprintu.....	24
6.2.2. Úlohy šprintu.....	25
6.2.3. Velocity chart .....	26
6.2.4. Burndown chart .....	26
6.3. Retrospektíva tretieho šprintu.....	28
6.3.1. Priebeh šprintu.....	28
6.3.2. Úlohy šprintu.....	28

6.3.3. Burndown chart .....	30
6.3.4. Velocity chart .....	30
6.4. Retrospektíva 4. šprintu.....	31
6.4.1. Priebeh šprintu.....	31
6.4.2. Úlohy šprintu.....	32
6.4.3. Burndown chart .....	33
6.4.4. Velocity chart .....	34
6.5. Retrospektíva 5. šprintu.....	34
6.5.1. Priebeh šprintu.....	34
6.5.2. Úlohy šprintu.....	35
6.5.3. Burndown chart .....	36
6.5.4. Velocity chart .....	36
6.6. Zhrnutie retrospektív .....	37
A. Príloha: Preberací protokol.....	1
B. Príloha: Kompetencie tímu.....	1
B.1. Tím.....	1
B.2. Téma „3D UML“ .....	3
B.3. Rozvrh.....	4

# 1. Úvod

Tento dokument predstavuje dokumentáciu k riadeniu projektu s názvom 3D UML k predmetu Tímový projekt, ktorý je vypracovávaný na Fakulte informatiky a informačných technológií na Slovenskej univerzite v Bratislave. Členmi tímu sú: Bc. Miroslav Kudláč, Bc. Andrej Železnák, Bc. Francisc Juraš, Bc. Lukáš Markovič, Bc. Michal Valovič, Bc. et Bc, Martin Melis, Bc. Hana Baranovičová. Vedúcim tímu je Ing. Ivan Polášek, PhD.

Druhá kapitola dokumentu obsahuje podrobne popísané krátkodobé a dlhodobé roly a úlohy jednotlivých členov tímu a autorstvo dokumentácie k riadeniu a dokumentácie k inžinierskemu dielu. Tretia kapitola obsahuje popis aplikácií manažmentu. V štvrtej kapitole je sumarizácia šprintov a v piatej kapitole sú načrtnuté používané metodiky spolu s odkazmi na kompletne znenie metodík na webe tímového projektu. Šiesta kapitola obsahuje globálnu retrospektívu projektu. Dokument obsahuje ako prílohu preberacie protokoly.

Obsah tohto projektu a jeho implementácia sú bližšie popísané v Dokumentácii k inžinierskemu dielu.

## 2. Roly členov tímu a podiel práce

### 2.1. Rozdelenie manažérskych činností

Manažérske roly sme po uvážení rozdelili nasledovne:

<b>Vedúci tímu</b>	Miroslav Kudláč
<b>Manažér rozvrhu a plánovania</b>	Francisc Juraş
<b>Manažér komunikácie a ľudských zdrojov</b>	Martin Melis
<b>Manažér rizík</b>	Martin Melis
<b>Manažér rozsahu projektu</b>	Michal Valovič
<b>Manažér vývoja a integrácie</b>	Lukáš Markovič
<b>Manažér obstarávania a nákladov na projekt</b>	Hana Baranovičová
<b>Manažér kvality a monitorovania projektu</b>	Francisc Juraş
<b>Manažér dokumentovania</b>	Michal Valovič

### 2.2. Ostatné roly a úlohy členov tímu

Dlhodobé úlohy:

<b>Hlavný architekt</b>	Andrej Železnák
<b>Správca stránky</b>	Miroslav Kudláč
<b>Testovanie</b>	Andrej Železnák
<b>Návrh a implementácia metamodelu</b>	Andrej Železnák Participujú: Francisc Juraş, Hana Baranovičová, Martin Melis
<b>Návrh a implementácia GUI</b>	Lukáš Markovič Participujú: Michal Valovič, Miroslav Kudláč
<b>Zápis zo stretnutí</b>	Miroslav Kudláč

Krátkodobé úlohy:

Vytvorenie web stránky	Michal Valovič Participujú: Miroslav Kudláč
Návrh metamodelu	Martin Melis
Konfigurácia serveru tímu	Andrej Železnák
Vytvorenie prezentácie	Martin Melis
Vytvorenie dokumentácie pre inžinierske dielo	Michal Valovič
Vytvorenie dokumentácie pre riadenie	Hana Baranovičová
Návrh grafiky elementov	Andrej Železnák
Vylepšovanie funkcií INSERT a DELETE	Andrej Železnák
Vytvorenie GUI	Hana Baranovičová

## 2.3. Podiel práce na jednotlivých častiach dokumentácie

### 2.3.1. Dokumentácia k riadeniu

Retrospektívy (každý šprint): Miroslav Kudláč

Aplikácie manažmentov:

- Manažment rozvrhu, plánovania: Francisc Juraš
- Manažment kvality a monitorovania: Lukáš Markovič
- Manažment komunikácie: Martin Melis
- Manažment riadenia: Miroslav Kudláč
- Manažment dokumentovania: Michal Valovič
- Manažment vývoja a integrácie: Hana Baranovičová
- Hlavný architekt: Andrej Železnák

Metodiky:

- code review: Lukáš Markovič
- coding standard: Hana Baranovičová
- testovanie: Andrej Železnák
- dokumentovanie: Michal Valovič
- evidencia úloh: Francisc Juraš
- komunikácia: Martin Melis



- verziovanie: Miroslav Kudláč

Zoznam kompetencií tímu: všetci členovia tímu

Export evidencie úloh (každý týždeň): Francisc Juraş

## 2.3.2. Dokumentácia k inžinierskemu dielu

Hlavný autor: Michal Valovič

Revidoval: Lukáš Markovič

Analýza fragmentov: Lukáš Markovič

Analýza aktivity diagramu: Martin Melis

Analýza súčasného riešenia: Miroslav Kudláč

Návrh: Martin Melis

Príručka: Lukáš Markovič



## 3. Aplikácie manažmentov

### 3.1. Manažment rozvrhu, plánovania

V rámci plánovania a rozvrhu sa v našom tímovom projekte využívame nástroj *Jira*. Keďže sa v projekte vyvíja pomocou prístupu Scrum, je vývoj rozdelený do jednotlivých šprintov. Šprinty v *Jire* sú vytvárané nasledovne:

Product owner na pravidelných stretnutiach spolu s nami rozhoduje o úlohách, ktoré sú nasledujúci šprint prioritou. Podľa priority sa tieto úlohy zoradia v backlogu. Následne úlohy ohodnotíme pomocou Fibonacciho postupnosti podľa ich náročnosti, zatiaľ bez kartičiek. Snažíme sa dohodnúť na rovnakom čísle, ale ak má viac ľudí rôzne názory na ohodnotenie tasku, veľké slovo má člen tímu, ktorému úlohu následne priradíme, tiež na základe všeobecnej dohody a tak, aby boli úlohy rozdelené rovnomerne medzi všetkých v každom šprinte. Následne sa vloží ohodnotenie do Jiry, toto robí pri každom stretnutí Francisc, keďže je v rámci tímu hlavný správca Jiry. Do šprintu sa nesmie pridať úloha, ktorá nie je ohodnotená.

Francisc má tiež na starosti zastavovanie a spúšťanie šprintu. (raz za dva týždne). Ideálny stav by bol, keby sa úlohy zadávali iba na začiatku šprintu, ale to sa nám nikdy nepodarilo, lebo sa vždy po čase objavajú nové problémy, ktoré je nutné urgentne riešiť.

### 3.2. Manažment kvality a monitorovania

Pre rýchly a kvalitný vývoj je v rozsiahlych projektoch nutné sa zamerať na kvalitu vývojových procesov. Vhodným nástrojom na zlepšovanie kvality projektu je aj monitorovanie. Aby toto bolo možné, využívame správne nástroj Jiry. Snažíme sa, aby každý člen tímu dával svoje tasky do statusu „In progress“ zakaždým, keď na danej úlohe pracuje a do statusu „Resolved“ alebo „Closed“ keď úlohu dokončí. Chvíľu trvalo, kým sme si na to všetci zvykli, tak vznikalo pár nezrovnalostí. Aby sme sa toho vyvarovali, Francisc spolu s Miroslavom kontrolujú stavy taskov, a ak niektoré nie sú v stave, v ktorom by mali byť, alebo sa dlho neriešia, alebo sú prídlho v stave „In progress“, Miroslav ako vedúci tímu sa kontaktuje s členom tímu, ktorému je task priradený a pýta sa na progres, poprípade sa snaží urgovať prácu na tasku.

V prípade, že člen tímu už má vyriešené všetky svoje tasky, a nemá viac v sekcii „TO DO“ v Jire, ale sú tam tasky, ktoré nie sú pridelené nikomu, môže si ich zapísať pod seba a začať riešiť. Zvyčajne sa ale snažíme priradiť všetky vytvorené tasky.

Ďalšou metodikou použitou pri snahe o udržanie kvality v rámci kódu je uniformný coding standard. Toto je potrebné preto, aby bol všetok kód ľahko čitateľný a udržateľný. Dosahujeme to tým, že všetky súbory píšeme s rovnakým logickým odsadzovaním, premenné, metódy a triedy nazývame a Viac ku coding standard je v podkapitole [Metodika: Coding standard](#).

Toto všetko sa ešte následne kontroluje pomocou nástroja CODAN, viac informácií v podkapitole [Metodika: Code review](#).

### 3.3. Manažment komunikácie

Komunikácia medzi členmi prebieha najmä na formálnych stretnutiach, alebo pomocou sociálnej siete.

#### 3.3.1. Komunikácia na sociálnej sieti

Naša komunikácia mimo formálnych stretnutí prebieha hlavne pomocou sociálnej siete Facebook. Pre naše potreby máme vytvorenú súkromnú skupinu, kde sú iba členovia tímu. Slúži na hromadné oznamy, alebo otázky, na hromadné prekonzultovanie nejakého problému. V tejto skupine preberáme iba problémy týkajúce sa tímového projektu, žiadne iné. Cieľom takéhoto typu komunikácie je jednoduché archivovanie správ a prehľadnosť pri listovaní.

Súčasťou komunikácie je aj pravidelné podávanie informácie o statuse svojej práce. Toto reportovanie prebieha pravidelne v sobotu po oficiálnom vyhlásení vedúceho tímu. Úlohou každého člena tímu je na vypísaný status reagovať tak, že napíše stav riešenia v akom je a prípadne úlohami, ktoré ho ešte čakajú na doriešenie k najbližšiemu tímovému stretnutiu. Popríklad vysvetlí, prečo je pozadu, ak má nejaké problémy.

### 3.3.2. Komunikácia v rámci stretnutia

V rámci formálneho stretnutia má štruktúrovanú podobu. Stretnutie sa skladá z 3 základných častí:

- Úvodná rozprava tímu
- Stretnutie s product-ownerom
- Prehodnotenie úloh, snaha o pochopenie operačných cieľov, konkrétne rozdelenie úloh

V rámci úvodnej rozpravy sa očakáva od každého z členov tímu príprava podrobného reportu, ktorý bude sumarizovať vykonanú prácu od posledného stretnutia, jej výstupy a hodnotenie. V rámci tejto fázy sa očakáva vystúpenie každého, komu bola pridelená aspoň jedna úloha. Vzniknuté problémy sa operačne riešia.

V druhej fáze ide predovšetkým o predstavenie aktuálneho stavu product-ownerovi. Toto je úlohou vedúceho tímu, ktorý sa pokúsi zosumarizovať všetky statusy poskytnuté členmi tímu. Product-owner má následne prejav o jeho aktuálnych predstavách.

Po odchode product-ownera prichádza k neformálnej otvorenej diskusií a rozdelení nových úloh.

## 3.4. Manažment riadenia

Vývojový proces projektu *Bubbles* prebieha pomocou metodiky *SCRUM*. V každom šprinte určí vedúci tímu a manažér vývoja a integrácie ciele pre daný šprint, ktoré sa potom tím spoločnými silami snaží realizovať. Tieto ciele sú stanovené po konzultácií s Product managerom projektu. Manažér plánovania pomáha s rozbíjaním daných cieľov na úlohy. Tieto úlohy sa následne ohodnotia pomocou scrum pokeru a ak celková hodnota úloh presahuje kapacitu tímu, úlohy s menšou prioritou sa presunú do nasledujúceho šprintu. Vedúceho tímu informuje manažér vývoja o prograse na úlohách, ktoré sa nestihajú. Tieto problémy redukuje tak, že včas odhalíme riziká a riešime ich. Nesplnené úlohy sa presunú do nasledujúceho šprintu, kde majú najvyššiu prioritu.

Taktiež vedúci tímu dohliada na plnenie úloh počas šprint, resp. sleduje *Burndown chart* a reaguje na prípadné anomálie tohto grafu.

## 3.5. Manažment dokumentovania

Proces vývoja tímového projektu je potrebné dokumentovať. Manažér dokumentácie dohliada na tvorbu dokumentácií priebežne vytváraných počas šprintoch, ale i na dokumentovanie zdrojového kódu. Vytvára šablóny a upravuje dokumenty tak, aby bola dodržaná rovnaká identita pre všetky dokumenty v tíme.

Dokumentácia inžinierskeho diela je vytváraná na konci každého šprintu. Zozbierané výstupy od jednotlivých členov tímu manažér dokumentácie spracuje podľa pravidiel a šablóny uvedenej v [metodike dokumentovania](#). Po vytvorení dokumentácie je nutné, aby bola sprístupnená ostatným členom tímu cez vlastný zdieľaný adresár. Každý člen tímu sa môže k dokumentácií vyjadriť a je potrebné, aby každý člen si ju vo vlastnom záujme prečítal. Dohodnutým nástrojom na vytváranie dokumentácie je MS Word a ukladaný formát je *.docx*.

Dokumentácia musí byť dostupná po kontrolnom bode, teda po každom šprinte spravidla do 3 dní. Viac ako 3 dni musia mať svoje opodstatnenie, ale v prípade časového oneskorenia zo strany manažéra dokumentácie je potrebné, aby požiadal zvyšných členov tímu o podporu, prípadne rozhodne o tom vedúci tímu.

Dokumentácia je ihneď zverejnená na webovej stránke tímu a tiež uchovávaná kópia na zdieľanom úložisku tímu a centrálnom repozitári, pokiaľ tím používa centrálny repozitár na verziovanie zdrojového kódu.

### 3.6. Manažment vývoja a integrácie

Úloha manažéra podpory vývoja a integrácie pozostáva z manažmentu verzií, konfigurácií softvérového systému a integrácie použitých súčastí do celku. V projekte *Bubbles* je manažér vývoja a integrácie priamo zodpovedný za správu a údržbu repozitárov na *Bitbucket-e*. Taktiež dohliada na poriadok vo vetvách projektu a v udržiavaní [metodiky verziovania](#) projektu. Jeho zodpovednosťou je aj udržiavanie stabilnej produkčnej vetvy, resp. keď nastane problém v produkčnej verzií, promptne ho rieši a snaží sa čo najskôr dostať produkčnú vetvu do stabilnej formy.

Viac informácií k údržbe poriadku vo vetvách nájdete v podkapitole [Metodika: Verziovanie](#).

### 3.7. Hlavný architekt

Hlavný architekt je rola, ktorá zodpovedá za dodržiavanie architektúry v rámci programovania. Taktiež sa stará o stabilitu produktu udržiavaním testov. Hlavný architekt taktiež navrhuje architektúru, podľa ktorej sa riadi projekt. V projekte *Bubbles* je hlavným architektom Andrej Železnák. Počas predošlých šprintoch navrhol vhodný mvc model a taktiež vhodné prepojenie metamodelu s mvc modelom. Viac podrobností o architektúre sa nachádza v dokumente *Inžinierské dielo*.

Pre udržanie stability produktu hlavný architekt sa stará aj o správne písanie testov a ich údržbu. Tento proces je popísaný v nasledujúcej podkapitole.

### 3.7.1. Manažment testovania

Projekt *Bubble* vyvíja v jazyku C++ pod vývojovým prostredím *Eclipse*. Pre toto vývojové prostredie existuje nástroj na unit testovanie, ktorý sa nazýva CUTE. Pretože náš projekt musí byť otestovaný ako celok je nutné aby bol schopný bežať aj bez vývojového prostredia, preto na niektoré testy využívame Ruby a Windows comand line na písanie scriptov, ktorými testujeme výstupy projektu. Preto by mala nová funkcionality byť navrhnutá tak aby sa dala vyvolať aj z príkazového riadku za pomoci parametrov.

Viac informácií o testovaní je popísané v podkapitole [Metodika: Testovanie](#).

### 3.8. Manažment rizík

- Odchod členov z tímu – toto riziko sa snažíme redukovať tak, že všetky dôležité úlohy zadávame dvojici a nie jednému členovi. Aj napriek tomu, keď na úlohe pracovala jedna osoba, druhá ho kontrolovala a snažila sa mu vypomôcť čo najviac. Vďaka tomu by mala byť zainteresovaná v úlohe tak, že ho mohla hocikedy nahradiť.
- Chyby v prototype, na ktorom sme mali pracovať – keďže sme dostali prototyp, do ktorého sme mali za úlohu implementovať nové moduly, ktoré museli komunikovať s existujúcou funkcionality, bolo náročné overiť dané moduly do hĺbky. Toto riziko sme sa snažili redukovať tým, že sa snažíme o čo najmenšie využitie existujúceho kódu, ktorý nie je programovaný až tak kvalitne.
- Chybná implementácia – niektoré metódy a triedy neboli naprogramované správne a vzniklo nám riziko chybnej implementácie a riziko časovej zložitosti kódu. Riziko sme sa snažili odstrániť preprogramovaním niektorých častí existujúceho kódu, ktorým sme v rýchlosti porozumeli.
- Vznik chyby v zdrojovom kóde – Toto riziko sme čiastočne chránili párovým programovaním, počas ktorého sme robili aj code review. Code review je popísané bližšie v metodike Code review.

## 4. Sumarizácia šprintov

### 4.1. Šprint č. 1

V rámci šprintu č. 1 sme sa zoznámili s pravidlami predmetu tímový projekt a so zadaním projektu. Zistili sme, že náš projekt je už rozrobený a tak nás čaká zoznámenie sa s už vyrobeným protoypom a jeho rozbehanie. Problémom bolo, že pôvodný projekt má 19 branchov a museli sme sa kontaktovať s pôvodnými autormi, aby sme zistili kde vôbec nájdeme časti, ktoré potrebujeme pre svoju prácu. Ďalej sme zistili, že projekt je napísaný v jazyku C++ a je nutné ho vyvíjať v prostredí Eclipse, pretože externé knižnice by sa len s ťažkou námahou dali skompilovať pre Visual Studio. Po niekoľkých neúspešných pokusoch migrovať projekt do Visual Studia sme to vzdali a rozhodli sa pre ďalší vývoj v prostredí Eclipse.

V rámci šprintu sme urobili analýzu Activity Diagramu a Fragmentov, ktoré budú implementované do metamodelu 3D UML Diagramu Aktivít. Ako súčasť šprintu bola vytvorená webová stránka projektu a nastavené nástroje pre manažment projektu, teda Jira. Ďalej sme sfunkčnili náš tímový server a dohodli sa na mechanizmoch riadenia tímu.

#### 4.1.1. Úlohy v šprinte č. 1

Key	Summary	Assignee	Status	Created	Updated
<u>20</u>	Studium povodnej branche	Michal Valovic	Closed	19.10.2014 15:49	5.11.2014 17:55
<u>14</u>	Prihláška na TP CUP	Michal Valovic	Closed	15.10.2014 23:09	22.10.14 18:14
<u>13</u>	Analýza klasického Diagramu Aktivít	Martin Melis	Closed	15.10.2014 23:06	5.11.2014 18:04
<u>12</u>	Analýza klasických fragmentov	Lukas Markovic	Closed	15.10.2014 23:03	25.10.14 20:36
<u>11</u>	Vytvorenie Backlogu	Francisc Juras	Closed	15.10.2014 22:54	19.10.14 16:05
<u>9</u>	[3D-UML] Oživenie Activity Diagramu	Andrej Zeleznak	Closed	15.10.2014 22:51	1.11.2014 0:16
<u>6</u>	Vytvorenie našej branche	Andrej Zeleznak	Closed	15.10.2014 22:33	27.10.14 17:09
<u>5</u>	Rozbiehat prototyp	Andrej Zeleznak	Closed	15.10.2014 13:25	1.11.2014 0:21
<u>4</u>	[AR] jira	Francisc Juras	Closed	15.10.2014 13:24	19.10.14 16:04
<u>3</u>	[AR] git	Martin Melis	Closed	15.10.2014 13:23	19.10.14 16:05
<u>2</u>	Vytvorit webstranku	Miroslav Kudlac	Closed	15.10.2014 13:22	19.10.14 16:05
<u>1</u>	Vytvorit poster	Michal Valovic	Closed	15.10.2014 13:20	19.10.14 16:05

#### 4.2. Šprint č. 2

V rámci 2. šprintu sme vykonali analýzu metamodelu a aktivity diagramu, konečne sa nám podarilo spojzduť nový prototyp, i keď sme mali trochu problémy, pretože sa prototyp nenachádzal na odkazovanom mieste, ale úplne inde. Navyše sme neboli schopní rozbehať projekt bez návodu, ktorý sme nemali. Nakoniec sa nám to ale podarilo a návod je uložený v repozitári. Okrem toho sme mali za cieľ vytvoriť architektúru. Zároveň bolo nutné, aby sme vytvorili analýzu súčasného riešenia, aby sme zistili, čo všetko už v projekte je implementované a v čom sa ešte môžeme realizovať. V tomto šprinte sme poslali v riadnom čase prihlášku na TP Cup. Okrem toho sme sa zžívali s evidenciou úloh v Jire a učili sa tento nástroj používať.

#### 4.2.1. Úlohy v šprinte č. 2

Key	Summary	Assignee	Priority	Status	Created	Updated
<u>30</u>	Vytvorenie dokumentu s analýzou tried a funkcií v existujúcom riešení	Andrej Zeleznak	Major	Closed	29.10.2014 20:08	29.10.2014 21:40
<u>29</u>	Rozbiehanie nástroja na code review	Lukas Markovic	Major	Closed	29.10.2014 19:27	4.11.2014 20:52
<u>28</u>	Analýza testovacích nástrojov c++ nad eclipse	Francisc Juras	Major	Closed	29.10.2014 19:18	5.11.2014 17:54
<u>26</u>	Analýza 3D UML Diagramu Aktivít	Martin Melis	Major	Closed	25.10.2014 20:25	5.11.2014 17:53
<u>25</u>	Implementácia základu fragmentu	Andrej Zeleznak	Major	Closed	22.10.2014 19:36	5.11.2014 17:54
<u>23</u>	Korekcia metamodelu	Miroslav Kudlac	Major	Closed	19.10.2014 15:55	5.11.2014 17:53
<u>22</u>	Fragment	Lukas Markovic	Major	Closed	19.10.2014 15:54	2.11.2014 8:20
<u>20</u>	Studium povodnej branche	Michal Valovic	Major	Closed	19.10.2014 15:49	5.11.2014 17:55
<u>13</u>	Analýza klasického Diagramu Aktivít	Martin Melis	Major	Closed	15.10.2014 23:06	5.11.2014 18:04
<u>12</u>	Analýza klasických fragmentov	Lukas Markovic	Major	Closed	15.10.2014 23:03	25.10.2014 20:36
<u>9</u>	[3D-UML] Oživenie Activity Diagramu	Andrej Zeleznak	Major	Closed	15.10.2014 22:51	1.11.2014 0:16
<u>6</u>	Vytvorenie našej branche	Andrej Zeleznak	Major	Closed	15.10.2014 22:33	27.10.2014 17:09
<u>5</u>	Rozbiehať prototyp	Andrej Zeleznak	Major	Closed	15.10.2014 13:25	1.11.2014 0:21

#### 4.3. Šprint č. 3

V rámci šprintu č. 3 sme začali realizovať implementáciu metamodelu. Dôležité bolo vytvoriť dobrú architektúru a potom citlivo vložiť naše riešenie do prototypu.

Zároveň sa nám podarilo implementovať bloky merge, join a fork. Počas šprintu sme mali prezentáciu v rámci prezentačného panelu, táto prezentácia sa nám a aj publiku veľmi páčila a s tým sme boli veľmi spokojní. Okrem toho sme sa pre nasledujúci priebeh rozdelili na dva tímy, jeden ktorý bude



implementovať metamodel a jeden ktorý bude implementovať GUI projektu. Celkovo sme splnili takmer všetky naše ciele a s priebehom tohto šprintu sme spokojní.

### 4.3.1. Úlohy pre šprint 3

Key	Summary	Assignee	Status	Created	Updated
<u>46</u>	Vytvorenie dokumentácie riadenia	Hana Baranovicova	Closed	11/12/2014 17:07	11/19/2014 16:05
<u>45</u>	Vytvorenie inzinierskeho diela	Michal Valovic	Closed	11/12/2014 17:06	11/19/2014 16:05
<u>43</u>	Integracia nastroja na metriky kodu	Francisc Juras	Closed	11/5/2014 19:26	11/17/2014 22:15
<u>42</u>	Integracia testovacieho nastroja	Francisc Juras	Closed	11/5/2014 19:25	11/18/2014 23:46
<u>41</u>	Integracia sonaru do projektu	Francisc Juras	Closed	11/5/2014 19:25	11/17/2014 17:49
<u>40</u>	Overenie nasho riesenia 3D UML	Michal Valovic	Resolved	11/5/2014 19:16	11/16/2014 15:02
<u>39</u>	Odvodit merge	Lukas Markovic	Closed	11/5/2014 19:12	11/10/2014 21:46
<u>38</u>	Odvodit decision	Lukas Markovic	Closed	11/5/2014 19:12	11/9/2014 20:29
<u>37</u>	Odvodit join	Lukas Markovic	Closed	11/5/2014 19:11	11/10/2014 21:45
<u>36</u>	Odvodit fork	Lukas Markovic	Closed	11/5/2014 19:11	11/10/2014 21:45
<u>35</u>	Implementacia metamodelu	Martin Melis	Closed	11/5/2014 19:10	11/19/2014 16:09
<u>34</u>	Softverova analyza aktualneho stavu	Hana Baranovicova	Open	11/5/2014 19:08	11/19/2014 18:07
<u>33</u>	Finalizacia analyzy metamodelu	Martin Melis	Closed	11/5/2014 19:06	11/10/2014 21:46
<u>31</u>	Vytvorenie prezentacie	Martin Melis	Resolved	11/5/2014 19:04	11/10/2014 21:45

### 4.4. Šprint č. 4

V tomto šprinte sme mali ako hlavnú úlohu naplniť metamodel funkcionalitou, vylepšiť GUI pre naše zadanie, vytvoriť základnú podobu funkcie insert a delete diagramu. Tieto úlohy sa nám podarilo aj realizovať, s čím sme spokojní. Naša práca bola konečne viditeľná aj v zdrojovom kóde tohto tímového projektu. Avšak nie všetko ide podľa plánu, keďže napojenie na grafiku nášho projektu stále nie je na

dobrej úrovni. Najťažšou úlohou bolo vytvorenie MVC architektúry pre naše riešenie. MVC sme navrhli na stretnutí a neskôr ho Andrej Železnák aj implementoval.

V rámci tohto šprintu sme absolvovali aj SCRUM panel, na ktorom sme ako aj ostatné tímy mali lepšie pochopili túto metódu, zistili sme, že v tíme sme doteraz pracovali dobre, ale vďaka tomuto stretnutiu dokážeme lepšie identifikovať problémy a tak aj lepšie riadiť našu prácu.

#### 4.4.1. Úlohy v šprinte č. 4

Key	Summary	Assignee	Status	Created	Updated
<u>56</u>	Aktualizacia používateľskej príručky	<i>Unassigned</i>	Open	19.11.2014 22:54	3.12.2014 20:48
<u>55</u>	Aktualizacia technickej dokumentácie	<i>Unassigned</i>	Open	19.11.2014 22:53	3.12.2014 20:48
<u>54</u>	implementácia MVC	Andrej Zeleznak	Closed	19.11.2014 21:31	24.11.2014 18:14
<u>53</u>	GRAFIKA - štúdium knižnice OGRE vzhľadom na naše riešenie	Lukas Markovic	Closed	19.11.2014 21:29	3.12.2014 20:21
<u>52</u>	dokončenie metamodelu	Martin Melis	Closed	19.11.2014 21:26	26.11.2014 14:12
<u>51</u>	ošetrenie prvkov podľa metamodela	Miroslav Kudlac	Closed	19.11.2014 21:24	26.11.2014 13:39
<u>50</u>	Vytvorenie DELETE	Miroslav Kudlac	Resolved	19.11.2014 21:23	3.12.2014 16:10
<u>49</u>	Vytvorenie INSERT	Francisc Juras	Resolved	19.11.2014 21:21	24.11.2014 19:49
<u>34</u>	Softverova analyza aktualneho stavu	Hana Baranovicova	Closed	5.11.2014 19:08	3.12.2014 18:11
<u>21</u>	Navrh vhodnej struktury MVC	Andrej Zeleznak	Closed	19.10.14 15:52	24.11.2014 15:20

#### 4.5 Šprint č.5

V piatom šprinte sme potrebovali dokončiť a zdokonaľiť náš výstup v zimnom semestri. Identifikovali sme problémové časti, ktoré nie sú doladené. Dohodli sme sa, ktoré časti zdokonalíme ešte v tomto semestri a určili sme úlohy. V šprinte bolo nutné vytvoriť aspoň počiatočný algoritmus pre odchytyvanie pozície myši. Zlepšenie pridávania a odoberania objektov, vylepšenie grafických objektov.

#### 4.5.1 Úlohy v šprinte č. 5

Key	Summary	Assignee	Status	Created	Updated
<u>66</u>	Dokoncenie dokumentacie inzinierske dielo	Lukas Markovic	Closed	3.12.2014 20:19	11.12.2014 0:03
<u>65</u>	Odchytavanie pozicii mysi pre elementy	Michal Valovic	Closed	3.12.2014 20:17	8.12.2014 20:14
<u>64</u>	Prepojenie metamodelu s grafikou	Andrej Zeleznak	Closed	3.12.2014 20:17	10.12.2014 19:36
<u>63</u>	Vykreslovanie layera	Hana Baranovicova	Closed	3.12.2014 20:15	10.12.2014 17:32
<u>62</u>	Structured activity node	Lukas Markovic	Closed	3.12.2014 20:09	11.12.2014 0:02
<u>61</u>	Dokoncenie dokumentacie riadenia	Francisc Juras	Closed	3.12.2014 20:05	10.12.2014 17:33
<u>60</u>	Vytvorenie prezentacie riadenia	Martin Melis	Closed	3.12.2014 20:04	11.12.2014 0:00
<u>59</u>	Rozsiren timer grafickeho pouzivatel'skeho rozhrania	Miroslav Kudlac	Closed	3.12.2014 18:19	11.12.2014 0:02
<u>58</u>	Aktualizacia funkcie DELETE	Francisc Juras	Closed	3.12.2014 18:16	10.12.2014 23:45
<u>57</u>	Aktualizacia funkcie INSERT	Andrej Zeleznak	Closed	3.12.2014 18:15	10.12.2014 17:34
<u>56</u>	Aktualizacia pouzivatel'skej prirucky	Michal Valovic	Closed	19.11.2014 22:54	11.12.2014 0:03
<u>55</u>	Aktualizacia technickej dokumentacie	Lukas Markovic	Closed	19.11.2014 22:53	10.12.2014 23:53

## 5. Používané metodiky

### 5.1. Metodika: Coding standard

V dokumente [Coding standard](#) je predstavovaná metodika pre kódovací štandard použitý v tímovom projekte. Jeho úlohou je zavedenie pravidiel odsadzovania, názvov premenných a komentárov pre dosiahnutie vysokej úrovne čitateľnosti a prehľadnosti, a tým aj zlepšenie možností udržateľnosti a modifikovateľnosti. Tieto pravidlá budú záväzné pre všetkých pre všetkých vývojárov zúčastnených na tvorbe či úpravách kódu.

### 5.2. Metodika: Code review

Dokument [Code review](#) opisuje využitie nástrojov určených na automatizované prehliadanie, ich konfiguráciu, ale tiež ich nasadenie a následné spracovanie či úpravu ich výsledkov. Samotný proces je zameraný na vyvarovanie sa chýb a problémov v kóde, ktoré môžu byť zanášané zúčastnenými vývojármi, a to najmä odchýlok od pravidiel písania zdrojových kódov. Opísané postupy sa vzťahujú k nástroju na statickú analýzu zdrojového kódu s názvom CODAN.

### 5.3. Metodika: Dokumentovanie

Cieľom dokumentu [Dokumentovanie](#) je definovať metodiku na vytvorenie dokumentácie k tímovému projektu popisom pravidiel a postupov, ktoré majú formulovať výslednú podobu dokumentácie, ako napríklad kultúru formátovania či poznámky k procesu tvorby dokumentácie. Dokument tiež obsahuje rozdelenie rolí a zodpovedností členov tímu.

### 5.4. Metodika: Evidencia úloh

Metodika opísaná v dokumente [Evidencia úloh](#) sa venuje zavádzaniu jednotného postupu pre evidenciu úloh a ich následného pridelovania konkrétnym osobám. Postup pre evidenciu je potrebný pre granularitu úloh projektu potreba tohto postupu je nevyhnutná pri plánovaní. Evidencia je vykonávaná využitím webovej aplikácie Jira, ktorá okrem evidencie a plánovania umožňuje aj sledovanie priebehu projektu.

## 5.5. Metodika: Komunikácia

Metodika pre internú a externú [komunikáciu](#) predstavuje vopred definované postupy v rámci komunikácie v tíme a použité komunikačné kanály. V rámci týchto postupov je možné oznamovanie vyskytnutých problémov, rozoberanie týchto problémov, analýza jednotlivých úloh či ohlásenie nečakane naskytnutých situácií. Podobnejšie rozoberá priebeh komunikácie v jej rôznych formách.

## 5.6. Metodika: Testovanie

Úlohou metodiky [Testovanie](#) je zdefinovať základné metódy testovania tímového projektu. Ten je vytváraný v jazyku C++ v prostredí Eclipse. Na testovanie pomocou unit testov je používaný nástroj CUTE, ktorý beží v použítom vývojovom prostredí. Pre ďalšie testovanie budú použité skripty písané v Ruby, ktoré budú overovať výstupy projektu. Vďaka nim bude možné spúšťať testy len za použitia príkazového riadka. Tiež je spomenutý spôsob spracovania získaných výsledkov testov.

## 5.7. Metodika: Verziovanie

Dokument [Verziovanie](#) približuje metódy použité pri verziovaní projektu. Rozpisuje inštaláciu nástroja Git a jeho použitie pri manažmente verziovania. Objasňuje tiež obsluhu verziovaných dát uložených v repozitári BitBucket, vytváranie nových vetiev a riešenie prípadne vzniknutých konfliktov medzi nimi.

## 6. Globálna retrospektíva ZS

Globálna retrospektíva, ktorá sa nachádza na nasledujúcich stranách je z dôvodu prehľadnosti zoradená a zosumarizovaná podľa jednotlivých šprintov.

### 6.1. Retrospektíva prvého šprintu

#### 6.1.1. Priebeh šprintu

V rámci 1. šprintu boli urobené úvodné kroky v projekte. Keďže išlo o pokračujúci projekt, prioritnou úlohou šprintu bolo prevziať projekt a zoznámiť sa s doteraz implementovaným obsahom. Tu nastal problém, keďže projekt obsahoval 19 aktívnych branchov a bolo potrebné zistiť, ktorý je aktuálny.

V rámci šprintu bola taktiež urobená prvotná analýza Activity Diagramu a Fragmentov, ktoré budú implementované do metamodelu 3D UML Diagramu Aktivít. Ako súčasť šprintu bola vytvorená webová stránka projektu a nastavené nástroje, pre manažment projektu.

#### 6.1.2. Úlohy šprintu

##### **Štúdium UML definície**

– prvou *spoločnou* úlohou, ktorá bola úspešne zvládnutá, bolo naštudovanie špecifikácia jazyka UML.

##### **Rozbehanie existujúceho prototypu**

– *Andrej Železnák* spolu s *Lukášom Markovičom* spojzdnili poslednú verziu projektu, ktorá bola podľa dostupných informácií umiestnená na branchi *TP\_metamodel*

##### **Nástroj na manažment projektu**

– *Francisc Juras* pripravil a nakonfiguroval Jiru pre projekt 3D UML.

##### **Analýza metamodelu Activity Diagramu a Fragmentov**

– Hlavnou úlohou pokrývajúcou oblasť analýzy bola v prvom šprinte analýza metamodelu Activity Diagramu. O túto analýzu sa postaral *Martin Melis*. Keďže súčasťou 3D UML diagramu aktivít budú aj fragmenty, *Lukáš Markovič* urobil úvodnú analýzu existujúcich fragmentov zo sekvenčného diagramu a ich metamodelu.

##### **Oživenie Activity Diagramu**

– *Andrej Železnák* spolu s *Hanou Baranovičovou* začali pracovať na spojzdení Activity Diagramu v existujúcom prototypu.

##### **Vytvorenie web stránky projektu**

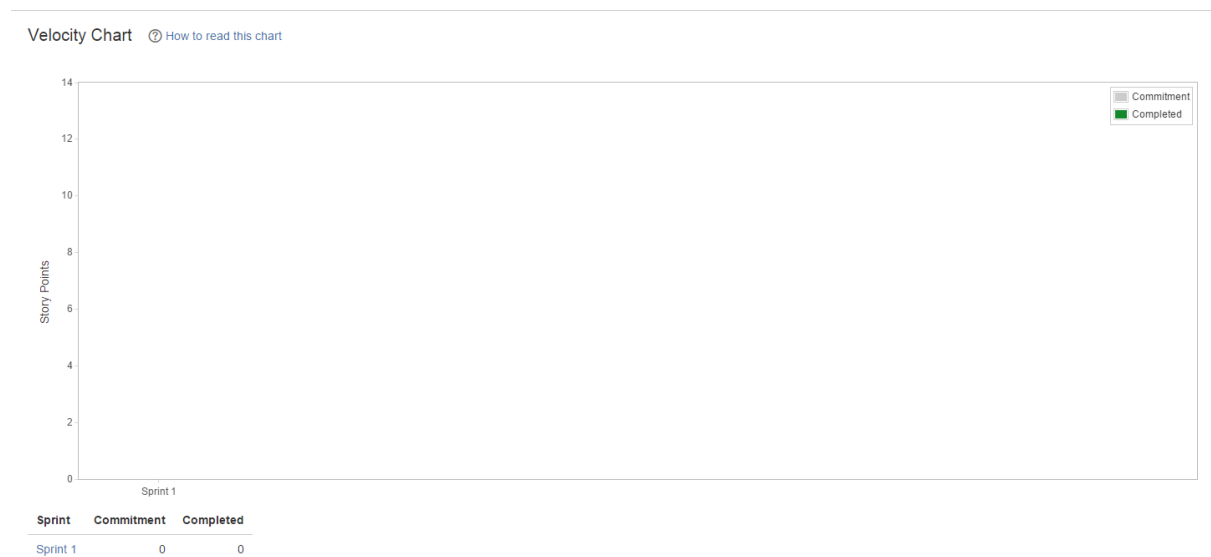
– **Michal Valovič a Miroslav Kudláč** pripravili projektovú stránku, ktorá bola nasadená na požadovaný server.

### Prihláška na TP – Cup

– V rámci prvého šprintu **Miroslav Kudláč** spolu s **Michalom Valovičom** napísali prihlášku do súťaže TP – Cup, ktorú navyše ešte skontroloval **Lukáš Markovič**.

V rámci tohto kola fungovalo dobre rozdeľovanie úloh a ich plnenie. Väčšina úloh sa prenesie aj do ďalších šprintov. Miro a Michal predstavili návrh stránky tímu a spoločne ho realizovali v krátkom čase. Nahodenie na server prebehlo bez problémov. Andrej sa postaral o sprístupnenie servera cez FTP. Stránku priebežne aktualizujeme. V rámci tohto šprintu sme začali realizovať aj analýzu dokumentov a rozbehanie aktuálneho prototypu. Boli sme veľmi spokojní s počiatočnými dokumentmi analýzy, v ktorých sa bude pokračovať aj v ďalšom šprinte. Branch sme rozbehli pomerne rýchlo na všetkých počítačoch, avšak začali sme rozbiehať Activity diagram a práve tam sme narazili na problémy, keďže sme v aktuálnom prototypu doposiaľ nenašli Activity diagram. Tento problém sme museli preniesť do ďalšieho šprintu.

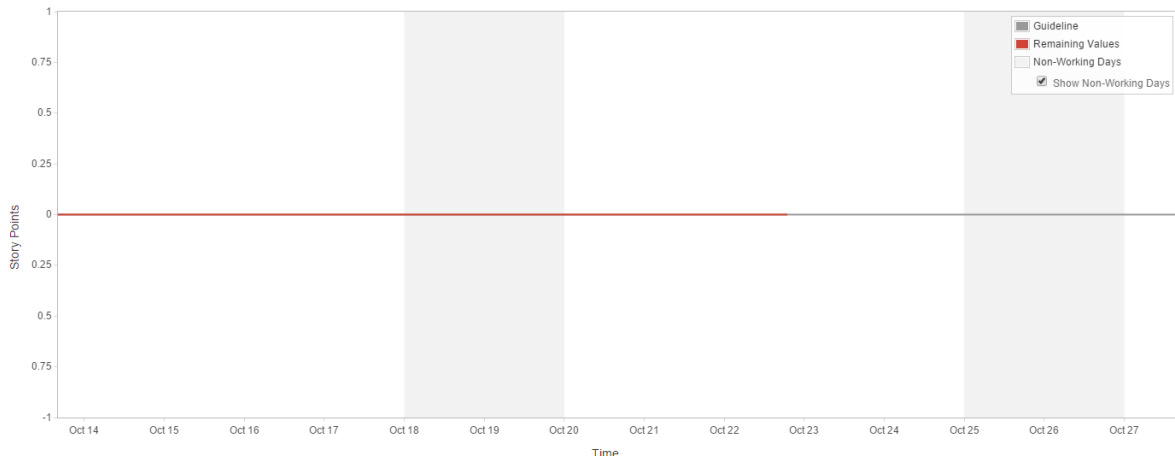
### 6.1.3. Velocity chart



Velocity chart prvého šprintu ukazuje, že sme nemali naplánovaný žiaden story point. V tomto šprinte sme sa zoznamovali s Jirou a z tohto dôvodu sme nevedeli pridať story pointy do taskov.

## 6.1.4. Burndown chart

Burndown Chart Sprint 1 [How to read this chart](#)



Rovnak ako pri Velocity charte, aj pri Burndown charte je vidieť, že sme mali naplánované 0 story pointov.

## 6.2. Retrospektíva druhého šprintu

### 6.2.1. Priebeh šprintu

V rámci 2. šprintu boli v rámci TP vykonané nasledujúce kroky.

- Analýza metamodelu a aktivity diagramu
- Rozbehание nového prototypu
- Vytvorenie architektúry
- Analýza súčasného riešenia

Tieto kroky sa dajú rozdeliť do dvoch veľkých úloh a tými sú Analýza metamodelu diagramu aktivít a spojzadenie a opravenie aktuálneho riešenia.



## 6.2.2. Úlohy šprintu

### Rozbehanie existujúceho riešenia

- *Andrej Železnák* spolu s *Michalom Valovičom* a *Hanou Baranovičovou* opravili súčasné riešenie a spojzdnili ho, teraz na ňom môžeme všetci pracovať na svojich počítačoch

### Vytvorenie našej branche

- *Andrej Železnák* vytvoril náš branch, do ktorého budeme vkladať naše riešenie

### Analýza súčasného riešenia

- *Miroslav Kudláč* vytvoril v rámci analýzy dokument v ktorom popisuje prototyp diagramu aktivít, ktorý vytvoril Matej Škoda

### Analýza metamodelu, Activity Diagramu a Fragmentov

- Hlavnou úlohou pokrývajúcou oblasť analýzy bola v prvom šprinte analýza metamodelu Activity Diagramu. O túto analýzu sa postaral *Martin Melis*. Keďže súčasťou 3D UML diagramu aktivít budú aj fragmenty, *Lukáš Markovič* urobil úvodnú analýzu existujúcich fragmentov zo sekvenčného diagramu a ich metamodelu.

### Oživenie Activity Diagramu

- *Andrej Železnák* spojzdnil nový prototyp

### Vytváranie plánu a aktualizácia webu o dokumenty (úlohy, záznamy zo stretnutia)

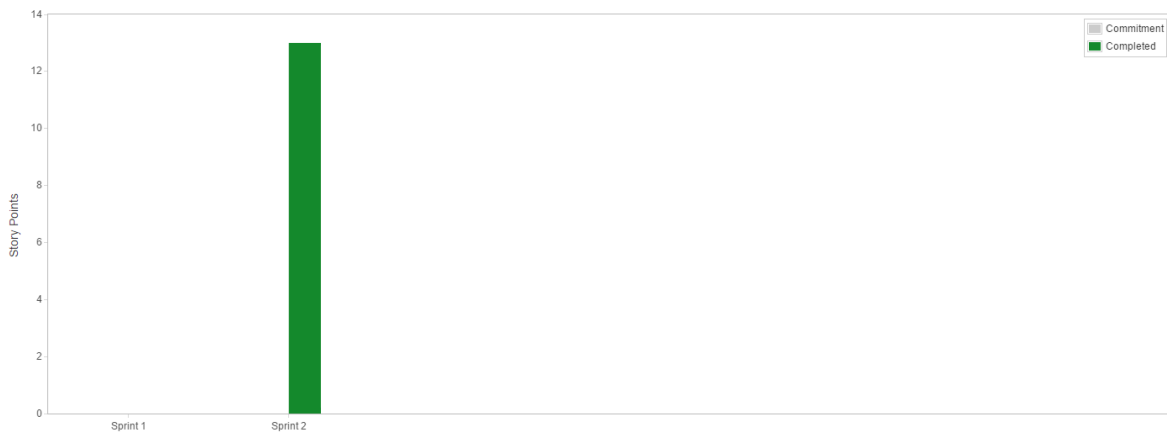
- *Miroslav Kudláč* a *Lukáš Markovič* vytvorili plán a zadelili ciele a úlohy pre celý tím, ktoré následne *Francisc Juras* nahodil do Jiry.

Počas druhého šprintu sme pociťovali väčší nátlak a nepracovalo sa nám dobre kvôli implementácií. Zistili sme, že v branchi TP\_Metamodel sa nenachádza nič z toho, čo sme očakávali. Bol nám dodaný zlý branch. Začali sme rozbehať prvotné riešenie Mateja Škodu do našej branche. Nedostali sme návod ako spojzdnit' tento projekt. Cesty v projekte boli naozaj zle nastavené. Taktiež bolo nutné nainštalovať si ďalšie knižnice pre tento projekt. Celý proces bol pre nás komplikovaný a časovo náročný. Do konca šprintu sme ale tieto problémy, ktoré nevznikli našou vinou vyriešili a teraz môžeme pokojne pracovať s týmto riešením. Vytvorili sme dokument, v ktorom je popísaný postup spojzdenia projektu, ktorý sa nachádza v našej branchi. Tento problém bol našťastie jediný, ktorý sme museli riešiť. V rámci týchto dvoch týždňov sa nám podarilo finalizovať analýzu a navrhli sme architektúru metamodelu pre diagram aktivít. Výsledkom sú skvelé dokumenty, v ktorých sa nachádzajú analýzy fragmentov, diagramu aktivít, aktuálneho riešenia a nášho riešenia. Zoznámili sme sa so zdrojovými kódmi diagramu aktivít pomerne rýchlo potom, čo sa nám podarilo spojzdnit' predchádzajúce riešenie.

Celkovo našu prácu hodnotíme dobre, keďže sme sa vyrovnali s neočakávanými problémami a rozdelili sme správne úlohy, každému podľa jeho schopností. Nálada v tíme sa rapídne zlepšila po jeho vyriešení.

### 6.2.3. Velocity chart

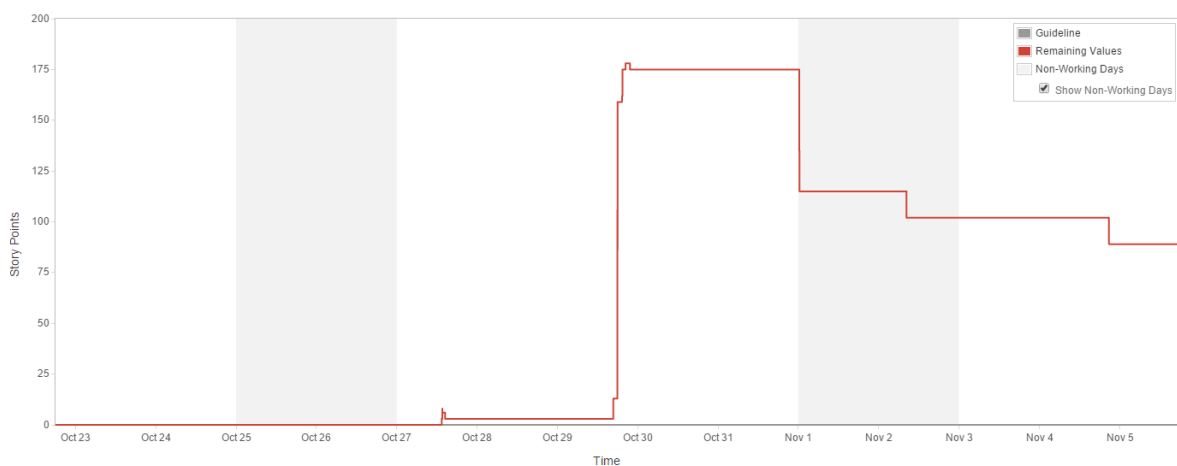
Velocity Chart [How to read this chart](#)



Velocity chart v tomto šprinte ukazuje, že sme spravili všetko čo sme si naplánovali. Taktiež v tomto šprinte sme sa ešte zoznamovali s prostredím, takže plánovanie ešte nie je na úrovni, ktorú by sme si želali.

### 6.2.4. Burndown chart

Burndown Chart Sprint 2 [How to read this chart](#)



Burndown chart, podobne ako velocity chart ukazuje, že sme spravili všetko, čo sme si na tento šprint naplánovali. Na začiatku šprintu sme nemali ohodnotené úlohy. V polovici šprintu sme ich nahodnotili a postupne uzatvárali. V budúcich šprintoch sa budeme snažiť lepšie plánovať, aby nebolo potrebné



pridávať úlohy počas šprintu. Počas tohto šprintu sme mali naplánovaných 178 story pointov, ktoré sa nám podarilo aj uskutočniť.

## 6.3. Retrospektíva tretieho šprintu

### 6.3.1. Priebeh šprintu

V rámci 3. šprintu boli v rámci TP vykonané nasledujúce kroky.

- Implementácia metamodelu
- Implementácia blokov fork, join, merge
- Dokumentácia riadenia a inžinierskeho diela
- Prezentácia v rámci panelu
- Softvérová analýza prototypu
- Vytvorenie a aplikácia metodík

### 6.3.2. Úlohy šprintu

#### Vytvorenie metamodelu

- **Martin Melis a Andrej Železnák** vložili do nášho kódu metamodel pre naše riešenie, Martin finalizoval svoju dokumentáciu a spoločne s Andrejom napojili metamodel na naše riešenie.

#### Implementácia blokov fork, join, merge

- **Lukáš Markovič** realizoval implementáciu týchto blokov

#### Dokumentácia riadenia

- **Hana Baranovičová a Miroslav Kudláč** vytvorili dokumentáciu riadenia, ktorú odovzdáme na konci tohto šprintu

#### Dokumentácia inžinierskeho diela

- **Michal Valovič a Lukáš Markovič** vytvorili dokumentáciu inžinierskeho diela, ktorú odovzdáme na konci tohto šprintu

#### Prezentácia

- V rámci tohto šprintu sme prezentovali. Prezentáciu vytvoril **Martin Melis**, Na prezentovaní zadania sa podieľali **Martin Melis, Francisc Juras a Miroslav Kudláč**.

#### Softvérová analýza projektu

- **Hana Baranovičová** mala za úlohu vytvoriť dokument so softvérovou analýzou prototypu.

#### Vytvorenie a aplikácia metodík

- Všetci v tíme vytvorili konkrétnu metodiku a postupne sa dané metodiky snažíme aplikovať v našom tíme. Francisc Juras spojil nástroje na testovanie a na metriky kódu.

Počas 3. šprintu sme museli splniť veľa formálnych vecí ako metodiky, prezentáciu a vytvorenie dokumentácie, ktorá sa odovzdáva na konci tohto šprintu. Kvôli tomu sme nesplnili všetky ciele, ktoré sme si zaumienili, avšak stále máme hlavy hore, keďže sa nám podarilo citlivo vložiť metamodel do nášho zadania. Martin a Andrej odvedli pri tom skvelú prácu. Andrej, spolu s ostatnými sa bude snažiť tento metamodel napojiť na celú funkcionality Activity diagramu, veríme, že sa nám to podarí.

V rámci prezentačného panelu sme prezentovali naše zadanie, prezentácia dopadla pre nás pomerne dobre a dostali sme pochvalu od vedúceho a aj od publika, týmto sa zdvihla nálada v tíme.

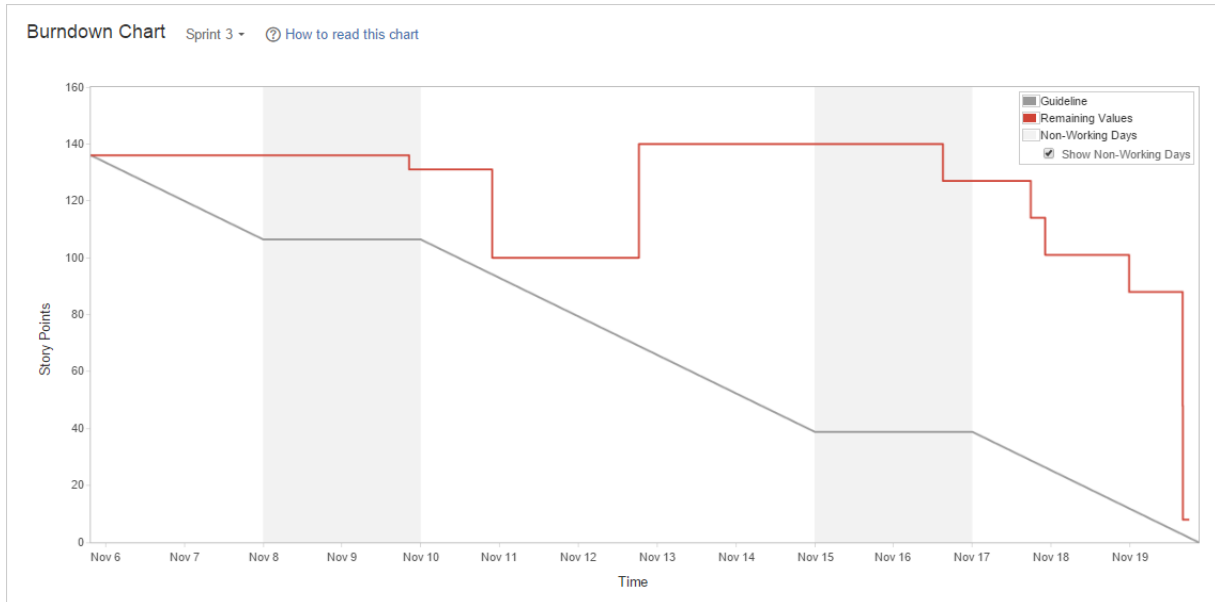
Lukáš implementoval nové bloky do nášho riešenia a ako manažér plánovania rozdelil dočasne implementačné tímy na 2. Jeden tím sa bude zaoberať metamodelom a druhý tím sa pokúsi zlepšiť GUI Activity diagramu. Myslíme si, že je to správny krok.

Fero sa pokúšal rozbehať nástroje na code review, code metrics a na testovanie. Žiaľ podarilo sa mu to splniť z 2/3. Nepodarilo sa mu integrovať nástroj Sonar, rozhodli sme sa, že naďalej s týmto nástrojom nebudeme zabíjať čas.

V rámci tímu sa zlepšila komunikácia a reakcie na zadané úlohy v Jire. Vieme, že celkový progres by mohol byť väčší, ale úlohy, ktoré riešime sú komplexné a nie triviálne.

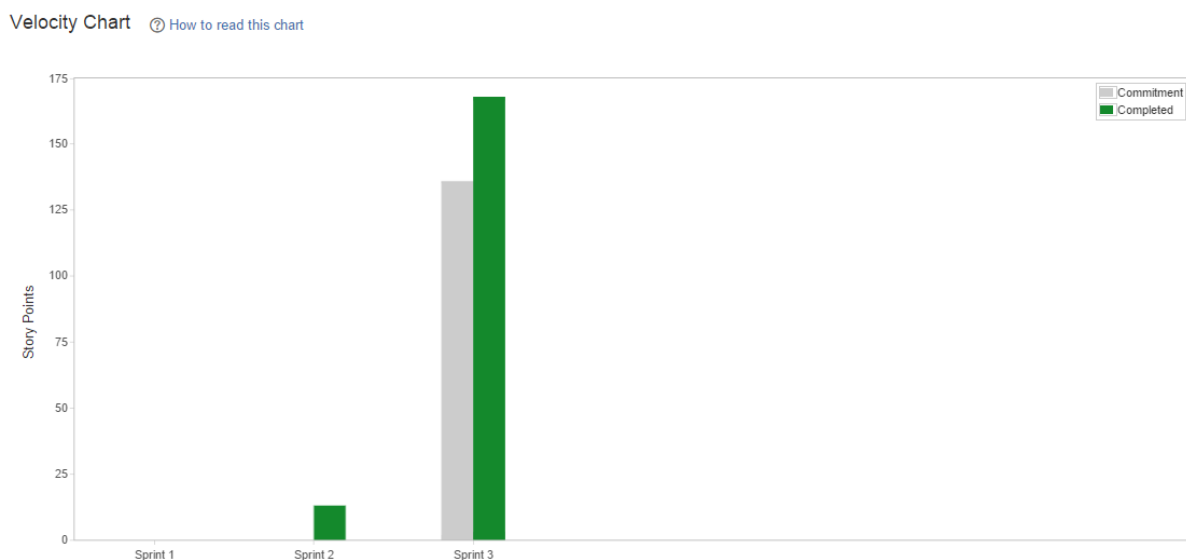
Pri vytváraní dokumentácie sa Miro dohodol s Michalom a rozdelili sme úlohy pre vytváranie a spájania dokumentácie. Tím bol s týmto rozdelením spokojný. A dokumentáciu sme úspešne a podľa požiadaviek skompletizovali.

### 6.3.3. Burndown chart



V treťom šprinte sme mali naplánovaných 176 story pointov, z ktorých sa nám podarilo vypracovať 168. V polovici šprintu sme pridali ešte dve úlohy, ktoré bolo nevyhnutné vypracovať do konca šprintu. Z tohto dôvodu sa nám nepodarilo dokončiť všetky úlohy. Do budúca sa budeme snažiť predvídať takéto udalosti.

### 6.3.4. Velocity chart



Velocity chart ukazuje, že na začiatku šprintu sme naplánovali 136 story pointov. Podarilo sa nám vypracovať 176 story pointov. V budúcnosti si naplánujeme bližšie hodnote 180 story pointov.

## 6.4. Retrospektíva 4. šprintu

### 6.4.1. Priebeh šprintu

V rámci 4. šprintu boli v rámci TP vykonané nasledujúce kroky.

Naplnenie metamodelu funkcionalitou

- Vytvorenie funkcie INSERT
- Vytvorenie funkcie DELETE
- Vytváranie grafických elementov
- Vylepšenie architektúry (MVC)
- Návrh GUI
- Softvérová analýza projektu

## 6.4.2. Úlohy šprintu

### Naplnenie metamodelu funkcionalitou

- **Miroslav Kudláč a Andrej Železnák** vložili do nášho kódu metamodel pre naše riešenie, podľa dokumentácie a pod dozorom **Martina Melisa**

### Implementácia grafických elementov

- **Lukáš Markovič** realizoval implementáciu týchto elementov

### Softvérová analýza

Hana baranovičová vypracovala v tomto šprinte daný dokument

### Počiatky GUI

- Počiatočné GUI pre naše riešenie vypracoval Michal Valovič. Navrhovali sme ho celý tím aby sa nám páčilo a držali sme sa štandardov.

### Insert

- Napísal **Francisc Juras** spolu s pomocou **Andreja Železnáka**

### Delete

- Začiatok funkcie delete vypracoval **Miroslav Kudláč** s pomocou **Andreja Železnáka**

### Vytvorenie MVC

- Dalo by sa povedať že vytvorenie tohto modelu bolo najobťažnejšou úlohou v tomto šprinte. Navrhli sme architektúru MVC a do nášho zadania ju implementoval **Andrej Železnák**.



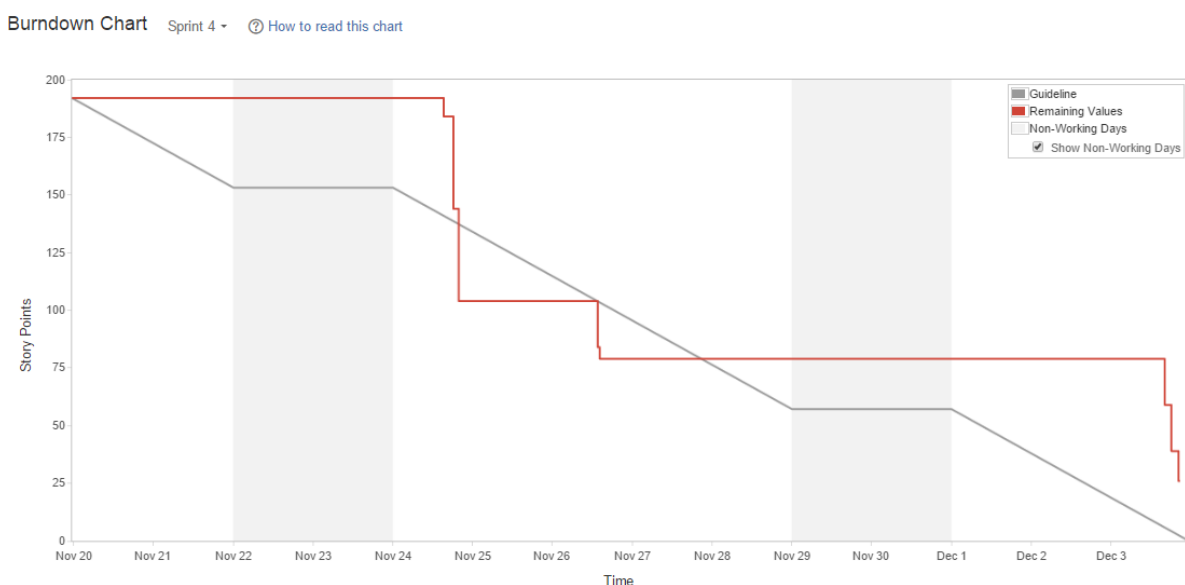
Počas šprintu 4 sme potrebovali preniesť náš prototyp na našu architektúru, keďže predchádzajúca nebola vhodná pre naše riešenie. Andrej, ako náš hlavný programátor realizoval túto pomerne ťažkú úlohu veľmi dobre, čím potešil ako tím, tak aj vedúceho projektu. Od toho sme sa potom všetci mohli odraziť ďalej.

Začali sme implementovať jednotlivé funkcie nášho metamodelu a grafiku pre náš 3D model. Navrhli sme prototypy funkcií INSERT, DELETE a naplnili sme náš metamodel funkcionalitou, ktorá je potrebná pre vykresľovanie elementov a je v špecifikácii UML. V tomto produktívnom tempe musíme pokračovať aj v poslednom šprinte.

Začali sme vytvárať novú grafiku a algoritmy pre vykresľovanie našich elementov. Je to na dobrej ceste. Žiaľ sme sa opäť zdržali kvôli neprehľadnosti kódu vo vytvorenom prototypu, ktorý nám bol dodaný, čomu veľmi nepomáha slabá znalosť jazyka C++ väčšiny tímu. Práve preto sme potrebovali zaťažiť našich znalcov na tento jazyk, žiaľ niektorí pracovali viac a niektorí menej. Tomuto javu sa v budúcnosti musíme vyhýbať.

V rámci tohto šprintu sme absolvovali aj scrum panel. Tento panel sa nám všetkým páčil a bolo to spojenie príjemného s užitočným. Zistili sme, že sme náš projekt riadili pomerne dobre a vďaka tomuto panelu dokonca zvýšime kvalitu jeho riadenia.

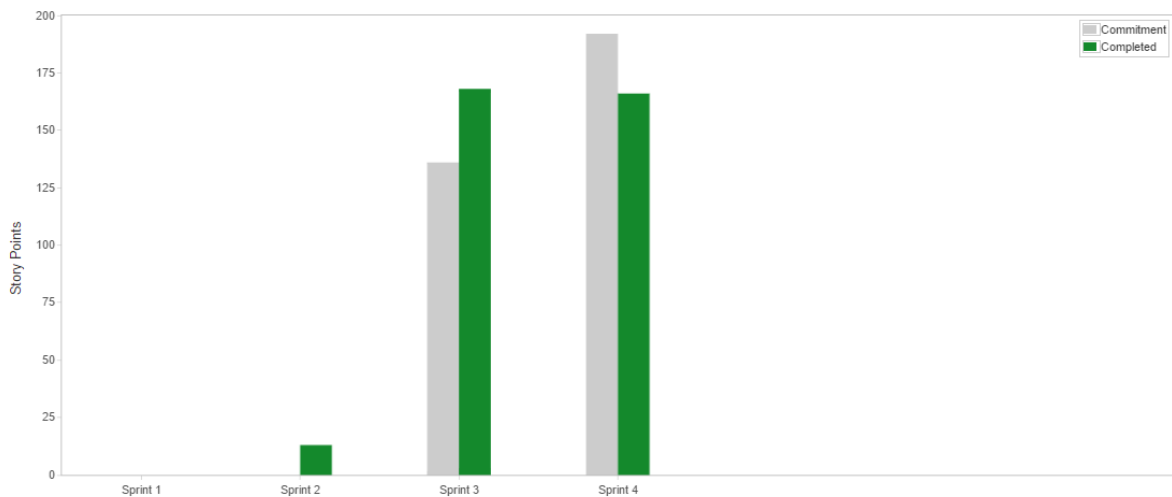
### 6.4.3. Burndown chart



V šprinte 4 sme pracovali na úlohách priebežne. Tieto úlohy sme vykonávali postupne a jeden čas sa nám podarilo udržať pod Guideline-ou. Nepodarilo sa nám uzavrieť dve úlohy, keďže naše kapacity boli vyplnené.

## 6.4.4. Velocity chart

Velocity Chart [How to read this chart](#)



V šprinte 4 sme mali naplánovaných 192 story pointov. Podarilo sa nám splniť 166 story pointov. Ako je možné vidieť vo velocity charte, je to priemerný počet story pointov, ktorý sme uskutočnili aj predošlý šprint. Do budúca sa budeme snažiť zmenšiť počet naplánovaných story pointov, kým nebudeme znovu presahovať naplánovanú hodnotu.

## 6.5. Retrospektíva 5. šprintu

### 6.5.1. Priebeh šprintu

V rámci 5. šprintu boli v rámci TP vykonané nasledujúce kroky.

Naplnenie metamodelu funkcionalitou

- Dokončenie Delete
- Dokončenie Insert
- Dokončenie Select
- Dokončenie Menu
- Dokončenie Grafiky
- Dokončenie Dokumentov

## 6.5.2. Úlohy šprintu

### Dokončenie Delete

- Keďže sa predovšetkým Andrej staral o implementáciu, tak pri týchto úlohách bol on inštruktorom a pomáhal v tomto prípade Franciscovi Jurasovi.

### Dokončenie Insert

- Túto úlohu začal a aj dokončil Andrej Železnák, bola najťažšou a najdôležitejšou v tomto šprinte

### Dokončenie Select

- Andrej Začal funkciu Select a dokončiť sme ju dali na starosť Hane Baranovičovej

### Dokončenie Menu

- Kvôli tomu, že chceme v tomto semestri pridávať prvky, sme sa rozhodli, že vytvoríme jednoduché menu. O jeho realizáciu sa postaral Miroslav Kudláč

### Dokončenie Grafiky

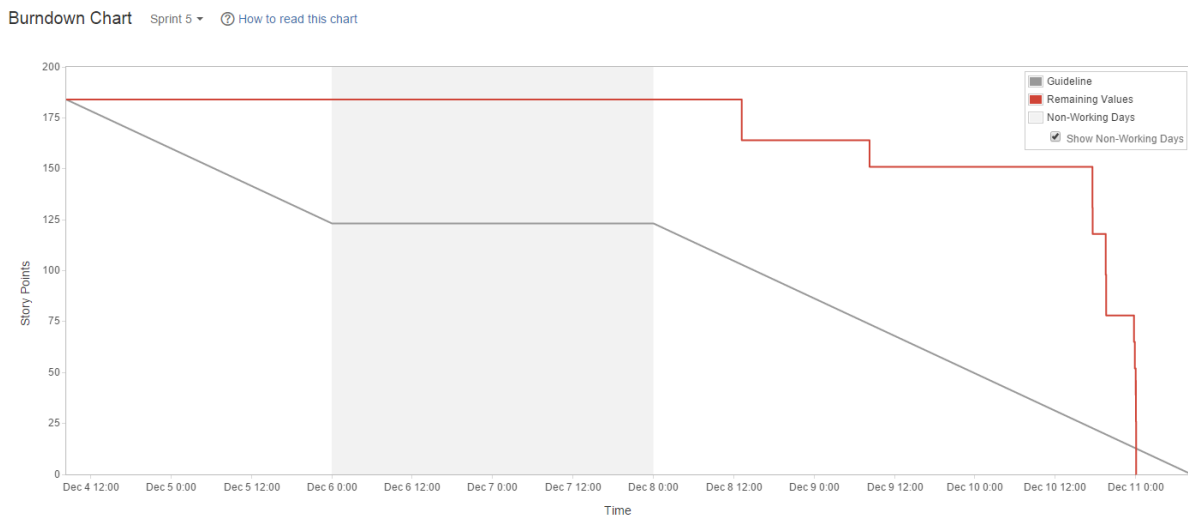
- V grafike bolo potrebné riešiť veľa problémov, tieto problémy riešili spoločne Lukáš Markovič a Michal Valovič. Michal mal na starosti výpočet polohy bodov a zachytávanie klikov na obrazovku. Lukáš mal na starosti spájanie jednotlivých objektov

### Dokončenie Dokumentov

- dokumenty sa predovšetkým staral Miroslav Kudláč, Michal Valovič a Lukáš Markovič

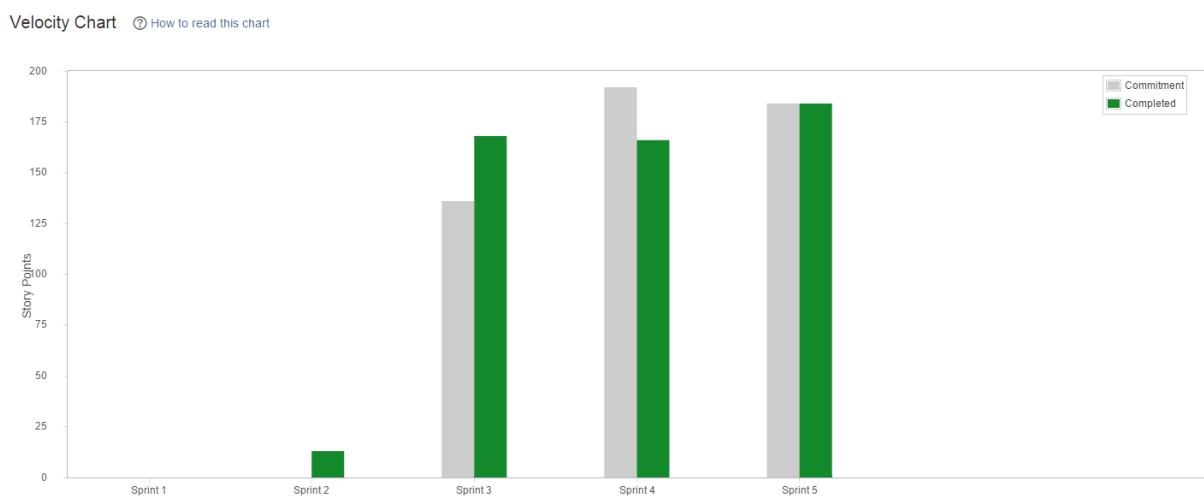
V šprinte sa nám podarilo dokončiť náš projekt tak, aby bol vhodný na prezentáciu pred product ownerom. Nefunguje v ňom všetko, no tieto veci sú zahrnuté do zimného semestra. Andrej odviezol s implementáciou nesmierny kus práce. Na túto prácu sme nadviazali aj ostatní, najmä Michal a Lukáš. Každý ale priložil či väčšou alebo menšou časťou ruku k dielu a v tom musíme pokračovať aj v letnom semestri. Sme v polovici projektu a posledné dva šprinty boli z nášho pohľadu veľmi produktívne vďaka rozdeleniu úloh a tomu, že Andrej vytvoril počiatočnú implementáciu a MVC model vďaka čomu je kód rozdelený a písaný čitateľne.

### 6.5.3. Burndown chart



V šprinte 5. sme pracovali na úlohách priebežne. Počas prvej polovice šprintu sa nám nepodarilo uzavrieť žiadnú úlohu, ale v druhej polovici sme pozatvárali všetky úlohy. Tá prvá polovica súvisí s tým, že úlohy na vývoj boli náročné a nebolo možné nad nimi pracovať paralelne. Vzhľadom na to, že šprint trval polovičný čas, ako zvyčajne, stihli sme vykonať pomerne veľké množstvo úloh.

### 6.5.4. Velocity chart



V šprinte 5. sme mali naplánovaných 184 story pointov. Podarilo sa nám ich splniť všetky vďaka tomu, že sme mali viac času pracovať na projekte. To znamená, že tento šprint sme venovali v kratšej dobe viac času projektu.

## 6.6. Zhrnutie retrospektív

V rámci prvého šprintu sme sa zoznamovali s projektom, jednotlivými nástrojmi pre riadenie a vývoj, ktoré budeme potrebovať pri práci na tímovom projekte. Vytvorené úlohy sa nám úspešne podarilo splniť. S Jirou sme sa len učili narábať, preto sa burndown chart ani velocity chart nedajú brať do úvahy. Druhý šprint sa práca s Jirou začala zlepšovať, ale stále boli isté nedostatky v prístupe jednotlivých členov. Vytvorené úlohy, týkajúce sa najmä analýz diela a spojzdnenia prototypu sme rovnako väčšinou úspešne splnili. V tomto šprinte sme aj prezentovali o našom projekte. Prezentácia mala úspech. V treťom šprinte už práca s Jirou bola na dobrej úrovni, okrem toho, že sme v strede šprintu museli pridávať urgentné tasky na vytvorenie dokumentácie. Tomuto sa chceme v budúcnosti vyhnúť. Okrem dokumentácie sa tasky šprintu zameriavali na implementáciu a návrh metamodelu.

# A. Príloha: Preberací protokol

Tímový projekt 2014/2015

Tím č. 4 – Bubbles

**Predmet odovzdávania:**

Dokumentácia riadenia

Projektová dokumentácia

Vedúci tímového projektu: Ing. Ivan Polášek, PhD.

Podpisom potvrdzuje prevzatie vyššie uvedených častí dokumentácie

V Bratislave

.....

Dátum

.....

Podpis

# B. Príloha: Kompetencie tímu

## B.1. Tím

Baranovičová Hana

- *Znalosti* - C/C++, Python3, SQL, HTML, CSS, GIT, Bash, Qt, LaTeX
- *Bakalárska práca* - 3D hra v OpenGL s využitím analýzy hudby
- *Pracovné skúsenosti* - aplikačný IT support v AVG Technologies CZ

Juras Francisc

- *Znalosti* – JavaScript, C, XML, SQL, HTML, Java, iOS, UIAutomation, správa úloh, dokumentovanie a rôzne nástroje na technické dokumentácie
- *Bakalárska práca* – Značkovanie zdrojových kódov
- *Pracovné skúsenosti* – Tester

Kudláč Miroslav

- *Znalosti* – PHP, JavaScript, C, XML, SQL, HTML, CSS, XSL, Java
- *Bakalárska práca* – Sémantická wiki (aplikácia, ktorá na základe označeného textu v prehliadači vráti zoznam ľudí, ktorí sú autoritami v danej oblasti).
- *Pracovné skúsenosti* – Web developer/projektový manažér

Markovič Lukáš

- *Znalosti* – Java SE/EE, Python, Groovy, C, XML a pridružené jazyky, HTML, MongoDB, MySQL, Oracle Database, Apache Tomcat, Spring framework, IBM WebSphere, UML, BPMN, Unix, Linux, Windows, Eclipse (Spring Tool Suite), Visual Studio, RSA, Git, atď.
- *Bakalárska práca* – Podpora refaktoringu pomocou transformácií medzi jazykom Java a XML (publikované na IIT.SRC 2014)
- *Pracovné skúsenosti* – System analysis and design – Gratex International, a.s. (aktuálne)  
– Java developer – Unicorn Systems (01.07.2013 – 31.05.2014)

Melis Martin



- *Znalosti* - C, Java, XML, XSLT, CSS, MySQL, Ogre, UML, ArchiMate
- *Bakalárska práca* - Využitie nástrojov podnikovej architektúry na zefektívnenie procesov v organizácií

Valovič Michal

- *Znalosti* – PHP, HTML, CSS, JavaScript, C, C#, Lua, XML, Git
- *Bakalárska práca* – Správa binárnych softvérových balíkov a ich závislostí
- *Pracovné skúsenosti* –PHP developer

Železnák Andrej

- *Znalosti* – C, C++ (Win/Unix), C++/CLI, C#, .Net, Bash, Linux, MYSQL, MSSQL, GIT, počítačové videnie, OpenCV, Visual Studio, Pascal, Linux administrátor, Ruby, XML
- *Bakalárska práca* – Optické rozpoznávanie notových záznamov
- *Pracovné skúsenosti* – software developer – ME-Inspection
  - software developer, administrátor, software architekt - Anima Technika 3



## B.2. Téma „3D UML“

UML, jazyk asi najčastejšie používaný na modelovanie pri analýze, návrhu a vývoji informačných systémov, považujeme aj v našom tíme za veľmi perspektívnu oblasť hodnú primeranej pozornosti.

Rozšírenie UML o trojdimenzionálny pohľad náš tím zaujalo hlavne preto, že ide o tému skôr vedecky zameranú, ktorá by však v prípade úspešného dokončenia mohla mať obrovské využitie. Či už ide o uľahčenie prehľadnosti zložitých diagramov alebo o pohodlné znázornenie viacvrstvovej architektúry systému, náš tím vidí pri 3D UML široké možnosti. V tíme si kladieme za hlavnú úlohu rozšíriť existujúci prototyp o požadovanú funkcionálnu, ktorá zatiaľ nebola implementovaná a zároveň skvalitniť existujúci prototyp tak, aby sa jazyk 3D UML priblížil k reálne použiteľnej implementácii, ktorá by v budúcnosti uľahčovala prácu aj „bežným“ používateľom klasického UML.

Náš tím pozostáva z mnohých skúsených ľudí, ktorí majú praktické znalosti z práce na tímových projektoch vo firmách, no zároveň aj akademické výsledky. V tíme máme niekoľko ľudí, ktorí majú široké znalosti jazyka C++, ale aj ďalších, zameraných hlavne na jazyk Java, ktorý má s C++ veľa spoločného. S OGRE, ktoré je uvedené, ako druhá požadovaná technológia sa už niektorí z nás rovnako stretli. Niekoľkí členovia tímu sa taktiež zaujímajú o počítačovú grafiku, čo je pri projekte daného zamerania veľmi prospešné. Ako silný tím, so široko zameranými členmi sme však schopní zájsť aj za hranice zadania projektu a prichádzať aj s vlastnými nápadi na zlepšenie.

Veríme, že projekt 3D UML dokáže náš tím dotiahnuť na novú, vyššiu úroveň, ktorá by v budúcnosti mohla byť reálne využitá, či už v akademickej, vedeckej, alebo biznis oblasti.

### B.3. Rozvrh

		8-9	9-10	10-11	11-12	12-13	13-14	14-15	15-16	16-17	17-18	18-19	19-20	20-21				
Po.	H.Baranovičová						Kódovanie		Tímový projekt I			Výskum informačných systémov						
	F. Juras																	
	M. Kudláč																	
	L. Markovič																	
	M. Melis	Vyhľadavanie informácií														Výskum informačných systémov		
	M. Valovič																	
	A. Železnák							Kódovanie									Výskum informačných systémov	
Ut.	H.Baranovičová									Výskum softvérových systémov								
	F. Juras	Pokročilé databázové technológie												Architektúra informačných systémov				
	M. Kudláč			Pokročilé databázové technológie														
	L. Markovič									Výskum softvérových systémov								
	M. Melis	Pokročilé databázové technológie		Strategický manažment				Manažment					Architektúra informačných systémov					
	M. Valovič									Výskum softvérových systémov								
	A. Železnák	Pokročilé databázové technológie												Architektúra informačných systémov				
St.	H.Baranovičová		Softvérové jazyky						Manažment v informačných systémoch / softvérovom inžinierstve				Formálne stretnutie k tímovému projektu					
	F. Juras																	
	M. Kudláč																	
	L. Markovič			Softvérové jazyky														
	M. Melis		Softvérové jazyky															
	M. Valovič																	
	A. Železnák																	
Št.	H.Baranovičová								Architektúra softvérových systémov									
	F. Juras																	
	M. Kudláč																	
	L. Markovič								Architektúra softvérových systémov									
	M. Melis											Investičné analýzy						
	M. Valovič								Architektúra softvérových systémov									
	A. Železnák																	
Pia.	H.Baranovičová																	
	F. Juras	Kódovanie		Pokročilé databázové technológie														
	M. Kudláč																	
	L. Markovič																	
	M. Melis				Pokročilé databázové technológie													
	M. Valovič																	
	A. Železnák	Kódovanie		Pokročilé databázové technológie														