

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Dokumentácia k riadeniu projektu

Autori: Ondrej Čičkán, Šimon Dekrét, Dušan Javorník,
Dušan Jom, Miroslav Laco, Anton Ján Vrban

Vedúci projektu: Ing. Karol Rástočný, Phd

Akademický rok: 2016/2017

Dátum odovzdania: 21.11.2016

Obsah

| | | |
|----------|--|-----------|
| 1 | Úvod..... | 3 |
| 1.1 | Prehľad dokumentu | 3 |
| 2 | Úlohy členov tímu a podiel práce | 4 |
| 2.1 | Členovia tímu..... | 4 |
| 2.2 | Manažérske roly | 4 |
| 2.3 | Krátkodobé roly | 4 |
| 2.4 | Podiel práce na jednotlivých častiach dokumentácie | 4 |
| 3 | Aplikácie manažmentov | 5 |
| 3.1 | Manažment komunikácie | 5 |
| 3.2 | Manažment dokumentácie | 5 |
| 3.3 | Manažment integrácie a podpory vývoja | 5 |
| 3.4 | Manažment verziovania systému | 5 |
| 3.5 | Manažment testovania | 5 |
| 3.6 | Manažment posúdenia kódu..... | 6 |
| 4 | Sumarizácia šprintov | 6 |
| 4.1 | Prvý šprint – Aragorn | 6 |
| 4.2 | Druhý šprint – Boromir | 7 |
| 4.3 | Tretí šprint – Celeborn | 8 |
| 5 | Používané metodiky | 10 |
| 5.1 | Metodika dokumentovania | 10 |
| 5.2 | Metodika testovania | 10 |
| 5.3 | Metodika posudzovania kódu | 10 |
| 5.4 | Metodika komunikácie | 10 |
| 5.5 | Metodika verzií | 10 |
| 5.6 | Metodika konvencií písania zdrojového kódu | 10 |
| 6 | Globálna retrospektíva | 11 |

1 Úvod

Tento dokument hovorí o riadení projektu v rámci predmetu Tímový projekt I,II v roku 2016/2017. Projekt, ktorý bol predmetom vývoja v našom tíme, má za úlohu uľahčiť aktívnym programátorom ich prácu.

1.1 Prehľad dokumentu

Dokument je rozdelený do niekoľkých kapitol. V kapitole 2 sú spomenutí členovia tímu z manažérskymi úlohami a krátkodobými rolami, ktoré zastávajú. Kapitola 3 rozoberá jednotlivé šprinty počas vývoja. V kapitole 5 sú v krátkosti spomenuté používané metodiky v našom tíme s referenciami na dokumenty, v ktorých sú tieto metodiky dopodrobna rozpracované. Kapitola 6 je venovaná globálnej retrospektíve za celý semester. V prvom kontrolnom bode nie je zahrnutá, keďže globálna retrospektíva ešte nebola.

2 Úlohy členov tímu a podiel práce

2.1 Členovia tímu

- Ondrej Čičkán
- Šimon Dekrét
- Dušan Javorník
- Dušan Jom
- Miroslav Laco
- Anton Ján Vrban

2.2 Manažérske roly

V nasledujúcej tabuľke sú zobrazené manažérske úlohy s prislúchajúcim členom, ktorý je za úlohu zodpovedný.

| <i>Manažérska úloha</i> | <i>Zodpovedný člen</i> |
|---|------------------------|
| <i>Manažér komunikácie</i> | Ondrej Čičkán |
| <i>Manažér dokumentácie</i> | Šimon Dekrét |
| <i>Manažér kvality</i> | Dušan Javorník |
| <i>Manažér verzii</i> | Dušan Jom |
| <i>Manažér testovania</i> | Miroslav Laco |
| <i>Manažér systémovej integrácie a podpory vývoja</i> | Anton Ján Vrban |

Tabuľka 1 - Manažérske úlohy

2.3 Krátkodobé roly

| <i>Úloha</i> | <i>Zodpovedný člen</i> |
|------------------------------------|--|
| <i>Scrum master</i> | Dušan Javorník, Miroslav Laco, Dušan Jom |
| <i>Zapisovateľ na stretnutiach</i> | Ondrej Čičkán, Šimon Dekrét, Dušan Jom |
| <i>Webová stránka</i> | Anton Ján Vrban |

Tabuľka 2 - Krátkodobé úlohy

2.4 Podiel práce na jednotlivých častiach dokumentácie

V nasledujúcich tabuľkách je zobrazené, kto sa ktorej časti dokumentácie venoval.

Dokumentácia k riadeniu projektu

| <i>Časť dokumentácie</i> | <i>Zodpovedný člen</i> |
|---|--------------------------------|
| <i>Úvod</i> | Šimon Dekrét |
| <i>Úlohy členov tímu a podiel práce</i> | Ondrej Čičkán, Šimon Dekrét |
| <i>Aplikácie manažmentov</i> | Miroslav Laco |
| <i>Sumarizácia šprintov</i> | Ondrej Čičkán, Anton Ján Vrban |

Tabuľka 3 - Podiel práce na častiach dokumentácie riadenia projektu

Dokumentácia k inžinierskemu dielu

| <i>Časť dokumentácie</i> | <i>Zodpovedný člen</i> |
|---------------------------------|--|
| <i>Úvod</i> | Šimon Dekrét |
| <i>Globálne ciele projektu</i> | Dušan Jom |
| <i>Celkový pohľad na systém</i> | Anton Ján Vrban, Miroslav Laco, Dušan Javorník |

Tabuľka 4 - Podiel práce na častiach dokumentácie k inžinierskemu dielu

3 Aplikácie manažmentov

V tejto kapitole je uvedený popis činností potrebných pre riadenie projektu.

3.1 Manažment komunikácie

Na komunikáciu v tíme mimo tímových stretnutí sme si zvolili nástroj #Slack. Zo začiatku mali členovia problém si privyknúť na aktívne používanie tohto nástroja a často využívali správy len cez sociálne siete. Po prvom šprinte sa aktivita začala zlepšovať a členovia tímu boli dostupní cez tento komunikačný kanál. Za nejaký čas sa začala komunikácia na #Slacku zlepšovať. Členovia tímu začali využívať aj dedikované kanály na komunikáciu a pri riešení problémov sa mohli zapájať viacerí.

3.2 Manažment dokumentácie

V prípade projektov ako je náš, je veľmi dôležité dbať na dodržiavanie priebežného písania dokumentácií. V našom prípade bolo treba písať technickú dokumentáciu a používateľskú príručku. Písanie sa nám spočiatku nedarilo a čiastočne to bolo zapríčinené aj tým, že sme sa zoznamovali s prostredím a veľa sme toho nevytvorili. V ďalších týždňoch sa to zlepšovalo.

3.3 Manažment integrácie a podpory vývoja

Nakoľko projekt CodeCrutches je vyvíjaný ako súčasť už existujúceho systému, je nutné pri jeho vývoji brať ohľad aj na ostatné časti systému a zabezpečiť ich vzájomnú kooperáciu. Dôležitá je nie len funkčnosť samotného systému, ale aj kvalitný architektonický návrh a dodržiavanie zásad súvisiacich s udržateľnosťou projektu. Jednou z nich je zásada dodržiavania kódových konvencií popísaná v metodike, ktorou sa pri vývoji v tíme riadime s cieľom písať zdrojový kód tak, aby bol ľahko pochopiteľný a použiteľný nie len pre nás, ale aj ostatných vývojárov, ktorí budú v budúcnosti s týmto systémom pracovať.

3.4 Manažment verziovania systému

Verziovanie nami implementovaného softvéru je zabezpečené prostredníctvom systému Git. Ten nám zabezpečuje, že každý člen tímu môže svojvoľne pracovať a upravovať svoju lokálnu verziu kódu, bez toho aby zasahoval do vývoja človeka pracujúceho na inej funkcionalite systému. Zároveň sa vďaka zvoleným pravidlám a princípom pomáha efektívne monitorovať životný cyklus vývoja softvéru a odhaľovať chyby prípadne ľudí, ktorí mohli danú chybu spôsobiť.

3.5 Manažment testovania

Unit testovanie implementovaných funkcionalít v systéme je neoddeliteľnou súčasťou vývoja robustnej a spoľahlivej aplikácie pokiaľ sa na vývoji podieľa viac členov. Potrebu implementácie unit testov sme pocítili už pred koncom prvého šprintu. Žiaden z členov tímu nemal doposiaľ s unit testovaním kódu skúsenosti, preto si manažér pre testovanie problematiku našťudoval a vypracoval odporúčanie, ktoré po 1. šprinte predniesol členom tímu. Počas 2. šprintu manažér pre testovanie zapracoval unit testing do implementácie a pripravil si prezentáciu k unit testingu pre ostatných členov tímu. Odprezentované odporúčania a konvencie k unit testingu začal tím postupne aplikovať už pred koncom 2. šprintu a počas 3. šprintu. Koncom 3. šprintu sa členovia s unit testingom plne oboznámili a unit testing je riadne zaradený do procesu implementácie softvéru. Osvojiť si princípy unit testingu rozsiahlejšieho projektu pomôže všetkým členom tímu pri riešení budúcich rozsiahlejších projektov na poste programátorov, ale aj ušetriť zdroje na integráciu robustného kódu na poste manažérov.

3.6 Manažment posúdenia kódu

Posudzovania kódu sú dôležité pre zabezpečenie kvality samotného kódu. V neposlednom rade má taktiež za úlohu oboznámiť ostatných členov tímu s kódom ako s celkom a tiež vzdelávať členov tímu v používaných implementačných technológiách. Posúdenia kódu sa po začiatku prvého šprintu javili ako výborný prostriedok na oboznámenie sa s jazykom C# členmi tímu, ktorí v ňom nikdy nerobili. Expertnejší členovia tímu, čo sa tohto jazyka týka, mohli úpravami kódu poučiť ostatných členov a tí sa mohli vzdelávať z kódu expertnejších členov. Pull request ako metóda posúdenia kódu vznikla zo vzájomnej dohody na 2. stretnutí a táto metóda sa začala aplikovať. Po 1. šprinte sa každá User story dostala do stavu pull requestu a bola členmi tímu posúdená, pripomienkovaná a všetky pull requesty boli nakoniec akceptované. Systém posudzovania kódu prostredníctvom pull requestov je v súčasnosti povinnou súčasťou vývoja úloh a aktivita a promptnosť pri pripomienkovaní kódu stúpa.

4 Sumarizácia šprintov

4.1 Prvý šprint – Aragorn

V prvom šprinte sme sa dohodli na hlavných metodikách a procesoch využívané pri práci v tíme. Vytvorili sme tímový mail, spoločný cloudový priestor na zdieľanie dokumentov a vytvorili novú doménu v #slacku slúžiacu na komunikáciu v tíme. Členovia si rozdelili prvé roly a zodpovednosti. Anton Ján Vrban vytvoril webovú stránku tímu a postaral sa o jej nasadenie na určený server. Ondrej Čičkán založil potrebné kontá používané v tíme na komunikáciu a vypracoval k nim príslušné metodiky. Šimon Dekrét sa ujal procesu dokumentácie a vytvoril šablóny pre zápisnice zo stretnutí a na retrospektívu po šprintoch. Dušan Jom vypracoval metodiku na prácu s gitom a pravidlá pre vytváranie a manažment vetiev. Dušan Javorník sa zaoberal spôsobmi kontroly zdrojového kódu a pravidlám využívania pull-requestov v TFS. Miroslav Laco vypracoval príklady a metodiky k testovaniu a k mockovaniu.

V tomto šprinte si tím nainštaloval potrebný softvér a rozbehal vývojové prostredie. Zoznámili sme sa aj s existujúcimi projektami, ktoré budeme využívať pri vývoji. Za úlohu mal tím založiť nový projekt pre rozšírenie Visual Štúdia a implementovať funkcionality na začatie a ukončenie práce na úlohe a jej uloženie na ITMaintenance serveri.

Celková *velocity* tohto šprintu mala hodnotu 10.

| Title | State | Assigned To | Story Points | Completed Work |
|--|----------|-----------------|--------------|----------------|
| 4 Ukončenie práce na úlohe | Resolved | Dusan Jom | 3 | |
| Položka v menu na ukončenie úlohy | Closed | Dusan Jom | | 2 |
| Načítanie zoznamu prebiehajúcich úloh | Closed | Simon Dekret | | 2 |
| Formulár so zoznamom úloh | Closed | Anton Jan Vrban | | 3 |
| Aktualizácia stavu úlohy v ITM | Closed | Simon Dekret | | 1 |
| 4 Zmena predpokladaného ukončenia úlohy | Closed | Miroslav Laco | 2 | |
| Úprava formuláru na ukončenie úlohy pre editáciu dátumu ukončenia | Closed | Miroslav Laco | | 1 |
| Vytvorenie formuláru(popupu) pre zmenu dátumu | Closed | Miroslav Laco | | 4 |
| Vytvorenie formuláru pre ukončenie úlohy | Closed | Miroslav Laco | | 0 |
| 4 Určenie predpokladaného ukončenia práce na úlohe | Closed | Simon Dekret | 1 | |
| Rozšírenie Objektu ITM o predpokladaný dátum ukončenia úlohy | Closed | Dusan Javornik | | 4 |
| Rozšíriť formulár o začiatku úlohy o field pre dátum predpokladaného ukon... | Closed | Simon Dekret | | 0,45 |
| 4 Zaevidovanie začatia novej úlohy | Closed | Ondrej Cickan | 3 | |
| Vytvorenie projektu VS pluginu | Closed | Ondrej Cickan | | 1 |
| Zaregistrovanie položky menu CodeCrutches vo VS s podmenu Začni úlohu | Closed | Ondrej Cickan | | 1,45 |
| Formulár pre zadanie opisu úlohy | Closed | Anton Jan Vrban | | 2,5 |
| Odoslanie ulohy na ITM | Closed | Ondrej Cickan | | 6 |
| 4 Spojenie identifikácie používateľa s DevACTs | Closed | Anton Jan Vrban | 1 | |
| Vychítanie hodnoty z registra, kde je používateľ | Closed | Anton Jan Vrban | | 1 |

Obrázok 1: Zoznam úloh z prvého šprintu

4.2 Druhý šprint – Boromir

Cieľom druhého šprintu bolo rozšírenie pre Visual Studio vytvorené v rámci prvého šprintu rozšíriť o možnosť prerušenia práce na aktuálne rozrobenej úlohe a výber inej aktívnej úlohy.

V rámci šprintu sme taktiež založili nový webový projekt, ktorého úlohou bolo zatiaľ len načítanie a zobrazenie zoznamu úloh všetkých používateľov vrátane ich stavu. Tento webový projekt by mal v budúcnosti slúžiť ako manažérsky nástroj, prostredníctvom ktorého bude možné sledovať úlohy jednotlivých programátorov a ich vykonávané aktivity v rámci týchto úloh. Z povahy tohto projektu preto vyplynula požiadavka na automatickú synchronizáciu dát o úlohách medzi našim systémom a ostatnými známymi *Issue trackermi* (aby sa zamedzilo nutnosti práce s viacerými nástrojmi súčasne). Súčasťou šprintu preto bola analytická User Story, ktorej cieľom bolo preskúmať možnosti prepojenia s nástrojom TFS.

V rámci šprintu sa nám podarilo úspešne zaviesť a využívať revíziu kódu pomocou *pull requestov*, avšak identifikovali sme potrebu ich zrýchlenia v budúcich šprintov – stanovili sme si pravidlá, ktoré sa pri vytváraní *pull requestov* a žiadaní o revíziu kódu budú dodržiavať. Zlepšila sa komunikácia a spoločná práca v tíme a práca bola priebežnejšia ako v prvom šprinte vďaka lepšiemu určovaniu priority a závislostí medzi jednotlivými úlohami.

Napriek zlepšenej činnosti tímu sa nám nepodarilo dokončiť jednu *User Story*, ktorá bola preto presunutá do ďalšieho šprintu. Celková *velocity* tohto šprintu po presunutí nedokončenej *User Story* mala hodnotu 9.

Ako potrebu zlepšenia do budúceho šprintu sme identifikovali integráciu *Unit testov* a zlepšiť dodávanie produktu načas.

| Title | State | Assigned To | Story Points | Completed Work |
|--|--------|-----------------|--------------|----------------|
| Prerušenie práce na aktualne rozrobenej ulohy | Closed | Simon Dekret | 2 | |
| Rozsiriť objekt ITM o stav ulohy | Closed | Simon Dekret | | 1,5 |
| Rozsiriť formulár pre manipuláciu s ulohami o prerušenie ulohy | Closed | Simon Dekret | | 3 |
| Rozsiriť formulár manipulácie s ulohami o stavy uloh | Closed | Simon Dekret | | 2 |
| Nastavenie migrácie pre Mongo a webový projekt | Closed | Ondrej Cickan | 1 | |
| Vytvorenie nugetu pre Mongo migrations v DevACTs projekte | Closed | Ondrej Cickan | | 1 |
| Implementovanie vytvoreného nuget balíku vo webovom projekte | Closed | Ondrej Cickan | | 0,25 |
| Opatovne zahajenie práce na ulohy | Closed | Ondrej Cickan | 1 | |
| Rozsiriť formulár manipulácie s ulohami o pokračovanie v práci | Closed | Ondrej Cickan | | 5,5 |
| Preskúmanie možnosti pripojenia na TFS | Closed | Dusan Javornik | 1 | |
| Analýza | Closed | Dusan Javornik | | 4 |
| Zmena ulohy na ktorej sa pracuje | Closed | Miroslav Laco | 1 | |
| Prepnutie stavu aktívnej ulohy na paused a aktivovanie novej ul... | Closed | Miroslav Laco | | 1 |
| Zobrazenie stavu uloh | Closed | Anton Jan Vrban | 3 | |
| Založiť webový projekt | Closed | Anton Jan Vrban | | 1 |
| Nacítať zoznam uloh z ITM | Closed | Anton Jan Vrban | | 2 |
| Nadizajnovat a implemtnovat view pre model zoznamu uloh | Closed | Anton Jan Vrban | | 1 |

Obrázok 2: Zoznam úloh v rámci 2. šprintu

4.3 Tretí šprint – Celeborn

Tretí šprint bol zameraný najmä na prácu na webovom projekte, ktorý bol vytvorený v predchádzajúcom šprinte.

Cieľom bolo vytvoriť tejto webovej aplikácii responzívne používateľské rozhranie, ktoré bude klásť dôraz na intuitívnosť a interaktivitu s používateľom podľa návrhov, ktoré sme si na začiatku šprintu nakreslili na tabuľu. Webová aplikácia tiež mala poskytovať možnosť prihlasovania pomocou účtu z platformy DevACTs, vytvárania projektov (za využitia databázy MongoDB), pridávanie používateľov k projektom a správu ich oprávnení.

Po skončení tohto šprintu na jeho retrospektíve sme vyjadrili spokojnosť s dizajnom používateľského prostredia vytvoreného webového projektu, ako aj s implementovanými funkcionalitami. Osvedčili sa nám tiež pravidlá ohľadom *pull requestov*, ktoré sme si definovali v predchádzajúcom šprinte, vďaka ktorým sa rýchlosť ich vyriešenia značne zvýšila. Boli sme tiež spokojní s tým, ako prebiehala komunikácia a spolupráca v rámci tímu, avšak bohužiaľ nie medzi všetkými členmi.

Ani v tomto šprinte sa nepodarilo uzavrieť *User Story*, ktorá bola prenesená už z druhého šprintu. Taktiež nebola uzavretá *User Story*, ktorá nespĺňala kritéria na jej uzavretie (chýbajúca používateľská a technická dokumentácia). Celková *velocity* tohto šprintu mala hodnotu 12.

Pri retrospektíve šprintu sme identifikovali možnosti, ktoré nám môžu pomôcť zlepšiť plánovanie a prácu na projekte. Jedná sa o zadefinovanie si pracovnej dostupnosti jednotlivých členov tímu, potrebu *team-buildingu* pre zlepšenie komunikácie a spolupráce v tíme, podrobnejšie popisovať a analyzovať úlohy a aplikovať mikromanažment v rámci *User Story*.

Na konci šprintu Šimon Dekrét navrhol zriadiť nový komunikačný kanál *#planning*, v ktorom vždy na začiatku šprintu bude *Scrum master* pridávať diagram zobrazujúci závislosti medzi

jednotlivými úlohami šprintu, pričom úlohy v prvej línii (od ktorých závisia ďalšie) budú mať vždy určený termín uzatvorenia najneskôr v polovici šprintu (týždeň od jeho začiatku) s cieľom zlepšiť plánovanie a pridelovanie úloh konkrétnym ľuďom, a tým pádom zlepšenia priebežnej práce na úlohách.

| Title | State | Assigned To | Story Points | Completed Work |
|--|----------|-----------------|--------------|----------------|
| ↳ Vytvorenie projektu | Closed | Miroslav Laco | 2 | |
| Vytvorenie formularu pre vytvorenie projektu | Closed | Miroslav Laco | | 5 |
| Vytvorenie zakladu projektovej databazy | Closed | Miroslav Laco | | 4,5 |
| ↳ Pridavanie a nastavenie prav pouzivatelov priradenych k projektu | Closed | Ondrej Cickan | 3 | |
| Vytvorenie formularu vsetkych pouzivatelov projektu | Closed | Ondrej Cickan | | 10 |
| Pridanie formularu pre pridanie pouzivatela | Closed | Simon Dekret | | 5 |
| Pridanie zmeny role do formularu | Closed | Simon Dekret | | 3 |
| ↳ Prihlasovanie pomocou DevACTs uctu | Resolved | Dusan Javornik | 2 | |
| Implementovat vyuzitie autentifikacie a autorizacie z centralnych... | Closed | Dusan Javornik | | 10 |
| Nastylavat uvodnu obrazovku pred prvym prihlasenim | Closed | Dusan Javornik | | 3 |
| ↳ Sprava uloh vyuziva centralne sluzby | Active | Dusan Jom | 2 | |
| Nacitanie konfiguracie endpointu ITM z centralnych sluzieb | Active | Dusan Jom | | 2 |
| Vytvorenie proxy pre pripojenie na ITM | New | Dusan Jom | | 0 |
| ↳ Web CodeCrutches ma nastylowane pouzivatelske rozhranie | Closed | Anton Jan Vrban | 3 | |
| Definovanie stylu pre uvodnu obrazovku | Closed | Anton Jan Vrban | | 2 |
| Definovanie stylu pre menu projektu | Closed | Anton Jan Vrban | | 1 |
| Implementacia polozky tasks | Closed | Anton Jan Vrban | | 1 |
| Implementovanie funkcionalit z pouzivatelskeho rozhrania | Closed | Anton Jan Vrban | | 4 |

Obrázok 3: Zoznam úloh v rámci 3. šprintu

5 Používané metodiky

Pre správne fungovanie celého tímu a dobrý vývoj je potrebné mať zadaných niekoľko metodík. My sme pracovanie s nasledujúcimi:

5.1 Metodika dokumentovania

Pre správne písanie dokumentácií používame príslušnú metodiku. Táto metodika zahŕňa opis procesov písania dokumentácií k

- Riadenie projektu
- Inžinierske dielo
- Zápisnice
- Retrospektívy

Link na metodiku: <https://1drv.ms/b/s!AsjBbCgWL-mQhTRUyLk9h4HWA7u5>

5.2 Metodika testovania

Pri vývoji implementujeme unit testing prostredníctvom dostupného frameworku. Na testovanie používame pri vývoji Microsoft Unit Test Framework for Managed Code. Ako mockovací framework sme si zvolili Microsoft Fakes, ktorý vyžaduje IDE MS Visual Studio 2015 Enterprise. Vybrané frameworky sú priamo integrované v IDE MS Visual Studio a zvolili sme si ich práve vďaka silnej podpore v IDE.

Link na metodiku: <https://1drv.ms/b/s!AsjBbCgWL-mQhTBUyLk9h4HWA7u5>

5.3 Metodika posudzovania kódu

Na prvotnú kontrolu a posúdenie kódu pred jeho pridaním do hlavnej vývojovej vetvy v systéme Git používame techniku Pull Request. Vďaka nemu je možné o daných zmenách diskutovať a kód upravovať, až kým nenadobudne dostatočnú kvalitu na pridanie.

Link na metodiku: <https://1drv.ms/b/s!AsjBbCgWL-mQhTNUyLk9h4HWA7u5>

5.4 Metodika komunikácie

Pri spolupráci viacerých ľudí na jednom projekte je dôležitá komunikácia a šírenie informácií medzi jednotlivými členmi tímu. Táto metodika opisuje pravidlá pri používaní komunikačného nástroja #Slack a rozdelenie komunikácie na základe jej účelu a témy do jednotlivých kanálov.

Link na metodiku: <https://1drv.ms/b/s!AsjBbCgWL-mQhTJUyLk9h4HWA7u5>

5.5 Metodika verzií

Pre vývoj softvéru je nevyhnutné zabezpečiť, aby mohli developeri pracovať na zdrojovom kóde nezávisle, teda každý na svojej verzii. Správa a riadenie verzií softvéru je zabezpečené pomocou systému Git. Táto metodika popisuje samotný systém, jeho princípy a stanovené pravidlá pre prácu s ním.

Link na metodiku: <https://1drv.ms/b/s!AsjBbCgWL-mQhSpUyLk9h4HWA7u5>

5.6 Metodika konvencií písania zdrojového kódu

V projekte, ktorý je vyvíjaný agilnou metódou (tímom ľudí kde dochádza k častým zmenám v kóde) je potrebné mať presne definované pravidlá písania zdrojového kódu tak, aby bol ľahko pochopiteľný aj pre ostatných vývojárov. Náš projekt je navyše súčasťou väčšieho systému, na ktorého vývoji pravdepodobne budú pravdepodobne pracovať aj iné tímy programátorov

v budúcnosti. Vytvorili sme preto metodiku, ktorá definuje konvencie písania zdrojového kódu jednotlivu pre programovacie jazyky, ktoré pri vývoji používame. Dodržiavanie tejto metodiky vývojármi je overované v rámci procesu posudzovania kódu.

Link na metodiku: <https://1drv.ms/b/s!AsjBbCgWL-mQhTFUyLk9h4HWA7u5>

6 Globálna retrospektíva

Doteraz sme mali celkovo tri retrospektívy, na ktorých sme zhodnotili doterajšiu činnosť. V jednej z nasledujúcich častí máme dokumenty k jednotlivým retrospektívam. Globálnu retrospektívu za celý semester budeme mať až v decembri.

Tím č. 20

Kontakt: fiiit.team.project.no20@gmail.com

Členovia: Šimon Dekrét
Ondrej Čičkán
Miroslav Laco
Dušan Jom
Dušan Javorník
Anton Ján Vrban

Sme partia mladých a ambiciózných ľudí, ktorí sa snažia rozširovať si svoje vedomosti popri štúdiu aj aktívnou prácou na rôznych komerčných projektoch. Tieto projekty boli pritom rôzneho charakteru - niektorí z nás vo svojom zamestnaní vytvárajú desktopové aplikácie, iní webové aplikácie a iní sa zas venujú najmä designu a použiteľnosti. Ako tím sme teda veľmi dobre vyvážení a máme príležitosť sa navzájom zdokonaľovať.

V tíme máme skúsenosti hlavne s programovacími jazykmi Java, PHP a C#, ktoré využívame aj v zamestnaniach. Okrem nich sme pracovali aj s HTML, CSS, Javascriptom, Ruby on Rails, Python, C++, Node.js. Samozrejmosťou je tiež práca s databázami rôznych typov. Viacerí z nás majú v inžinierskom štúdiu zapísané predmety Objavovanie znalostí a Objektovo orientovanú analýzu a návrh softvéru.

V tíme sa pomerne dobre poznáme, a väčšina z nás spolu už v nejakej forme spolupracovala. Ondrej so Šimonom a Dušan Javorník s Miroslavom Lacom pracovali spolu na väčšine tímových projektoch počas bakalárskeho štúdia. Šimon, Miroslav a Dušan Jom sú súčasní kolegovia v práci, takže pri tímovej komunikácii a spolupráci nebudeme mať väčšie problémy.

Motivácia

Téma: 5. Pomôcky pre aktívnych programátorov

Všetci členovia nášho tímu sú aktívni programátori- či už vo voľnom čase, v škole, ale aj v zamestnaní. Pravidelne sa pritom stretávame s každodennými programátorskými rutinami, ktoré dobre poznáme. Taktiež každodenne pracujeme so súčasnými pomôckami pre programátorov a vidíme v tejto oblasti veľký potenciál pre rozšírenia, či vylepšenia takýchto nástrojov. Okrem iného využívame pre svoju prácu vývojové prostredia Visual Studio, Eclipse, IntelliJ, CLion a ostatné nástroje, ktoré sú dnes už takmer nutnosťou pri vývoji softvéru. Spomeňme napríklad Git, CVS, TFS, atď. Preto si myslíme, že máme veľmi dobrý prehľad v množine aktivít, ktoré programátor vykonáva

Hlavnou motiváciou nášho tímu je navrhnuť, opísať a implementovať pomôcky pre takých programátorov, ako sme my. Pomôcky, ktoré nielen náš každodenný programátorský život uľahčia, ale užívateľom prácu s vývojom softvéru aj mnohonásobne spríjemnia. Následne by sme veľmi radi úspešné pomôcky aj aktívne využívali a obohatili tak programátorský život sebe aj iným.

V neposlednom rade by sme radi spomenuli, že disponujeme nezanedbateľnými skúsenosťami so začiatkom práce na rozsiahlych existujúcich softvérových riešeniach. Preto predpokladáme, že práca s existujúcou infraštruktúrou DevACTs bude bezproblémová a oboznámenie sa s DevACTs bude svižné a rýchle. Budeme tak môcť, dúfame, čo najskôr pracovať na cieľoch nášho preferovaného zadania.

Téma 14. Extrakcia dát z webu

Motiváciou pre výber tejto témy je najmä práca v stále rastúcom webovom prostredí. Čím ďalej, tým viac informácií, komerčných či štátnych, ale aj súkromných je uverejnených na webe a nájsť dobré a správne informácie je stále náročnejšie. Takýto projekt ma obrovský potenciál na využitie širokou škálou používateľov.

Keďže každý z nás v tíme má skúsenosti s prácou na projektoch s reálneho sveta, pojmy ako Test-Driven Development a Continuous Integration nám nie su neznáme. K tomu našou náplňou práce je tvorba webového obsahu, takže vieme, ako taká hierarchia webovej stránky vyzerá.

Téma 16. Inteligentný sklad [SmartStore]

Motiváciou pre výber tejto témy nám bola najmä skutočnosť, že sa jedná o projekt z komerčného prostredia, v ktorom máme skúsenosti s viacerými projektami. Niektorí z nás už spolupracovali s viacerými e-shopmi, či pracovali na vývoji informačných systémov pre podobne veľkých zákazníkov ako je firma Martinus.sk. Práca s reálnymi zákazníkmi a dátami je totiž presne to, čomu sa chceme v našej kariére po skončení štúdia aj ďalej venovať.

Téma e-commerce je nám obzvlášť blízka a pri svojej práci sa aktívne zaujímame o hľadanie riešení, ktoré môžu nášmu zákazníkovi pomôcť akcelerovať jeho biznis. Jeden z členov nášho tímu sa podieľal na príprave celoslovenskej súťaže pre elektronické obchody, ktorej cieľom bolo pomôcť im inovatívnymi riešeniami prilákať viac zákazníkov.

Práve v systéme, ktorý by pomáhal elektronickým obchodom riešiť nie len vybavovanie objednávok (čo je väčšinou jedinú, čo začínajúce e-shopy od vývojárov požadujú), ale aj efektívne využívať ich skladové priestory a plánovať logistické operácie medzi nimi, vidíme veľký potenciál. V prípade, že by bol tento systém univerzálny a ľahko integrovateľný aspoň na niektoré z bežne dostupných e-shop systémov, mohol by sa z neho stať veľmi užitočný a žiadaný nástroj pomáhajúci optimalizovať náklady obchodom, ktoré by ho využívali.

Príloha A - Priorita tém

1. 5. Pomôcky pre aktívnych programátorov
2. 14. Extrakcia dát z webu
3. 16. Inteligentný sklad
4. 17. Tvorba vzdelávacích simulácií
5. 11. Prepájanie dát o vývoji softvéru
6. 9. Zber a vyhodnocovanie požiadaviek
7. 1. Vizualizácia informácií v obohatenej realite
8. 4. Simulácia správania dronov v roji
9. 12. Manažment zdravotného stavu pacienta
10. 6. Servisný modul pre stratosferický balón
11. 13. Vyhľadávanie so sémantikou
12. 20. Navigácia v budove
13. 18. Monitorovanie a vyhodnocovanie fyziologických procesov človeka
14. 7. 3D UML
15. 2. Rekonštrukcia 3D scény
16. 3. Nový návrh systému pre MOD
17. 21. 3D simulovaný robotický futbal
18. 19. Aplikácia deterministického ethernetu
19. 10. Monitorovanie bezpečnostných udalostí

Príloha B - Rozvrh tímu

| | 7:00- 8:00 | 8:00- 9:00 | 9:00- 10:00 | 10:00- 11:00 | 11:00- 12:00 | 12:00- 13:00 | 13:00- 14:00 | 14:00- 15:00 | 15:00- 16:00 | 16:00- 17:00 | 17:00- 18:00 | 18:00- 19:00 | 19:00- 20:00 | 20:00- 21:00 |
|-----------------|---------------|---------------------------|-----------------------------------|-----------------|-------------------------|-----------------|-----------------|------------------|-----------------|-----------------|---------------------------|-----------------|-----------------|-----------------|
| Pondelok* | | | Javorník Jom | | Jom | | | | | | | | | |
| Utorok | | | Čičkán Dekrét Laco Vrban | | Javorník Laco Jom | | | Čičkán Dekrét | | | | | | |
| | | Laco Jom Javorník | | | | | | | | | | | | |
| Streda | | | | | | | Laco Vrban | | | | | | | |
| Štvrtok | | Čičkán Dekrét Vrban | | | | | | | | | Čičkán Dekrét Vrban | | | |
| Piatok | | | | | | | | | Javorník | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| Prednášky | | | | | | | | | | | | | | |
| Cvičenia | | | | | | | | | | | | | | |
| Preferovaný čas | | | | | | | | | | | | | | |

*V pondelok by sme preferovali voľný, kvôli pracovným povinnostiam. Ak by to časovo nevychádzalo, povinný blok hodín by sme ocenili viac vo večerných hodinách.

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

Metodika dokumentácie

Tím č.20 – RavensTeam

Účel metodiky

V rámci predmetu Tímový projekt vytvárame veľké množstvo dokumentov. Každý dokument musí obsahovať určité časti na to, aby spĺňal požadovanú kvalitu. V tejto metodike sú rozobrané jednotlivé typy dokumentov, process vytvárania dokumentov a konvencie používané pri písaní dokumentov.

Všetky dokumenty vytvorené naším tímom, sú uložené v zdieľanom adresari *RavensTeam* na OneDrive. Každý člen má k týmto dokumentom prístup a môže ich kedykoľvek aktualizovať.

Zdieľaný adresár je rozdelený na niekoľko podadresárov združujúcich súvisiace dokumenty:

- Dokumenty k inžinierskemu dielu
 - Big picture – Inžinierske dielo
 - Moduly systému
- Dokumenty k riadeniu projektu
 - Big picture – Riadenie projektu
 - Export evidencie úloh
 - Metodiky
- Retrospektívy
- Zápisnice zo stretnutí

Dokumentovanie po každom šprinte

Každý user story na to aby bol ukončený musí byť nielen správne naimplementovaný a otestovaný ale musí byť k nemu napísaná aj zodpovedajúca dokumentácia. V tomto prípade sa jedná o *technickú dokumentáciu* a *používateľskú príručku*. Niektoré úlohy, ktoré sa naimplementujú sa hodí viac zdokumentovať v technickej dokumentácii a niektoré zase v používateľskej príručke.

Dokumentovanie riadenia projektu

Jeden z dokumentov, ktorý sa priebežne v daných termínoch odovzdáva je **dokumentácia riadenia projektu** – “**Big picture**”. Tento dokument zahŕňa časti, ktoré dajú externému človeku obraz o práci v tíme. Tento dokument je vytváraný priebežne a obsahuje nasledujúce časti:

- Úvod
- Role členov tímu a podiel práce
- Aplikácia manažmentov
- Sumarizácia šprintov
- Používané metodiky
- Globálna retrospektíva
- Preberacia protokoly
- Retrospektívy

V rámci tejto časti sú vypracované metodiky, ktoré sme si v tíme rozdelili. Každý člen vytvoril osobitný dokument k svojej metodike. Tento dokument je referencovaný v “Big picture” dokumente. Aktualizovať dokument k metodikám môžu len majitelia danej metodiky. V prípade nejasností môžu ostatní členovia tímu pridávať komentý do tohoto dokumentu.

Dokumentovanie inžinierskeho diela

Ďalším dokumentom, ktorý je potrebné neustále aktualizovať a ktorý sa taktiež priebežne odovzdáva je **dokumentácia k inžinierskemu dielu – “Big picture”**. V technickej dokumentácii sú jednotlivé modely podrobne opísané a v inžinierskom diele je zhrnutie celého systému. Je to akoby zhrnutie technickej dokumentácie a zaobalenie o celkový pohľad na vyvíjaný systém. Tento dokument obsahuje nasledujúce časti:

- Úvod
- Globálne ciele projektu
- Celkový pohľad na system

Po technickej stránke je náš projekt rozdeliteľný na niekoľko modulov. Každý z týchto modulov má vlastný dokument. V rámci každého modulu treba zohľadniť 4 fázy, ktoré avšak nie sú explicitne oddelené:

- Analýza
- Návrh
- Implementácia
- Testovanie

Zápisnice

Počas každého stretnutia píše jeden člen tímu zápisnicu. Všetky zápisnice sa nachádzajú v adresári *Zápisnice zo stretnutí*. Každá zápisnica má nasledovnú štruktúru:

- Dátum a miesto konania
- Prítomní členovia
- Priebeh stretnutia
- Poznámky zo stretnutia
- Úlohy do ďalšieho stretnutia
- Plnenie úloh za týždeň

Exportované úlohy treba taktiež vložiť do adresára *Export evidencie úloh*, ktorý sa nachádza v adresári *Dokumenty k riadeniu projektu*. Vyexportované úlohy, ktoré patria aktuálnemu šprintu budú obsahovať nasledovné stĺpce :

| Title | State | Assigned To | Completed Work | Remaining Work | Description |
|-------|-------|-------------|----------------|----------------|-------------|
|-------|-------|-------------|----------------|----------------|-------------|

Exportovať treba aj *user stories*, ktoré ešte neboli ukončené. V tomto prípade budú stĺpce nasledovné:

| Order | Title | State | Story Points | Assigned To | Description |
|-------|-------|-------|--------------|-------------|-------------|
|-------|-------|-------|--------------|-------------|-------------|

Retrospektívy

Po skončení každého šprintu sa robí retrospektíva, ktorú je taktiež potrebné dokumentovať. V rámci retrospektívy sa zaujímame o nasledovné časti:

- Ciele šprintu
- Podarilo sa nám/páčilo sa nám
- Nepodarilo sa nám/nepáčilo sa nám
- Plánujeme zlepšiť

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

Metodika testovania

Tím č.20 – RavensTeam

Účel metodiky

V rámci predmetu Tímový projekt implementujeme veľké množstvo funkcionality, ktorá je navzájom prepojená. Túto funkcionality implementujeme v tíme, pričom na implementácii jednej funkcionality môžu pracovať viacerí členovia tímu. Každý člen tímu potrebuje časť funkcionality vyvíjať nezávisle od ostatných, aby si mohol korektnosť svojej implementácie overiť. Zároveň potrebuje každý člen tímu zabezpečiť, aby svojou implementáciou neporušil korektnú funkcionality, ktorá bola už v minulosti implementovaná, a to s ohľadom nato, že člen tímu nemusí poznať pozadie predošlej implementácie (postačujú jej princípy). Preto sme zaviedli v tíme povinnosť unit testingu. **Unit testing** je spôsob testovania korektnosti implementácie určitej atomickej časti funkcionality. V tejto metodike je opísaná konvencia, postup a princípy unit testingu pri vyvíjaní tímového projektu tímom RavensTeam.

Unit testing

Doposiaľ používaným IDE pre vývoj tímového projektu tímom RavensTeam je IDE Visual Studio s napojením na Team Foundation Server. Preto sme zvolili framework pre testovanie, ktorý je priamo integrovaný do IDE Visual Studio- **Microsoft unit test framework for managed code**¹. Pre izoláciu v minulosti implementovaných funkcionalít v kóde sme zvolili rovnako framework priamo integrovaný do IDE Visual Studio- **Microsoft Fakes**². Pre používanie Microsoft Fakes je potrebné mať nainštalované IDE **Visual Studio 2015** v konfigurácii **Enterprise** s nainštalovaným **Update 3**. Pre prístup k panelu testov vo Visual Studio klikneme v hornom menu IDE na položku Test > Windows > Test Explorer.

Základné princípy

Atomicke časti funkcionality implementujeme do samostatných metód. Pokiaľ je to možné a ako z doposiaľ riešených úloh (taskov) z Team Foundation Server vyplýva, tak jeden task zodpovedá jednej alebo viacerým atomickým častiam funkcionality. Každý task má implementovať minimálne 1 samostatnú metódu v ktorej je implementovaný (prípadne združený súbor metód ak sa jedná o implementáciu Controllera, Service, Modelu..). Každá z takto implementovaných metód vrátane konštruktora triedy a komplexných get/set metód (zložitejších ako jednoduchý prístup k premennej triedy) musí mať napísaný zodpovedajúci unit test. Pre splnenie task-u je potrebné, aby napísané testy pre task a aj všetky doposiaľ napísané testy boli úspešne ukončené.

Každá ucelená funkcionality vystavuje rozhranie (Interface) a sprehľadňuje tak nielen kód samotný, ale aj zjednodušuje unit testing.

V implementácii riešenia je vytvorený práve 1 unit test projekt. Jeho štruktúra zodpovedá štruktúram ostatných projektov riešenia vynímajúc netestovateľné časti implementácie. Medzi časti implementácie, ktoré sa z testovania vynímajú patria:

¹ základný prehľad - <https://msdn.microsoft.com/en-us/library/ms182532.aspx>

² <https://msdn.microsoft.com/en-us/library/hh549175.aspx>

- Views
- User interface
- Konfiguračné súbory
- Projektové súbory
- Závislosti

Platí, že okrem výnimiek spísaných vyššie prislúcha každej metóde implementujúcej atomickú časť funkcionality unit testovacia metóda. Ďalej platí, že každému využívanému API prislúcha práve 1 Microsoft Fake mock. Mock je spôsob, ktorým sa pri preklade kódu, alebo za behu program nahrádza časť kódu tak, aby mala falošná metóda mock-u presne definovaný výstup.

Platí, že všetky závislosti testovanej metódy musia byť mockované, čiže nahradené falošnými údajmi generovanými pomocou Microsoft Fakes. Nie je dovolené pri testovaní metódy volať inú nemockovanú metódu pre zabezpečenie základného princípu unit testingu. Ďalej platí, že pre externé API, ktoré implementujú rozhranie vytvárame “stub”. Pokiaľ využívame závislosti na iných projektoch vrámci riešenia, prípadne na iných triedach vrámci projektu, alebo metódach vrámci triedy, tak je povolené používať “shim”.²

Nie je povolené poslať kód na posudzovanie pokiaľ sme implementáciu nepokryli testami. Výnimkou sú závažné bug fixy, ďalej implementácie, ktorých testovanie vyžaduje rozsiahlejšiu konzultáciu, ktorá by brzdila ďalší vývoj a implementácie, ktoré musia byť súrne zapracované do hlavnej vývojovej vetvy. Z unit testingu sa vynímajú tiež prípady, ktoré dostali výnimku povinnosti posudzovania kódu. Platí však, že pre všetky vyňaté prípady spomínané vyššie je potrebné dopracovať unit testy čo najskôr, ako to bude dodatočne možné.

Postup pri testovaní

1. Implementujeme metódu zodpovedajúcu atomickej časti funkcionality.
2. Vytvoríme testovaciu metódu pre implementovanú metódu. Pokiaľ je metóda v triede, ktorá v testovacom projekte neexistuje, vytvoríme najprv túto triedu.
3. Skontrolovanie závislostí testovanej metódy.
 - a. Ak sme pridávali do projektu závislosť na externom API, tak toto API pridáme do unit test projektu ako Microsoft Fakes.
 - b. Ak chyba mock metódy vrámci riešenia, ktorú testovaná metóda používa, tak tento mock vytvoríme prostredníctvom Microsoft Fakes.
4. Napíšeme do testovacej metódy shim-y a stub-y pre všetky závislosti metódy.
5. Zvolíme si vhodné vstupy metódy, pričom pokrývame tieto prípady:
 - Bežná očakávaná hodnota na vstupe
 - Hraničné hodnoty
 - Hodnoty mimo rozsahu
6. Determinujeme výstupy testovanej metódy na základe zvolených vstupov.
7. Pre každú sadu vstupov voláme testovanú metódu a používame asertovacie porovnanie³ očakávaného výsledku metódy so skutočným výsledkom. Poznámka- každá sada vstupov musí byť otestovaná asertívnym porovnaním v dosiahnuteľnom kóde.
8. Spustíme všetky unit testy v Test Explorer paneli.

³ <https://msdn.microsoft.com/en-us/library/ms182530.aspx>

9. Pokiaľ všetky testy úspešne prebehli, testovanie končí. Pokiaľ aspoň jeden z testov neprebehol správne, implementáciou odstránime chybu v kóde a pokračujeme bodom 8.

Odporúčania k testovaniu

Odporúča sa navrhovať implementáciu funkcionality systému tak, aby mohla jedna ucelená funkcionality vystaviť rozhranie pre systém. Potom všetky ostatné interné a externé súčasti systému využívajú miesto konkrétnych tried rozhranie pre prístup k vybraným metódam. Toto pravidlo by malo ešte silnejšie platiť pri komunikácii medzi projektami vrámci riešenia. Zabezpečí sa tak možnosť jednoducho odizolovať funkcionality v systéme do komponentov a umožní sa využívanie “stub” metód, miesto “shim” metód. Rozdiel medzi stub a shim je okrem ich implementácie (stub ju má výrazne jednoduchšiu) aj dopad na rýchlosť testov, ktorá pre nás môže byť smerodajná. Shim nahrádza časť programu priamo pri jeho vykonávaní, zatiaľ čo stub časť programu nahradí už pri preklade a beh programu nijako neovplyvní.

Odporúča sa vyvarovať duplicitným výnimkám (Exceptions) vrámci 1 metódy implementujúcej atomickú časť funkcionality. Duplicitné výnimky nie je možné bez doplňujúcej úpravy korektne otestovať a takýto test je považovaný za nedostatočný. Pokiaľ nastane situácia, kedy je nutné použiť rovnaké výnimky vrámci 1 metódy, tak tieto výnimky rozšírime pri implementácii dodatočným opisom druhu výnimky vo forme string. Dodatočný opis si vieme pri implementácii unit test metódy po zachytení výnimky asertovacím porovnaním pre string porovnať so želaným druhom duplicitnej výnimky.

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

Metodika posudzovania kódu

Tím č.20 – RavensTeam

Účel metodiky

V rámci predmetu Tímový projekt vytvárame množstvo zdrojového kódu. Programátori často robia nevedomé chyby, ktoré môžu spôsobiť nestabilitu v iných častiach projektu a takéto chyby sa môžu zanášať až do produkčného kódu. V tejto metodike je popísaný postup, akým sa riadi náš tím pri zjednocovaní kódu tak, aby nedochádzalo k zanášaniam chýb do hlavnej vývojovej vetvy.

Posudzovanie

Posudzovanie zdrojového kódu je technika určená pre odhaľovanie bežných chýb prehliadnutých počas programovania a zlepšovanie celkovej kvality kódu. Posudzovanie môže prebiehať rôzne, napr. párovým programovaním alebo formálnymi inšpekciami. Pre účely tímového projektu budeme používať posudzovanie kódu formou **Pull Requestov**.

Pull request

Vo verziovacom systéme Git, pull request umožňuje oznámiť ostatným členom tímu zmeny, ktoré sa chystá vývojár pridať do repozitára. Keď sa pull request otvorí, členovia môžu diskutovať a komentovať zmeny. Pull requestu môže byť zadaný posudzovateľ, ktorý má určiť, či sú dané zmeny v poriadku a či odsúhlasuje ich pridanie do repozitára. Pokiaľ sa v zmenách nájdu pochybné časti, žiadateľ pull requestu problémový kód upraví a commitne do vetvy, z ktorej je Pull Request žiadaný. Pokiaľ posudzovateľ nemá ďalšie námietky voči kódu, Pull Request môže byť zapracovaný.

Postup pri pull requeste

1. Zadávateľ vytvorí pull request
2. Nastaví sa branch z ktorého sa bude mergovať a do ktorého sa bude mergovať (za normálnych okolností by to mala byť vetva dev)
3. Zadávateľ určí jedného človeka, reviewera, ktorý má spraviť code review
4. Daného človeka informuje súkromnou správou na Slacku.
5. Daný človek by mal do zajtrajšieho večera urobiť code review.
6. Ak daný človek nestíha urobiť code review, pridá ako reviewera RavensTeam
7. Po spravení code review daný človek označí pull request ako approved/approved with suggestions/rejected
8. Ak je Pull Request označený ako approved/approved with suggestions všetkými reviewerami, Zadávateľ Pull Requestu môže Pull Request skompletizovať stlačením tlačidla Complete Pull Request.

Odporúčania k pull requestu

Code review je odporúčané robiť spolu so zadávateľom, ktorý tie zmeny urobil. Tým sa myslí fyzicky si za to spoločne sadnúť a jednotlivé zmeny prejsť a skonzultovať. Keďže každý člen tímu má iné povinnosti a nie vždy je možné sa stretnúť, nie je to nutné.

Ďalším odporúčaním je napísať ešte jednému ďalšiemu členovi, aby pull request len zbežne prebehol, aby o aktivitách jednotlivcov vedelo viac členov tímu. Tento ďalší človek nemusí byť zadaný ako reviewer.

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

Metodika komunikácie

Tím č.20 – RavensTeam

Účel a dedikácie metodiky

Pri práci viacerých ľudí v jednom tíme je kritický prvok práve komunikácie medzi jednotlivými členmi a šírenie informácií. Za týmto účelom využívame viaceré nástroje podporujúce vzájomnú komunikáciu v tíme.

Emailový alias

Slúži hlavne na komunikáciu s tretími stranami, a píše sa v mene celého tímu.

Emailový alias fiit.team.project.no20@gmail.com

Komunikácia v tíme - #Slack

Na vnútornú komunikáciu medzi členmi používame nástroj #Slack. Ten okrem súkromnej konverzácie medzi dvoma členmi, umožňuje vytvorenie kanálov, kde môžu prispievať viacerí členovia. Kanály umožňujú rozdeliť komunikáciu na základe určitej témy, alebo účelu, čo sprehľadňuje skupinový chat a komunikuje sa k danej téme. #Slack poskytuje dobrú podporu vo vyhľadávaní informácií v histórii správ. Navyše, umožňuje aj integráciu ďalších nástrojov.

Odkaz #Slack - <https://ravens-team-fiit-stu.slack.com/>

Pravidlá komunikácie v kanáloch

Pri vytváraní nového kanálu ho výstižne pomenujeme a do detailoch napíšeme krátky opis, načo kanál používame. Po vytvorení pozveme členov. Pri komplikovanejších pravidlách kanála sa v prvej správe napíšu bližšie informácie o jeho používaní. Táto správa sa následne pripne k tomuto kanálu (pin to).

Aktuálne kanály:

- **announcement** – slúži na posielanie dôležitých oznamov pre všetkých členom tímu. Cieľom správy je informovať ostatných a nemala by od nich očakávať žiadnu odpoveď, a preto by mala byť čo najvýstižnejšia a jasne napísaná.
- **work-report** – Priestor pre členov tímu na zhlásenie svojej aktivity od posledného stretnutia. Hlásia na čom pracovali, pracujú, s čím majú problémy a na čom budú pracovať. Pred začiatkom každého bloku hlásení, manažér komunikácie jasne oddelí konverzáciu správou v tvare:

----- [Názov dňa v týždni] DD.MM.YYYY -----

Hlásenia sa podávajú tieto dni v týždni: nedeľa.

- **usefull-links** – užitočné odkazy na rôznorodé návody a zdroje, ktoré považujú členovia za zaujímavé a dôležité pri riešení projektu. Odporúča sa uviesť v správe aj krátky opis.
- **document-activity** - Informácie o aktivite na zdieľanom úložisku OneDrive. Správa by mala mať nasledujúci tvar:

[typ dokumentu] názov dokumentu – odkaz na dokument

stručný opis

- **random** - Voľná komunikácia tímu
- **development** - Kanál určený pre komunikáciu problémov pri plnení úloh. Autor príspevku by mal začať diskusiu správou, v ktorej na začiatku uvedie v hranatých zátvorkách číslo úlohy a následne stručne opíše svoj problém.
- **planning** – Diskusia o pridelení jednotlivých úloh a plánovania ich plnenia.

Zdieľanie dokumentov – OneDrive

Na ukladanie zdieľaných dokumentov je využívaná služba OneDrive. Vytvorené dokumenty sa zaraďujú do nasledujúcich priečinkov:

- **Dokumenty k inžinierskemu dielu**
 - **Moduly systému** – technická dokumentácia k jednotlivým projektom
 - **Používateľské príručky** – používateľské príručky k jednotlivým projektom
- **Dokumenty k riadeniu projektu**
 - **Metodiky**
 - **Export evidencie úloh**
- **Návody** – postupy a návody vytvorené členmi tímu – za účelom vyriešenia nejakej úlohy
- **Retrospektívy**
- **Zápisnice zo stretnutí**

Ukladajú sa vo formáte .pdf. Názov v tvare *zápisnica_čX_DD-MM-YY*.

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

Metodika dokumentácie

Tím č.20 – RavensTeam

Účel metodiky

Pri práci na rozsiahlejšom projekte treba počítať s väčším počtom ľudí, ktorí do neho budú môcť určitým spôsobom zasahovať. Pre vývoj softvéru to znamená, že každý programátor má prístup k zdrojovému kódu a môže ho upravovať. Ak by každý programátor pracoval na tej istej verzii zdrojového kódu, hrozilo by, že svojimi zmenami naruší funkcionality implementované iným vývojárom a tým napr. znemožní prácu celému tímu. Z tohto dôvodu treba zabezpečiť, aby mal každý developer v reálnom čase svoju verziu kódu, vďaka čomu budú môcť pracovať a vykonávať zmeny nezávisle. Táto metodika pojednáva o použitých systémoch, princípoch a pravidlách verziovania pri vývoji nášho softvéru.

Verziovací systém - Git

Git je voľne dostupný, distribuovaný systém pre správu verzií nie len zdrojových kódov, ale rôznych typov dokumentov a súborov, plne kompatibilný s TFS.

Link: <https://git-scm.com/>

Slovník pojmov

Branch – vetva projektu, reprezentuje jeho určitú verziu

Merge – spojenie dvoch vetiev do jednej

Push – príkaz pre presunutie zmien z jednej vetvy do inej (remote)

Pull – príkaz pre získanie vetvy a jej obsahu

Checkout – príkaz na presúvanie sa medzi vetvami

Princíp – GitFlow Workflow

Tiež nazývaný „Branching model“, je súbor odporúčaných pravidiel verziovania pri vývoji softvéru. Jeho základom sú dve vetvy, podľa konvencií zvyčajne s názvom „master“ a „develop“. Vetva master reprezentuje hlavný zdrojový kód softvéru a slúži na uchovávanie údajov o „releases“ resp. o zmenách vykonávaných v hlavnom zdrojovom kóde. Develop je integračná vetva, prostredníctvom ktorej sa vykonávajú zmeny v systéme. Z nej si každý vývojár pri implementácii novej funkcionality vytvorí novú vetvu, na ktorej pracuje a tú potom merge-ne naspäť do develop-u. Odtiaľto sú neskôr zmeny push-nuté do vetvy master.

Postup a základné príkazy

Pre prácu s Git-om existuje niekoľko voľne dostupných nástrojov, ktoré poskytujú buď grafické používateľské rozhranie, alebo základné konzolové rozhranie. Vo všeobecnosti je však ich fungovanie založené na rovnakom princípe.

Každý developer zodpovedný za svoju user-story si vytvorí novú vetvu z vetvy dev. Na tejto vetve budú vykonávané zmeny spojené s task-ami pre danú user-story.

1. Presunieme sa do vetvy dev
 - `git checkout dev`
2. Uistíme sa, že pracujeme na aktuálnej verzii branch-e dev
 - `git pull`
3. Developer zodpovedný za user-story vytvorí novú vetvu pre túto user-story
 - `git branch <BranchNameByUserStory>`
4. Presunieme sa do danej vetvy
 - `git checkout <BranchNameByUserStory>`
5. Po dokončení práce na task-u sa požadované zmeny pridajú na commit-nutie
 - `git add <FilesToBeCommitted>`
6. Commit-nutie zmien do vetvy
 - `git commit -m "<CommitMessage>"`
7. Presunieme sa do vetvy dev
 - `git checkout dev`
8. Uistíme sa, že máme aktuálnu verziu
 - `git pull origin dev`
9. Merge zmien z user-story vetvy do dev
 - `git merge <BranchNameByUserStory>`

Pravidlá

Reprezentujú pravidlá pre pomenovávanie jednotlivých vetiev a commit správ kvôli prehľadnosti a konzistentnosti práce počas vývoja softvéru. Zároveň vďaka nim dokážeme ľahšie odhaliť prípadné zlyhania systému a tým pádom ich aj rýchlejšie riešiť.

Vytváranie novej branche

- Pre novú **UserStory**:
 - **US-<UserStoryNumber>/<Descriptive_Text>**
 - **US-4507/Spojenie_s_databazou**

Commit message

- Pre **Task**:
 - [**<ProjectName>**] **Task <TaskNumber>: <Descriptive Text>**
 - [**VsExtension**] **Task 5005: Nacitanie pouzivatelov z databazy**
- Pre **Fix** ku konkrétnemu task-u:
 - [**<ProjectName>**] **Fix <TaskNumber>: <Descriptive Text>**
 - [**VsExtension**] **Fix 5005: Oprava SQL query**
- Pre **Fix** generálne:
 - [**<ProjectName>**] **Fix: <Descriptive Text>**
 - [**VsExtension**] **Fix: Zmena textu v menu**

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

Metodika konvencií písania zdrojového kódu

Tím č.20 – RavensTeam

1. Účel a dedikácia metodiky

V projekte, ktorý je vyvíjaný agilnou metódou (tímom ľudí kde dochádza k častým zmenám v kóde) je potrebné mať presne definované pravidlá písania zdrojového kódu tak, aby bol ľahko pochopiteľný a modifikovateľný pre členov tímu, ako aj pre ostatných vývojárov. Náš projekt je navyše súčasťou väčšieho systému, na ktorého vývoji budú pravdepodobne pracovať aj iné tímy programátorov v budúcnosti.

Táto metodika definuje konvencie písania zdrojového kódu jednotlivu pre programovacie jazyky, ktoré pri vývoji používame. Jej účelom je zachovať udržateľnosť zdrojových kódov vytvorených v rámci projektu. Slúži pre súčasných aj budúcich vývojárov systému DevACTs, ktorého je projekt CodeCrutches súčasťou.

Dodržiavanie tejto metodiky je overované v rámci procesu posudzovania kódu.

2. Všeobecné zásady pri písaní zdrojového kódu

Pri vytváraní zdrojového kódu je nutné dodržiavať nasledujúce pravidlá, bez ohľadu na programovací jazyk, v ktorom je kód vytváraný:

- Po každej úprave kódu je potrebné vykonať jeho formátovanie:
 - Visual Studio IDE:

| |
|------|
| CTRL |
|------|

 +

| |
|---|
| K |
|---|

 +

| |
|---|
| D |
|---|
 - Eclipse IDE:

| |
|------|
| CTRL |
|------|

 +

| |
|-------|
| Shift |
|-------|

 +

| |
|---|
| F |
|---|
- Používať komentáre
 - Prednostne v syntaxe určenej daným programovacím jazykom pre automatické generovanie dokumentácie
- Výstižné pomenovávanie premenných a elementov
 - Nepoužívať jednopísmenové a nič nevypovedajúce názvy premenných s výnimkou počítadiel
- **Ak nie je v metodike definované inak, používajú sa zaužívané konvencie pre jednotlivé programovacie jazyky:**
 - C# / .NET: <https://msdn.microsoft.com/en-us/library/ff926074.aspx>
 - HTML5: http://www.w3schools.com/html/html5_syntax.asp
 - JavaScript: http://www.w3schools.com/js/js_conventions.asp
 - CSS: <http://cssguidelin.es>

3. Zásady písania zdrojového kódu pre jednotlivé programovacie jazyky

Nakoľko zaužívané konvencie písania zdrojových kódov sa líšia v závislosti od programovacích jazykov, táto kapitola popisuje základné zásady písania zdrojového kódu pre jednotlivé programovacie jazyky využívané pri vývoji projektu CodeCrutches.

3.1. C# / .NET

3.1.1. Model – View – Controller

- **Model**
 - Databázový model
 - Názov: rovnaký ako pomenovanie v databáze v singulári
 - Umiestnenie: ~/Models

- ViewModel
 - Názov: končiaci „ViewModel“ príp. „ViewModels“
 - Umiestnenie: ~/Models
- **Controller**
 - Názov: končiaci slovom „Controller“
 - Umiestnenie: ~/Controllers
- **View**
 - Názov: akcia kontroléru, ktorá daný View renderuje
 - Umiestnenie: ~/Views/názov_kontroléra

3.1.2. Komentáre

Používať automaticky generovanú štruktúru komentárov, ktorá sa zobrazí po troch stlačeniach lomky („///<“).

- **Triedy**

Priamo nad začiatkom definície triedy:

```
/// <summary>
/// popis triedy
/// </summary>
public class IndexViewModel
```

- **Metódy**

Priamo nad začiatkom definície metódy:

```
/// <summary>
/// popis metódy
/// </summary>
/// <param name="tag">popis parametra s názvom „tag“</param>
/// <returns>typ_návratovej_hodnoty : popis</returns>
public static int cislo (int a)
```

- **Ostatné časti kódu**

Komentovať všetky časti kódu, ktoré môžu byť nejasné:

- viacriadkovým komentárom nad danou časťou kódu:

```
/* viacriadkový
 * komentár
 * */
Body body = new Body();
```

- jednoriadkovým komentárom priamo nad daným riadkom kódu:

```
// jednoriadkový komentár
Body body = new Body();
```

3.1.3. Názvy

- používať veľké začiatkové písmeno pri:
 - názvoch tried
 - názvoch metód
- používať malé začiatkové písmeno pri:
 - názvoch premenných
- používať anglické názvy

- používať celoslovné názvy vystihujúce podstatu premennej pri dvoj- a menej slovných názvoch
- používať skratky pri troj- a viacslovných názvoch premenných, kde slovo je dlhšie ako 8 znakov:

```
// nesprávne
UserRegistrationForm userRegistrationForm;

// správne
UserRegistrationForm userRegForm;
```

- jednotlivé slová názvu premennej neoddeľovať podčiarkovníkom, ale veľkým písmenom
- názvy tried podstatným menom
- názvy rozhraní začínať písmenom „I“ (napr. `public interface IProject`)
- názvy súborov definujúcich triedu rovnaké ako názov triedy, ktorú definujú

3.1.4. Typy premenných

- používať preddefinované typy premenných namiesto systémových (`string` namiesto `String`, `bool` namiesto `Boolean` a pod.)

3.1.5. Formátovanie

- pri viacriadkovej časti kódu ohraničenej zátvorkami tieto zátvorky vertikálne zarovnávať:

```
// správne
public static TaskTagHandler Instance
{
    get
    {
        if (_instance == null)
        {
            _instance = new TaskTagHandler();
        }
        return _instance;
    }
}
```

3.2. HTML5

3.2.1. Komentáre

- pri písaní komentárov brať ohľad na to, že sú viditeľné pre používateľa aplikácie
- používať len na miestach, kde je to potrebné, napríklad pri označovaní začiatku a konca elementu:

```
<!--#main start-->
<div id="main">
    ...veľa rôznych elementov...
</div>
<!--#main end-->
```

3.2.2. Názvy

- používať výhradne len názvy a atribúty elementov definované v HTML5 (case-sensitive)
- pokiaľ sa v projekte používa Bootstrap Framework:
 - využívať grid systém Bootstrapu pri tvorení layoutu http://www.w3schools.com/bootstrap/bootstrap_grid_system.asp
 - využívať triedy, ktoré Bootstrap definuje pre jednotlivé elementy:

```

<!--správne-->
<table class="table">
  ...
</table>

<!--nesprávne-->
<table>
  ...
</table>

```

- používať triedy slúžiace na zvýraznenie elementu (`danger`, `warning`,...)
- nerobiť od nuly niečo, čo už je vytvorené v rámci Bootstrapu (panely, carousel, ikony, modálne okná a pod.)
 - kompletný tutoriál dostupný na <http://www.w3schools.com/bootstrap/default.asp>
- názvy tried a identifikátorov s malými písmenami, oddelené pomlčkou:

```

<!--správne-->
<i class="code-crutches" />

<!--nesprávne-->
<i class="codeCrutches" />

```

3.2.3. Syntax

- dodržiavať syntax definovanú HTML5
- výsledný kód validovať prostredníctvom nástroja:
 - <https://validator.w3.org/>

3.3. JavaScript

Nakoľko je v projekte CodeCrutches momentálne využívaný jazyk JavaScript len v malom rozsahu, definujeme len niektoré základné pravidlá pri písaní kódu v tomto jazyku. Podrobnejšie konvencie používané pre JavaScript sú dostupné na adrese http://www.w3schools.com/js/js_conventions.asp.

- pri viacslovných názvoch premenných jednotlivé slová neoddeľovať, ale použiť veľké začiatkové písmeno druhého a ďalšieho slova (`codeCrutches` miesto `code-crutches`)
- ak daný skript nepotrebuje byť vykonávaný skôr ako je načítaný celý dokument (event `Document.ready()`), jeho referenciu pridať až na konci HTML dokumentu priamo nad tag `</body>`
- pri zostavovaní verejnej verzie projektu („*public release*“) vykonať minifikáciu kódu:
 - napr. prostredníctvom nástroja <https://jscompress.com>

3.4. CSS

Táto časť definuje základné pravidlá čisto pre jazyk CSS. Pravidlá pre pomenovávanie tried a identifikátorov HTML elementov popisuje časť [3.2 HTML5](#) tohto dokumentu. Podrobné konvencie zaužívané pre CSS sú dostupné na stránke <http://cssguidelin.es>.

- dodržiavať organizáciu kódu, držať definície patriace k jednému prvku (príp. skupine prvkov) pri sebe
- používať čo možno najpresnejšie selektory:

```
/*nesprávne*/
.menu li {
...
}
/*správne*/
ul.menu>li {
...
}
```

- vyhýbať sa používaniu `!important`, pokiaľ to nie je nutné
- oddeľovať definície patriace k rozdielnym skupinám prvkov (príp. k iným stránkam) prostredníctvom komentárov:

```
/*header*/
div#header {...}
div#header ul.menu {...}
div#header ul.menu>li>a {...}

/*content*/
div.banner {...}
div.sidebar {...}
```

- pri zostavovaní verejnej verzie projektu („*public release*“) vykonať minifikáciu kódu:
 - napr. prostredníctvom nástroja <https://cssminifier.com>

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Ilkovičova 2, 842 16 Bratislava 4

Zápis zo stretnutia #2

Tím č.20 - RavensTeam

Dátum: 6.10.2016 13:00

Miesto konania: FIIT STU 4.26

Vypracoval: Ondrej Čičkán

Účastníci stretnutia:

Ing. Karol Rástočný, PhD.

Ondrej Čičkán

Dekrét Šimon

Javorník Dušan

Jom Dušan

Laco Miroslav

Vrban Anton Ján

Priebeh stretnutia

1. Zoradenie epikov podľa priorít
2. Rozbitie prvého epiku na feature a user stories
3. Priradenie story pointov k jednotlivým user stories
4. Identifikácie úloh pri jednotlivých user stories
5. Štruktúra InformationTag a práca s ITM
6. Rozdelenie jednotlivých úloh

Poznámky zo stretnutia

Inštalácia ITM: Code -> ITMaintenance -> ITMaintenance.Driver

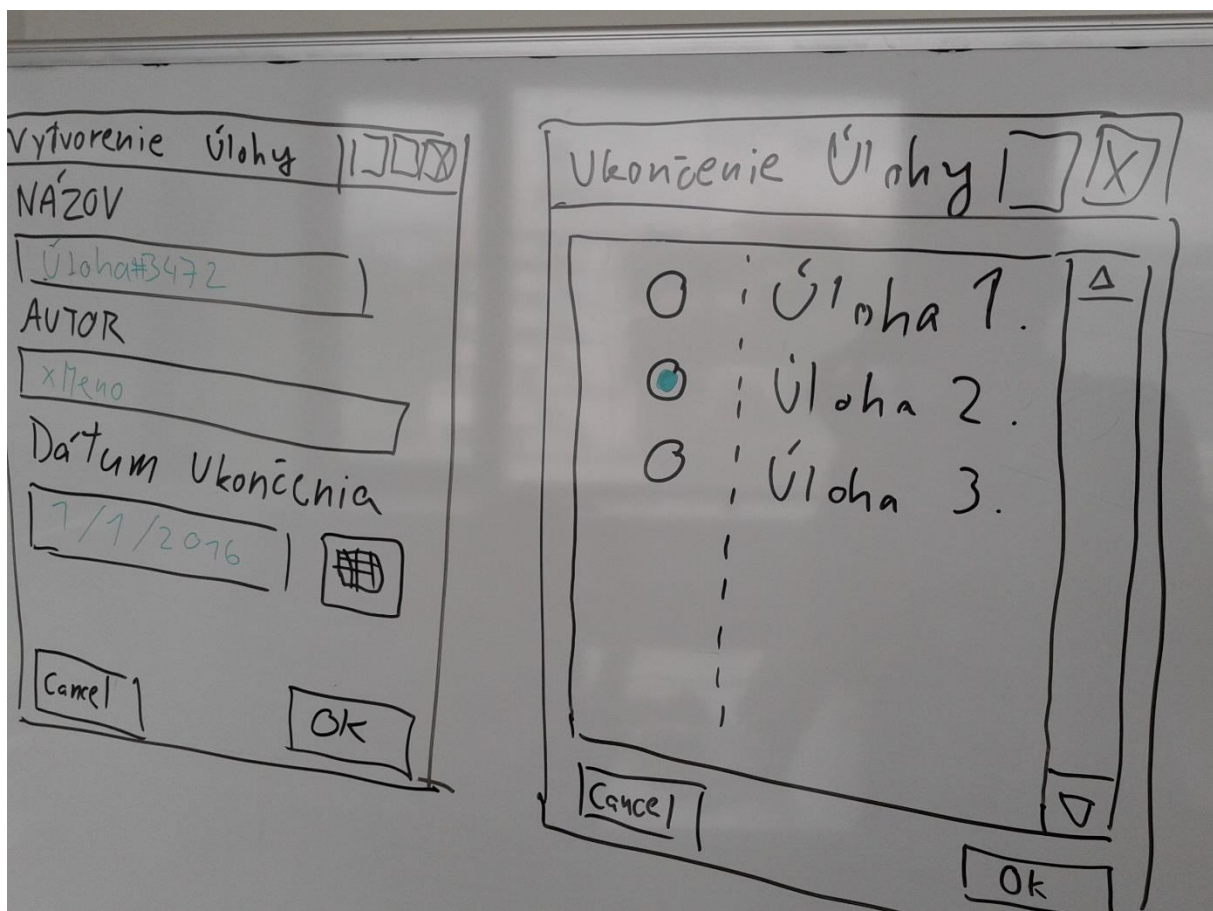
Štruktúra InformationTag –

Target – môže byť solution ID z Visual Studio

Author – vytiahnuť z registra

Nainštalovať plugin do VS:

http://devacts.fiit.stuba.sk/documents/uaca/UACAInstaller_1.0.19_64bit.zip



Obrázok 1 Vzhľad formulárov pri vytvorení a dokončení úlohy

Evidencia úloh

| ID | Title | Work Item Type | Story Points | State | Assigned To | Description |
|----------------------|---|----------------|--------------|-------|---------------------|--|
| 4148 | Priradenie úlohy k aktuálnej práci | Epic | | New | | Ja chcem mať možnosť priradiť úlohu k zmenám aby som vedel po zložitejších úpravách spraviť viacero oddelených commitov, podľa jednotlivých vykonávaných úloh. |
| 4333 | Určenie začiatku novej práce | Feature | | New | | Programátor chcem povedať, že som začal pracovať na novej úlohe |
| 4339 | Zaevidovanie začatia novej úlohy | User Story | 3 | New | Bc. Ondrej Cickan | Ako programátor chcem mať možnosť oznámiť, že som začal pracovať na novej úlohe, lebo chcem vedieť čo som robil a informovať ostatných, že som na tom začal robiť. Budem mať tlačidlo v menu, kde zadám názov úlohy. Začiatok úlohy sa uloží ako značka v ITM. |
| 4341 | Vytvorenie projektu VS pluginu | Task | | New | Bc. Ondrej Cickan | |
| 4342 | Zaregistrovanie položky menu CodeCrutches vo VS s podmenu Začni úlohu | Task | | New | Bc. Ondrej Cickan | |
| 4343 | Formulár pre zadanie opisu úlohy | Task | | New | Bc. Anton Jan Vrban | Text Field OK, Cancel |
| 4344 | Odoslanie úlohy na ITM | Task | | New | Bc. Dusan Jom | |
| 4352 | Spojenie identifikácie používateľa s DevACTs | User Story | 1 | New | Bc. Anton Jan Vrban | |
| 4353 | Vyčítanie hodnoty z registra, kde je používateľ | Task | | New | Bc. Anton Jan Vrban | HKCU\SOFTWARE\JavaSoft\Prefs\com\gratex\perconik\useractivity\app Reg. Name: user/Name |
| 4334 | Určenie konca práce | Feature | | New | | Ako programátor chcem mať možnosť určiť koniec práce aby som vedel koľko som toho na úlohe urobil |

| | | | | | | |
|----------------------|--|------------|---|-----|---------------------|--|
| 4337 | Ukončenie práce na úlohe | User Story | 3 | New | Bc. Dusan Jom | Ako programátor, chcem povedať, že som práve dorobil aktuálne rozpracovanú úlohu aby som upozornil kolegov, že táto úloha je splnená. |
| 4346 | Položka v menu na ukončenie úlohy | Task | | New | Bc. Dusan Jom | |
| 4345 | Načítanie zoznamu prebiehajúcich úloh | Task | | New | Bc. Simon Dekret | Z ITM sa vytiahne zoznam úloh aktuálneho programátora, ktoré sú aktívne |
| 4347 | Formulár so zoznamom úloh | Task | | New | Bc. Anton Jan Vrban | Radiobuttons pri úlohách OK, Cancel |
| 4348 | Aktualizácia stavu úlohy v ITM | Task | | New | Bc. Dusan Jom | |
| 4338 | Určenie predpokladaného ukončenia práce na úlohe | User Story | 1 | New | Bc. Simon Dekret | Ako programátor chcem mať možnosť povedať, kedy asi bude rozrobená úloha ukončená, aby vedeli plánovať ďalšie pokračovanie naväzujúcich prác |
| 4349 | Rozšírenie Objektu ITM o predpokladaný dátum ukončenia úlohy | Task | | New | Bc. Dusan Javornik | presnosť stačí na deň |
| 4351 | Rozšíriť formulár o začiatku úlohy o field pre dátum predpokladaného ukončenia úlohy | Task | | New | Bc. Simon Dekret | |
| 4335 | Spätná úprava úlohy aktuálnej práce | Feature | | New | | Ako programátor... |
| 4350 | Zmena predpokladaného ukončenia úlohy | User Story | 2 | New | Bc. Miroslav Laco | |
| 4354 | Úprava formuláru na ukončenie úlohy pre editáciu dátumu ukončenia | Task | | New | Bc. Miroslav Laco | |

| | | | | | | |
|----------------------|---|---------|--|-----|-------------------|---|
| 4355 | Vytvorenie formuláru(popupu) pre zmenu dátumu | Task | | New | Bc. Miroslav Laco | |
| 4356 | Vytvorenie formuláru pre ukončenie úlohy | Task | | New | Bc. Miroslav Laco | |
| 4336 | Prerušenie aktuálnej práce | Feature | | New | | |
| 4145 | Zoznam zmien elementu | Epic | | New | | Kto, čo a na akej ulohe |
| 4149 | Zoznam úloh pre ktoré bol element menený | Feature | | New | | |
| 4147 | Posúdenie kódu z IDE (rýchle nejakej jednoty) | Epic | | New | | |
| 4340 | Poznačenie budúcej práce | Feature | | New | | AKo programátoch chcem mať možnosť označiť, že tu bude treba niečo ďalej spraviť. |
| 4332 | Metriky zdrojového kódu | Epic | | New | | |
| 4146 | Pachy v kóde | Epic | | New | | |
| 4331 | Štatistika práce programátora | Epic | | New | | |
| 4144 | Konzistencia kódu | Epic | | New | | Kódovacie pravidlá |
| 4151 | Automatická organizácia triedy | Epic | | New | | Usporiadanie prvkov podľa kovencie |
| 4150 | Zobrazenie závislostí v kóde | Epic | | New | | |

Zápis zo stretnutia č. 3

Dátum a miesto konania

13.10.2016 - FIIT STU 4.26

Prítomní

- Ing. Karol Rástočný, PhD.
- Ondrej Čičkán
- Šimon Dekrét
- Dušan Javorník
- Dušan Jom
- Miroslav Laco
- Anton Ján Vrban

Priebeh stretnutia

- Vyhodnotenie práce za predošlý týždeň
- Rozdelenie rolí v tíme
- Prekonzultovanie nejasností ohľadom úloh

Poznámky zo stretnutia

- Technická dokumentácia sa skladá z rozhrania, konfigurácie a dátového modelu
- Manažovanie komunikácie má na starosti Ondrej
- Code review pomocou pull requests (Dušan Javorník)
- Verzie zatiaľ robíme branching per user story (Dušan Jom)
- Unit-testing (zodpovedný Miro)
- Kódovacie štandardy (zodpovedný Anton)

Úlohy do ďalšieho stretnutia

- Konfigurácia servera – presmerovanie na našu stránku
- Práca na úlohach
- Pripraviť metodiky (1. verzia)

Plnenie úloh za týždeň

| | | | |
|----------------------|---|--------|---------------------|
| 4337 | Ukončenie práce na úlohe | New | Bc. Dusan Jom |
| 4346 | Položka v menu na ukončenie úlohy | New | Bc. Dusan Jom |
| 4345 | Načítanie zoznamu prebiehajúcich úloh | Active | Bc. Simon Dekret |
| 4347 | Formulár so zoznamom úloh | New | Bc. Anton Jan Vrban |
| 4348 | Aktualizácia stavu úlohy v ITM | New | Bc. Dusan Jom |
| 4352 | Spojenie identifikácie používateľa s DevACTs | Active | Bc. Anton Jan Vrban |
| 4353 | Vyčítanie hodnoty z registra, kde je používateľ | Closed | Bc. Anton Jan Vrban |
| 4339 | Zaevidovanie začatia novej úlohy | Active | Bc. Ondrej Cickan |
| 4341 | Vytvorenie projektu VS pluginu | Closed | Bc. Ondrej Cickan |
| 4342 | Zaregistrovanie položky menu CodeCrutches vo VS s podmenu Začni úlohu | Closed | Bc. Ondrej Cickan |
| 4343 | Formulár pre zadanie opisu úlohy | New | Bc. Anton Jan Vrban |
| 4344 | Odoslanie ulohy na ITM | New | Bc. Dusan Jom |
| 4338 | Určenie predpokladaného ukončenia práce na úlohe | New | Bc. Simon Dekret |

| | | | |
|----------------------|--|-----|--------------------|
| 4349 | Rozšírenie Objektu ITM o predpokladaný dátum ukončenia úlohy | New | Bc. Dusan Javornik |
| 4351 | Rozšíriť formulár o začiatku úlohy o field pre dátum predpokladaného ukončenia úlohy | New | Bc. Simon Dekret |
| 4350 | Zmena predpokladaného ukončenia úlohy | New | Bc. Miroslav Laco |
| 4354 | Úprava formuláru na ukončenie úlohy pre editáciu dátumu ukončenia | New | Bc. Miroslav Laco |
| 4355 | Vytvorenie formuláru(popupu) pre zmenu dátumu | New | Bc. Miroslav Laco |
| 4356 | Vytvorenie formuláru pre ukončenie úlohy | New | Bc. Miroslav Laco |

Zápisnica zo stretnutia č. 4

Dátum a miesto konania

20.10.2016 - FIIT STU 4.26

Prítomní

- Ing. Karol Rástočný, PhD.
- Ondrej Čičkán
- Šimon Dekrét
- Dušan Javorník
- Dušan Jom
- Miroslav Laco
- Anton Ján Vrban

Priebeh stretnutia

- Prezentácia stavu jednotlivých user-stories
- Sumarizácia prvého šprintu
- Retrospektíva
- Nápady na zlepšenie
- Vytváranie nových úloh
- Priradenie úloh

Poznámky zo stretnutia

Informácie pre Tag do ITM:

Meno_ulohy

Stav: Active, Paused, Done

Predpokladaný dátum ukončenia

- Pri vytvorení vždy v stave active

Manažment úloh

zobrazované informácie

Task ID - Meno - Stav – Dátum zač. – Dátum ukončenia – strávený čas

Configurácia servera:

Nuget Central Services -> knižnica, pre prístup do konfigurácie

<https://devacts.fiit.stuba.sk:8443/DevACTs/Config>

Proxy pre ITM driver -> vytvorenie pomocou ITM drivera , ale s endpointom získaného z Central Services

Vytvorenie usera pre konfiguráciu

User config DevActs Cord Core CodeReview ITGenerator ITMaintance Tagging UserActivity

Set config user

New name of service account New password of service account

serviceUser ***** Save

- Nemôžeme použiť hneď v source code pre plugin (security problém)

Vytvorenie nuget package v DevACTS

- DevACTs -> AdministrationPortal -> MongoMigration
- Príklady na nuget package Core -> Core.Rcs -> *.nunspec => skopírovať a prepísať potrebné polia -> nuget.exe pack() => urobí nuget.pck – nuget.exe publish - publishne na server
- Nuget.org -> documentation -> create package -> publish package
- <https://devacts.fiit.stuba.sk:8443/nugetserver>

Úlohy do ďalšieho stretnutia

- Nainštalovať TeamViewer

Spísať metodiky:

Miroslav Laco – Testovanie

Dušan Jom – Verziovanie & Git

Dušan Javorník – Code Review

Anton Ján Vrban – Štandardy na písanie kódu

Export úloh pre aktuálny šprint

| Title | State | Priority | Assigned To | Completed Work | Remaining Work | Description |
|--|-------|----------|-----------------|----------------|----------------|--|
| Prerusenie prace na aktualne rozrobenej ulohy | New | 2 | Simon Dekret | | 6 | |
| Rozsirit objekt ITM o stav ulohy | New | 2 | Simon Dekret | 0 | 2 | stav ulohy: active paused done |
| Rozsirit formular pre manipulaciu s ulohami o prerusenie ulohy | New | 2 | Simon Dekret | 0 | 2 | Zahrna implementaciu obsluznej metody |
| Rozsirit formular manipulacie s ulohami o stavy uloh | New | 2 | Simon Dekret | 0 | 2 | Zelena sipka pri active, paused ikona pri paused. Prva uloha je aktivna uloha |
| Opatovne zahajenie prace na ulohy | New | 2 | Ondrej Cickan | | 2 | |
| Rozsirit formular manipulacie s ulohami o pokracovanie v praci | New | 2 | Ondrej Cickan | 0 | 2 | Moznosti zmeny stavu sa budu ponukat podla aktualneho stavu ulohy (pre pauznutu pokracuj, pre aktivnu pauzni) |
| Zmena ulohy na ktorej sa pracuje | New | 2 | Miroslav Laco | | 3 | Ja ako vyvojar chcem zmenit ulohu na ktorej pracujem, aby som mohol zacat pracovat na inej ulohy. |
| Prepnutie stavu aktivnej ulohy na paused a aktivovanie novej ul... | New | 2 | Miroslav Laco | | 3 | Tyka sa pridavania novej ulohy, aj oznacenia active niektorej z rozpracovanych |
| Zobrazenie stavu uloh | New | 2 | Anton Jan Vrban | | 5 | Ja ako manazer chcem vidiet aj ukoncene, prerusene ulohy a aktivne ulohy a straveny cas. |
| Zalozit webovy projekt | New | 2 | Anton Jan Vrban | | 1 | |
| Nacitat zoznam uloh z ITM | New | 2 | Anton Jan Vrban | | 2 | Zahrna prepojit webovy projekt s ITM |
| Nadizajnovat a impelemntovat view pre model zoznamu uloh | New | 2 | Anton Jan Vrban | | 2 | |
| Sprava uloh vyuziva centralne sluzby | New | 2 | Bc. Dusan Jom | | | Ja ako administrator chcem, aby sa vyuzivali ulohy ulozene v ITM, ktore je nakonfigurovane v centralnych sluzbach |
| Nacitanie konfiguracie endpointu ITM z centralnych sluzieb | New | 2 | | | | Pridat nuget core.centalservices do webového projektu. |
| Vytvorenie proxy pre pripojenie na ITM | New | 2 | | | | Vsade kde sa pouzivalo spojenie pouzivat odteraz toto |
| Nastavenie migracii pre Mongo a webovy projekt | New | 2 | Ondrej Cickan | | 2 | |
| Vytvorenie nugetu pre Mongo migrations v DevACTs projekte | New | 4 | Ondrej Cickan | 0 | 1,5 | https://devacts.fiit.stuba.sk:8443/nugetserver/ |
| Implementovanie vytvoreneho nuget baliku vo webovom projekte | New | 4 | Ondrej Cickan | 0 | 0,5 | |
| Preskumanie moznosti pripojenia na TFS | New | 2 | Dusan Javornik | | 4 | Vystupom analyticka sprava |
| Analyza | New | 2 | Dusan Javornik | | 4 | |

Zápisnica zo stretnutia č. 5

Dátum a miesto konania

27.10.2016 - FIIT STU 4.26

Prítomní

- Ing. Karol Rástočný, PhD.
- Ondrej Čičkán
- Šimon Dekrét
- Dušan Javorník
- Dušan Jom
- Miroslav Laco
- Anton Ján Vrban

Priebeh stretnutia

- Prezentácia stav šprintov a metodík
- Vytvorenie vlastného úložiska na ukladanie informácií o ITM
- Ilustrácia výzoru webového projektu na management
- Práca sa s Mongom
- Rozširovanie backlogu o nové user-storeis a ich ohodnotenie

Poznámky zo stretnutia

Prístup k solution informáciám vo VS príklad v

UserActivity -> VsPackage -> Listeners -> Monitors -> IdePorjectListener - prístup k solution informáciám a eventom

Mongo:

- vytvorenie databazy a kolekcie

Model pre pripojenie

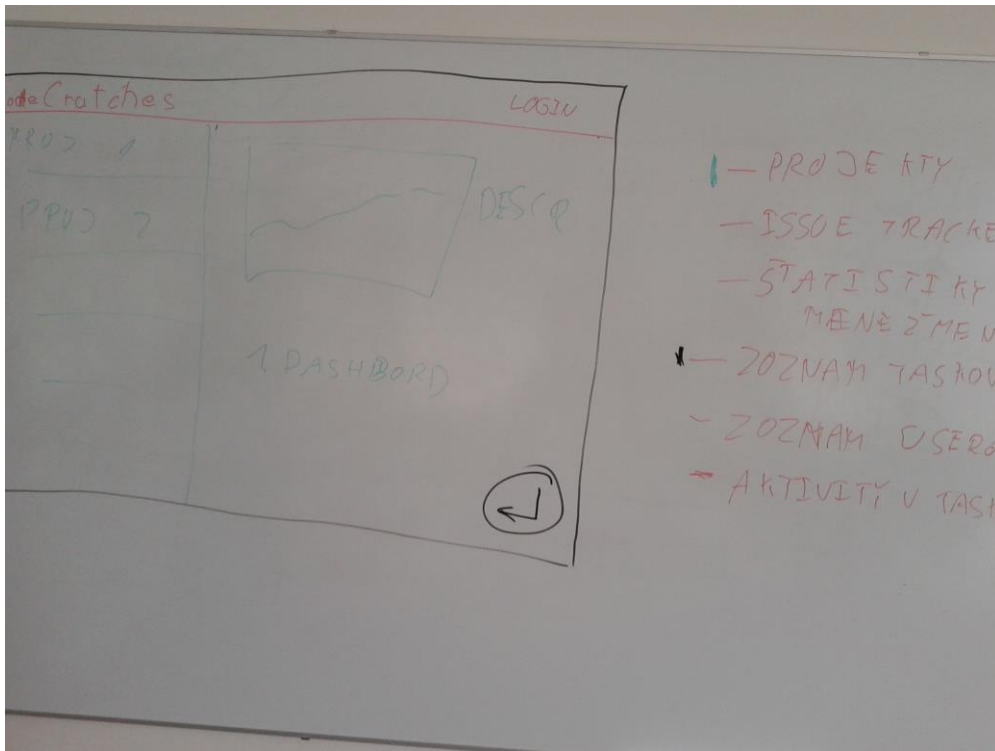
Dátový model AbstractConnection(name, type) -> dedia od neho ďalšie - TFSCConnection ...

Číselník

- pre type connections
- Pre roly

Vytvorenie Projectu -> bude mať usera, ktorý to vytvoril -> a priradenúú rolu

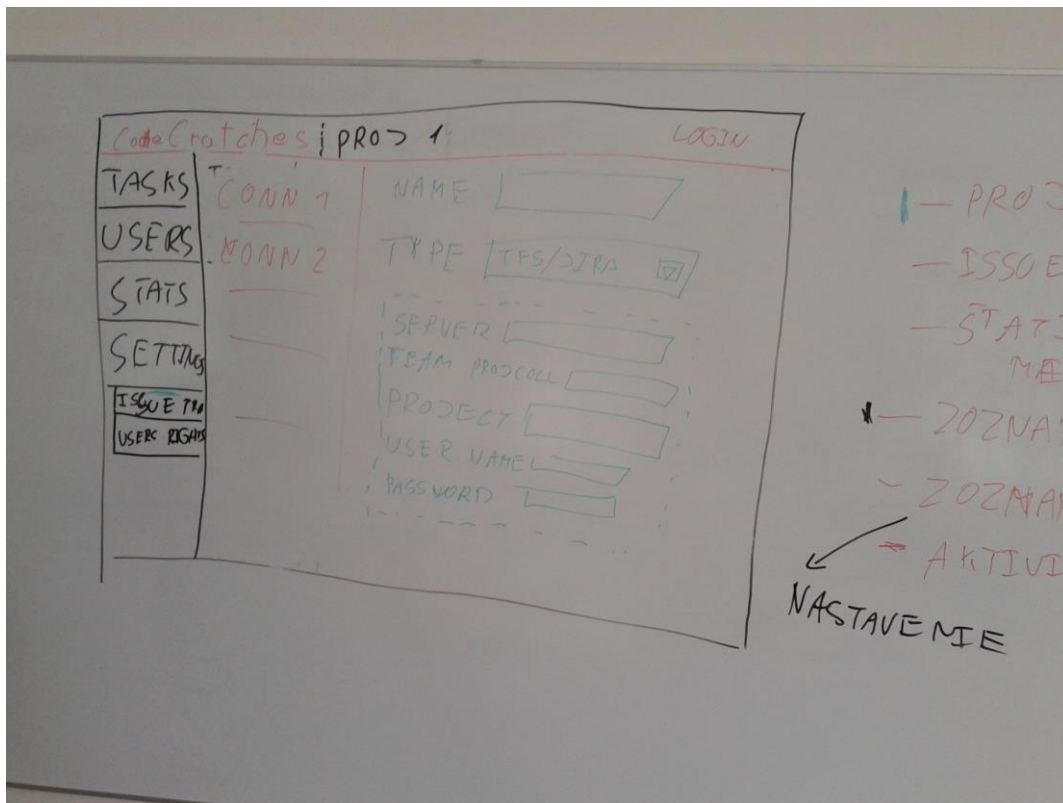
Pri vstupe na do web. Portálu sa musí najprv prihlásiť s DevAct. Bude mať zobrazené len projekty na ktorých je priradený.



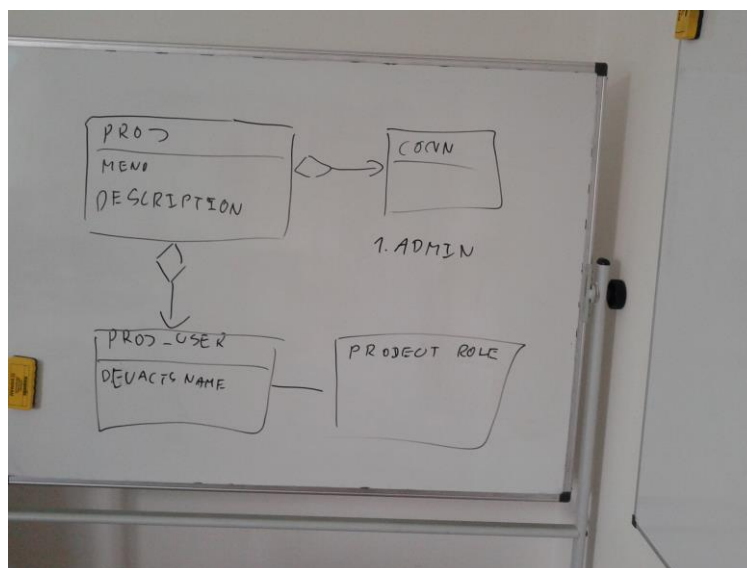
Obrázok 1 Hlavná obrazovka na webovom projekte



Obrázok 2 Obrazovka po otvorení projektu



Obrázok 3 Nastavenie pripojení (TFS/JIRA ..) k ku konkrétnemu projektu



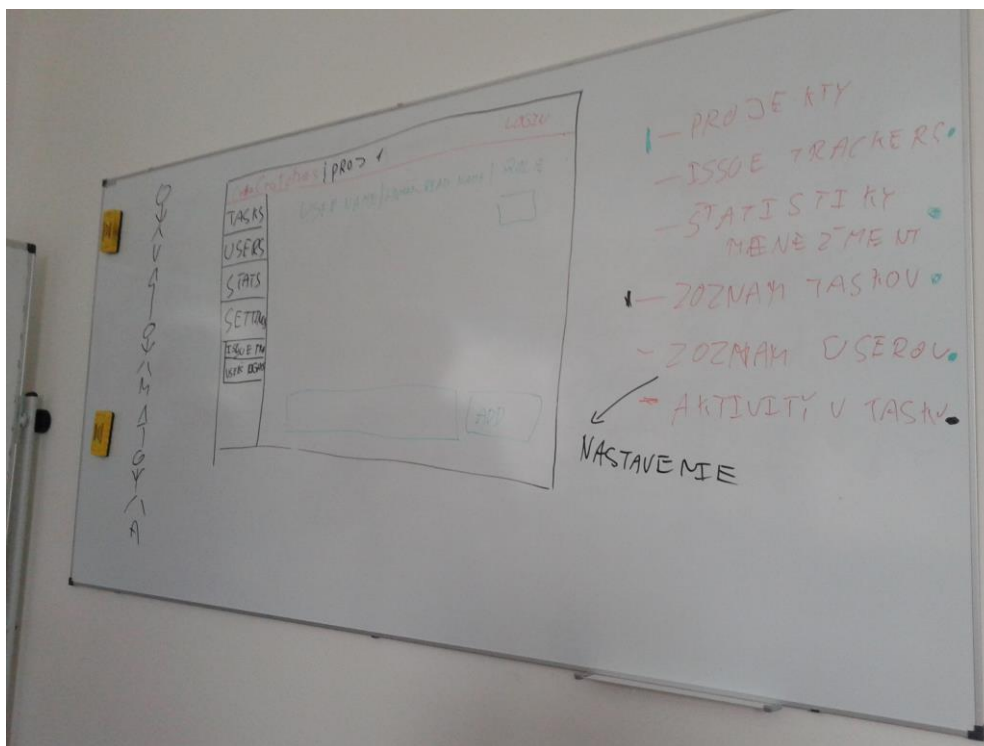
Obrázok 4 Dátový model k projektu

Manažment práv používateľov k projektu

Username | User full name | Role

Pridanie – pole s automatickým dopĺňaním používateľov z DevAct

Role – admin, manažér, vývojár



Obrázok 5 Obrazovka na manažment práv k projektu

Export úloh pre aktuálny šprint

| Title | State | Priority | Assigned To | Completed Work | Remaining Work | Description |
|--|--------|----------|-----------------|----------------|----------------|---|
| Nastavenie migrácii pre Mongo a webovy projekt | Active | 2 | Ondrej Cickan | | | |
| Vytvorenie nugetu pre Mongo migrations v DevACTs projekte | Closed | 4 | Ondrej Cickan | 1 | | https://devacts.fiit.stuba.sk:8443/nugetserver/ |
| Implementovanie vytvoreného nuget balíku vo webovom projekte | Closed | 4 | Ondrej Cickan | 0,25 | | |
| Prerušenie práce na aktuálne rozrobenej ulohe | Active | 2 | Simon Dekret | | 4 | |
| Rozšíriť objekt ITM o stav ulohy | Closed | 2 | Simon Dekret | 1,5 | | stav ulohy: active paused done |
| Rozšíriť formulár pre manipuláciu s ulohami o prerušenie ulohy | Active | 2 | Simon Dekret | 0 | 2 | Zahrna implementáciu obslužnej metódy |
| Rozšíriť formulár manipulácie s ulohami o stavy uloh | New | 2 | Simon Dekret | 0 | 2 | Zelená šípka pri active, paused ikona pri paused. Prvá uloha je aktívna uloha |
| Opatovne zahajenie práce na ulohe | New | 2 | Ondrej Cickan | | 2 | |
| Rozšíriť formulár manipulácie s ulohami o pokračovanie v práci | New | 2 | Ondrej Cickan | 0 | 2 | Možnosti zmeny stavu sa budú ponúkať podľa aktuálneho stavu ulohy (pre pauzovanú pokračuj, pre aktívnu pauzni) |
| Zmena ulohy na ktorej sa pracuje | New | 2 | Miroslav Laco | | 3 | Ja ako vyvojár chcem zmeniť ulohu na ktorej pracujem, aby som mohol začať pracovať na inej ulohe. |
| Prepnutie stavu aktívnej ulohy na paused a aktivovanie novej ul... | New | 2 | Miroslav Laco | | 3 | Týka sa pridávania novej ulohy, aj označenia active niektorých z rozpracovaných |
| Zobrazenie stavu uloh | Active | 2 | Anton Jan Vrban | | 5 | Ja ako manažer chcem vidieť aj ukončené, prerušené ulohy a aktívne ulohy a strávený čas. |
| Založiť webový projekt | Active | 2 | Anton Jan Vrban | | 1 | |
| Nacítanie zoznamu uloh z ITM | New | 2 | Anton Jan Vrban | | 2 | Zahrna prepojiť webový projekt s ITM |
| Nadizajnovat a implemntovat view pre model zoznamu uloh | New | 2 | Anton Jan Vrban | | 2 | |
| Preskúmanie možnosti pripojenia na TFS | Active | 2 | Dusan Javornik | | 4 | Vystupom analytická správa |
| Analýza | New | 2 | Dusan Javornik | | 4 | |
| Správa uloh využívajúca centrálnu službu | New | 2 | Dusan Jom | | 8 | Ja ako administrátor chcem, aby sa využívali ulohy uložené v ITM, ktoré je nakonfigurované v centrálnych službách |
| Nacítanie konfigurácie endpointu ITM z centrálnych služieb | New | 2 | Dusan Jom | 0 | 4 | Pridať nuget core.centraiservices do webového projektu. |
| Vytvorenie proxy pre pripojenie na ITM | New | 2 | Dusan Jom | 0 | 4 | Vsade kde sa používalo spojenie používať odteraz toto |

| Order | Title | State | Story Points | Assigned To | Description |
|-------|--|----------|--------------|-----------------|---|
| 1 | ▸ Ukončenie práce na úlohe | Active | 3 | Dusan Jom | Ako programátor, chcem povedať, že som práve dorobil aktuálne rozpracovanú úlohu aby som upozornil kolegov, že táto úloha je splnená. |
| 2 | ▸ Spojenie identifikácie používateľa s DevACTs | Active | 1 | Anton Jan Vrban | |
| 3 | ▸ Nastavenie migrácii pre Mongo a webový projekt | Active | 1 | Ondrej Cickan | |
| 4 | ▸ Zaevidovanie začatia novej úlohy | Resolved | 3 | Ondrej Cickan | Ako programátor chcem mať možnosť oznámiť, že som začal pracovať na novej úlohe, lebo chcem vedieť čo som robil a informovať ostatných, že som na tom začal ... |
| 5 | ▸ Určenie predpokladaného ukončenia práce na úlohe | Active | 1 | Simon Dekret | Ako programátor chcem mať možnosť povedať, kedy asi bude rozobrená úloha ukončená, aby vedeli plánovať ďalšie pokračovanie naväzujúcich prác |
| 6 | ▸ Zmena predpokladaného ukončenia úlohy | Active | 2 | Miroslav Laco | |
| 7 | ▸ Prerušenie práce na aktuálne rozrobenej úlohe | Active | 2 | Simon Dekret | |
| 8 | ▸ Opatovne zahajenie práce na úlohe | New | 1 | Ondrej Cickan | |
| 9 | ▸ Zmena úlohy na ktorej sa pracuje | New | 1 | Miroslav Laco | Ja ako vyvojár chcem zmeniť úlohu na ktorej pracujem, aby som mohol začať pracovať na inej úlohe. |
| 10 | ▸ Zobrazenie stavu úloh | Active | 3 | Anton Jan Vrban | Ja ako manažer chcem vidieť aj ukončené, prerušené úlohy a aktívne úlohy a strávený čas. |
| 11 | ▸ Preskúmanie možnosti pripojenia na TFS | Active | 1 | Dusan Javornik | Vystupom analytická správa |
| 12 | ▸ Sprava úloh využívajúca centrálnu službu | New | 2 | Dusan Jom | Ja ako administrátor chcem, aby sa využívali úlohy uložené v ITM, ktoré je nakonfigurované v centrálnych službách |
| 13 | ▸ Web CodeCrutches má nastavené používateľské rozhranie | New | 3 | | |
| 14 | ▸ Prihlasovanie pomocou DevACTs účtu | New | 2 | | Ako používateľ DevACTs systému sa chcem vedieť prihlásiť v CodeCrutches, aby som mohol sledovať svoje projekty/úlohy |
| 15 | ▸ Vytvorenie projektu | New | 2 | | Ja ako používateľ CodeCrutches chcem mať možnosť vytvoriť nový projekt aby... |
| 16 | ▸ Pridávanie a nastavenie práv používateľov priradených k projektu | New | 3 | | Ako administrátor projektu chcem mať možnosť pridať používateľa a nastaviť, aké role majú používateľa |
| 17 | ▸ Vyber aktívneho projektu | New | 3 | | Ako vyvojár chcem zadať na akom projekte pracujem, aby som si mohol vybrať z úloh priradených k projektu.<div> </div><div>Neposielat na ITM, ukladať len... |
| 18 | ▸ Nastavenie parovania s TFS | New | 2 | | Ja ako administrátor chcem mať možnosť spárovať služby pre manažment úloh z tfs |
| 19 | ▸ Test connection do TFS | New | | | Ako administrátor napojení projektu na Issue trackers chcem mať otestované zadane pripojenie, aby som vylúčil chybu uloženia zleho pripojenia |

Zápisnica zo stretnutia č. 6

Dátum a miesto konania

3.11.2016 – FIIT STU 1.31a

Prítomní:

- ~~Ing. Karol Rástočný, PhD.~~
- Ondrej Čičkán
- Šimon Dekrét
- Dušan Javorník
- Dušan Jom
- Miroslav Laco
- Anton Ján Vrban

Priebeh stretnutia

- Retrospektíva – zhodnotenie šprintu
- Návrh zmien v projekte
- Pridelovanie user stories z ďalšieho šprintu na ľudí
- Pridávanie nových taskov do backlogu

Poznámky zo stretnutia

- Vytvorenie nového modelu pre **TASK**
 - Prehľadnosť
- Zmeny v pull requestoch
 - Stanovenie pravidiel
- Vytvorenie nových taskov
- Prezentácia a pridanie **UNIT TESTOV**
- Nainštalovať - **VisualStudio Enterprise 2015 with Update 3**

Export úloh pre aktuálny šprint

| Title | State | Story Points | Assigned To |
|--|----------|--------------|-----------------|
| Ukončenie práce na úlohe | Active | 3 | Dusan Jom |
| Zmena predpokladaného ukončenia úlohy | Resolved | 2 | Miroslav Laco |
| Určenie predpokladaného ukončenia práce na úlohe | Resolved | 1 | Simon Dekret |
| Prerušenie práce na aktualne rozrobenej ulohe | Resolved | 2 | Simon Dekret |
| Zaevidovanie začatia novej úlohy | Resolved | 3 | Ondrej Cickan |
| Spojenie identifikácie používateľa s DevACTs | Resolved | 1 | Anton Jan Vrban |
| Nastavenie migrácii pre Mongo a webovy projekt | Resolved | 1 | Ondrej Cickan |
| Opatovne zahajenie práce na ulohe | Resolved | 1 | Ondrej Cickan |
| Zmena ulohy na ktorej sa pracuje | Active | 1 | Miroslav Laco |
| Zobrazenie stavu uloh | Active | 3 | Anton Jan Vrban |
| Preskumanie možnosti pripojenia na TFS | Active | 1 | Dusan Javornik |
| Sprava uloh vyuziva centralne sluzby | New | 2 | Dusan Jom |
| Web CodeCrutches ma nastylované používateľské rozhranie | New | 3 | Anton Jan Vrban |
| Prihlasovanie pomocou DevACTs uctu | New | 2 | Dusan Javornik |
| Vytvorenie projektu | New | 2 | Miroslav Laco |
| Pridavanie a nastavenie prav pouzivatelov priradenych k projektu | New | 3 | Ondrej Cickan |
| Vyber aktivneho projektu | New | 3 | Ondrej Cickan |
| Nastavenie parovania s TFS | New | 2 | |
| Test connection do TFS | New | | |

| Title | State | Story Points | Assigned To |
|---|--------|--------------|-----------------|
| Zobrazenie stavu uloh | Active | 3 | Anton Jan Vrban |
| Preskumanie možnosti pripojenia na TFS | Active | 1 | Dusan Javornik |
| Sprava uloh vyuziva centralne sluzby | New | 2 | Dusan Jom |
| Nacitanie konfiguracie endpointu ITM z centralnych sluzieb | Active | | Dusan Jom |
| Vytvorenie proxy pre pripojenie na ITM | New | | Dusan Jom |
| Web CodeCrutches ma nastylované používateľské rozhranie | New | 3 | Anton Jan Vrban |
| Definovanie stylu pre uvodnu obrazovku | New | | Anton Jan Vrban |
| Definovanie stylu pre menu projektu | New | | Anton Jan Vrban |
| Implementacia polozky tasks | New | | Anton Jan Vrban |
| Prihlasovanie pomocou DevACTs uctu | New | 2 | Dusan Javornik |
| Implementovať využitie autentifikácie a autorizácie z centralnych sluzieb | New | | Dusan Javornik |
| Nastylovať uvodnu obrazovku pred prvym prihlasenim | New | | Dusan Javornik |
| Vytvorenie projektu | New | 2 | Miroslav Laco |
| Vytvorenie formularu pre vytvorenie projektu | New | | Miroslav Laco |
| Vytvorenie zakladu projektovej databazy | New | | Miroslav Laco |
| Pridavanie a nastavenie prav pouzivatelov priradenych k projektu | New | 3 | Ondrej Cickan |
| Vytvorenie formularu vsetkych pouzivatelov projektu | New | | |
| Pridanie formularu pre pridanie pouzivatela | New | | |
| Pridanie zmeny role do formularu | New | | |
| Vyber aktivneho projektu | New | 3 | Ondrej Cickan |
| Dekriptovanie a autorizovanie hesla a mena pouzivatela | New | | |
| Rozsirenie formularu zaciatku novej prace o projekt | New | | |
| Nacitanie zoznamu projektov ku ktorym mam pristup | New | | |
| Refaktoring modelu Extension-u do Visual Studia | New | | Dusan Jom |
| Nastavenie parovania s TFS | New | 2 | |
| Vytvorenie formularu pre pripojenie do tfs | New | | |
| Uloženie údajov do Mongo | New | | |

Zápis zo stretnutia č. 7

Dátum a miesto konania

10.11.2016 - FIIT STU 4.26

Prítomní

- Ing. Karol Rástočný, PhD.
- Ondrej Čičkán
- Šimon Dekrét
- Dušan Javorník
- ~~Dušan Jom~~
- Miroslav Laco
- Anton Ján Vrban

Priebeh stretnutia

- Prezentácia stavu šprintov a ukončenie user stories
- Prezentácia možností komunikácie nášho projektu s TFS serverom
- Diskusia o práci s mongo a migráciami

Poznámky zo stretnutia

- Preformulovať jednotlivé časti v používateľskej príručke (zamerať sa na potrebu používateľa urobiť nejakú úlohu)
- V technickej dokumentácii treba opísať nuget a dopísať z ktorého registra sa načítava meno z User Activity

Úlohy do ďalšieho stretnutia

- Dokončiť tasky z tretieho šprintu

Plnenie úloh za týždeň

| Title | State | Assigned To | Completed Work | Remaining Work | Description |
|---|--------|-----------------|----------------|----------------|---|
| • Sprava uloh vyzivava centrale sluzby | Active | Dusan Jom | 6 | 6 | Ja ako administrator chcem, aby sa vyzivali ulohy ulozene v ITM, ktore je nakonfigurovane v centralnych sluzb |
| • Nacitanie konfiguracie endpointu ITM z centralnych sluzieb | Active | Dusan Jom | 2 | 2 | Pridat nuget core.centraiservices do webového projektu. |
| • Vytvorenie proxy pre pripojenie na ITM | New | Dusan Jom | 0 | 4 | Vsade kde sa pouzivalo spojenie pouzivat odteraz toto |
| • Web CodeCrutches ma nastylovane pouzivateliske rozhranie | Active | Anton Jan Vrban | 2 | 2 | |
| • Definovanie stylu pre uvodnu obrazovku | Closed | Anton Jan Vrban | 2 | 2 | Zoznam projektov vľavo, prazdna plocha pre dashboard a description vpravo, vpravo dole button (vyrazny) pre |
| • Definovanie stylu pre menu projektu | Closed | Anton Jan Vrban | 1 | 1 | Menu. <div>TasksUsersStatsSettingsIssue trackers |
| • Implementacia položky tasks | Closed | Anton Jan Vrban | 1 | 1 | Vyuzit implementaciu z predchadzajuceho sprintu a zobrazit na obrazovke pre Tasks |
| • Implementovanie funkcionalit z pouzivateliskeho rozhrania | Active | Anton Jan Vrban | 3 | 2 | Napr dynamicke menu, breadcrumb a pod |
| • Prihlasovanie pomocou DevACTs uctu | New | Dusan Javornik | 7 | 7 | Ako pouzivatel DevACTs systemu sa chcem vediet prihlasit v CodeCrutches, aby som mohol sledovat svoje pr |
| • Implementovat vyuzitie autentifikacie a autorizacie z centralnych ... | New | Dusan Javornik | 0 | 5 | Vid ine projekty (User activity, Cord) |
| • Nastylovat uvodnu obrazovku pred prvym prihlasenim | New | Dusan Javornik | 0 | 2 | Default login screen |
| • Vytvorenie projektu | Active | Miroslav Laco | 3 | 3 | Ja ako pouzivatel CodeCrutches chcem mat moznost vytvorit nový projekt aby... |
| • Vytvorenie formularu pre vytvorenie projektu | New | Miroslav Laco | 2 | 2 | Meno<div>Description</div><div>Button Save</div> |
| • Vytvorenie zakladu projektovej databazy | Active | Miroslav Laco | 1 | 1 | V Enums potrebujem mat zoznam projektovych roli (zatiaľ admin).<div>Thread </div>proj |
| • Pridavanie a nastavenie prav pouzivatelov priradenych k projektu | New | Ondrej Cickan | 7 | 7 | Ako administrator projektu chcem mat moznost pridať pouzivately a nastavit, ake role maju pouzivately |
| • Vytvorenie formularu vsetkych pouzivatelov projektu | Active | Ondrej Cickan | 2 | 2 | Iba list pouzivatelov, ktorí robia na projekte.<div> </div><div>Nastavenie prav pre role (ak nie som Admin t |
| • Pridanie formularu pre pridanie pouzivately | New | Simon Dekret | 3 | 3 | <div>Nového pouzivately zacnem pisat do boxu a vyhodí mi navrhly (nacitat pouzivately z DevACTs) |
| • Pridanie zmeny role do formularu | New | Simon Dekret | 2 | 2 | On-the-go bez save (AJAX) |

| O... | Title | State | Story Points | Assigned To | Description |
|------|--|----------|--------------|-----------------|---|
| 1 | Ukončenie práce na úlohe | Resolved | 3 | Dusan Jom | Ako programátor, chcem povedať, že som práve dorobil aktuálne rozpracovanú úlohu aby som upozornil kolegov, že táto úloha je splnená. |
| 2 | Nastavenie migrácii pre Mongo a webovy projekt | Resolved | 1 | Ondrej Cickan | |
| 3 | Sprava uloh vyziviva centralne sluzby | Active | 2 | Dusan Jom | Ja ako administrator chcem, aby sa vyzivivali ulohy ulozene v ITM, ktore je nakonfigurovane v centralnych sluzbach |
| 4 | Web CodeCrutches ma nastylované používateľske... | Active | 3 | Anton Jan Vrban | |
| 5 | Prihlasovanie pomocou DevACTs uctu | New | 2 | Dusan Javornik | Ako používateľ DevACTs systému sa chcem vedieť prihlásiť v CodeCrutches, aby som mohol sledovať svoje projekty/úlohy |
| 6 | Vytvorenie projektu | Active | 2 | Miroslav Laco | Ja ako používateľ CodeCrutches chcem mať možnosť vytvoriť nový projekt aby... |
| 7 | Pridavanie a nastavenie prav používateľov prirade... | New | 3 | Ondrej Cickan | Ako administrator projektu chcem mať možnosť pridať používateľa a nastaviť, aké role majú používatelia |
| 8 | Vyber aktívneho projektu | New | 3 | Ondrej Cickan | Ako vývojár chcem zadať na akom projekte pracujem, aby som si mohol vybrať z úloh priradených k projektu.<div> </div>Neodoslať na L... |
| 9 | Nastavenie parovania s TFS | New | 2 | | Ja ako administrator chcem mať možnosť spárovať služby pre manažment úloh z tfs |
| 10 | Test connection do TFS | New | | | Ako administrator napojení projektu na Issue trackers chcem mať otestované zadane pripojenie, aby som vylúčil chybu uloženia zleho pripojenia |
| 11 | Test2 | New | | | testujem jedna radost |

Zápis zo stretnutia č. 8

Dátum a miesto konania

16.11.2016 - FIIT STU 4.26

Prítomní

- Ing. Karol Rástočný, PhD.
- Ondrej Čičkán
- Šimon Dekrét
- Dušan Javorník
- Dušan Jom
- Miroslav Laco
- Anton Ján Vrban

Priebeh stretnutia

- Retrospektíva
- Vyhodnotenie šprintu
- Diskusia o architektúre

Poznámky zo stretnutia

| Člen | Zájmy |
|----------------|---|
| Miro Laco | Modely, db, .NET |
| Dušan Javorník | MVC, Node.js |
| Šimon Dekrét | Java, Rails, SQL, ASP.NET |
| Anton Vrban | ASP.NET, Mobilné apps, nové technológie |
| Dušan Jom | .NET (WPF), MVC, architektúry |
| Ondrej Čičkán | Java, učiť sa nové veci |

Vyberať si úlohy, ktoré jednotlivým členom vyhovujú.

Vytvoriť metodiku na plánovanie a určiť člena, ktorý bude za to zodpovedný. Vytvoriť zásady user stories.

Čo by malo byť v metodikách:

- Proces/zásady
- Roly (kto je za čo zodpovedný)
- Cieľ

Úlohy do ďalšieho stretnutia

- Dokončiť tretí šprint
- Napísať dokumentáciu k inžinierskemu dielu a riadeniu projektu.

Plnenie úloh za týždeň

| ID | Title | State | Assigned To | Completed Work | Remaining Work | Description |
|----------------------|---|--------|----------------|----------------|----------------|---|
| 4706 | Sprava uloh vyuziva centralne sluzby | Active | Dusan Jom | | 6 | Ja ako administrator chcem, aby sa vyuzivali ulohy ulozene v ITM, ktore je nakonfigurovane v centralnych sluzbach |
| 4707 | Nacitanie konfiguracie endpointu ITM z centralnych sluzieb | Active | Dusan Jom | 2 | 2 | Pridat nuget core.centernalservices do webového projektu. |
| 4708 | Vytvorenie proxy pre pripojenie na ITM | New | Dusan Jom | 0 | 4 | Vsade kde sa pouzivalo spojenie pouzivat odteraz toto |
| 4798 | Prihlasovanie pomocou DevACTs uctu | Active | Dusan Javornik | | 5 | Ako pouzivatel DevACTs systemu sa chcem vediet prihlasit v CodeCrutches, aby som mohol sledovat svoje projekty/ulohy |
| 4799 | Implementovat vyuzitie autentifikacie a autorizacie z centralnych sluzieb | Active | Dusan Javornik | 5 | 3 | Vid ine projekty (User activity, Cord) |
| 4800 | Nastylovat uvodnu obrazovku pred prvym prihlasenim | New | Dusan Javornik | 0 | 2 | Default login screen |
| 4795 | Vytvorenie projektu | Active | Miroslav Laco | | 1 | Ja ako pouzivatel CodeCrutches chcem mat moznost vytvorit novy projekt aby... |
| 4796 | Vytvorenie formularu pre vytvorenie projektu | Active | Miroslav Laco | 3.5 | 1 | Meno<div>Description</div><div>Button Save</div> |
| 4797 | Vytvorenie zakladu projektovej databazy | Closed | Miroslav Laco | 4.5 | | V Enums potrebujem mat zoznam projektovych roli (zatiaľ admin).<div>Thready </div><div>project name |

| ID | Title | State | Assigned To | Completed Work | Remaining Work | Description |
|----------------------|--|--------|---------------|----------------|----------------|--|
| | | | | | | >descriptionproject_user devacts nameproject role (Roly pouzivatelov (dedia sa zhora nadol): <div>AdminManazerVyvojar</div><div>Project + project_user ako jeden collection</div></div> |
| 4801 | Pridavanie a nastavenie prav pouzivatelov priradenych k projektu | Active | Ondrej Cickan | | | Ako administrator projektu chcem mat moznost pridať pouzivately a nastavit, ake roly maju pouzivately |
| 4802 | Vytvorenie formularu vsetkych pouzivatelov projektu | Closed | Ondrej Cickan | 10 | | Iba list pouzivatelov, ktorí robia na projekte.<div> </div><div>Nastavenie prav pre role (ak nie som Admin tak nemam pristup do settings.....)</div><div> </div><div>+ načítanie projektov s db a operácie nad databázou</div><div> </div> |
| 4803 | Pridanie formularu pre pridanie pouzivately | Closed | Simon Dekret | 4 | | <div>Noveho pouzivately zacnem pisat do boxu a vyhodi mi navrhly (nacitat pouzivately z DevACTs) <div>Novy pouzivately cez add (prida sa bez roly do zoznamu, rolu mu vyberiem potom)</div></div> |
| 4804 | Pridanie zmeny role do formularu | Closed | Simon Dekret | 3 | | On-the-go bez save (AJAX) |

| ID | Title | State | Assigned To | Completed Work | Remaining Work | Description |
|----------------------|---|--------|-----------------|----------------|----------------|--|
| 4790 | Web CodeCrutches ma nastylovane pouzivatel'ske rozhranie | Active | Anton Jan Vrban | | | |
| 4791 | Definovanie stylu pre uvodnu obrazovku | Closed | Anton Jan Vrban | 2 | | Zoznam projektov vlavo, prazdna plocha pre dashboard a description vpravo, vpravo dole button (vyrazny) pre vstup do projektu |
| 4792 | Definovanie stylu pre menu projektu | Closed | Anton Jan Vrban | 1 | | Menu: <div>TasksUsersStatsSettingsIssue trackersUser rights</div> |
| 4793 | Implementacia polozky tasks | Closed | Anton Jan Vrban | 1 | | Vyuzit implementaciu z predchadzajuceho sprintu a zobrazit na obrazovke pre Tasks |
| 4915 | Implementovanie funkcionalit z pouzivatel'skeho rozhrania | Closed | Anton Jan Vrban | 4 | | Napr dynamicke menu, breadcrumb a pod |



Retrospektíva I. šprintu

Ciele šprintu

- Zoznámiť sa s prostredím a technológiami
- Rozbehať prostredie
- Ucítť sa pracovať v tíme
- Vytvoriť metodiky

Podarilo sa nám / Páčilo sa nám

- Komunikácia s Antonom a Šimonom
- Chalani vyriešili väčšinu vecí
- Pomoc tým, ktorí nestíhali
- Komunikácia s Ondrejom
- Od pondelka to išlo dobre
- Chalani rozbehali Visual Studio

Nepodarilo sa nám / Nepáčilo sa nám

- Neaktivita komunikácie v slacku – riešenie:
 - Zapnúť notifikácie, stiahnuť #Slack do mobilu a desktop verziu
 - Adresovať v kanáli správy (@meno)
- Neaktívne riešenie problémov – riešenie:
 - Development kanal používať ako help (používať!!!)
- Zlá prioritizácia taskov (chýbala) pre lepšie plánovanie – riešenie:
 - Nastaviť predecessors tasku
 - Zadať taskom prioritu a listovať si ich pomocou query (shared queries -> current sprint -> all tasks)

Plánujeme zlepšiť

- Stretnúť sa aj mimo konzultácii (napríklad teamviewer)
 - Rozchodiť teamviewer (každý nainštaluje)
- Priorizovať tasky + deadliny
- Písať problémy do slacku (help/development channel)
- Work-report v slacku – využívať
 - Work report v nedelu do 23:59
- Prehľadne plánovanie



Retrospektíva 2. šprintu

Ciele šprintu

- Dokončiť ASAP task-y zo šprintu 1
- Dokončiť načas tasky šprintu 2
- Zlepšiť komunikáciu v tíme a tímového ducha
- Lepšie organizovať prácu na taskoch
- Pracovať priebežnejšie
- Dodržiavať „stand-up“ v Slacku jedenkrát za týždeň

Podarilo sa nám / Páčilo sa nám

- Zpracovať pull requesty
- Dodržiavať prvé metodiky (Code Review, Branch)
- Pracovali sme priebežne podľa Burn Down Chart
- Stihli sme viac ako predošlý šprint
- Lepšia komunikácia v slack-u
- Nezaseknúť sa na ničom (každý mal vlastnú user story a všetky tasky pod ním mal 1 človek) = nemusel nikto na nikoho čakať
- Osvojili sme si pracovné postupy
- Lepšia práca v tíme

Nepodarilo sa nám / Nepáčilo sa nám

- Dokončiť šprint načas
 - vyčistiť backlog z posledných šprintov
 - zobrať si rovnako story points (nezvyšovať)
- Neintegrovali sme zatiaľ Unit Testy
- Zdĺhavý systém pull requestov
 - vytvoriť pravidlá pre pull requesty (priorita requestov...)
 - Nastaviť súkromnú správu na slack ak ma niekto požiada o pull request
 - čo najrýchlejšie skontrolovať
 - Využívať možnosti statusov v pull requeste (waiting for pull requester, approved with suggestions..)
 - Assign na konkrétnu osobu

Plánujeme zlepšiť

- Integrovať Unit testy
- Zlepšiť dodávanie produktu na čas (splnenie nastaveného story points limitu)
- Pracovať priebežnejšie, aby sme identifikovali problémy včas



Retrospektíva III. šprintu

Ciele šprintu

- Vytvoriť webovú aplikáciu, ktorá bude používateľom umožňovať:
 - prihlasovať sa prostredníctvom DevACTs účtu
 - vytvárať projekty
 - priradovať a spravovať používateľov projektu a určovať ich oprávnenia podľa role (administrátor, manažér, vývojár)
 - zobrazovať tasky používateľov priradených k projektu zaznamenaných cez CodeCrutches plugin pre Visual Studio vrátane ich detailov
- Dokončiť User Stories z predchádzajúcich šprintov (dokumentácie a testy)

Podarilo sa nám / Páčilo sa nám

- Grafika webového projektu
- Rýchlosť pull-requestov
- Učenie sa práce s ASP.NET MVC na webovom projekte
- Spolupráca s Mirom, Antonom, Šimonom a Ondrejom
- Komunikácia cez Slack je stále aktívnejšia, dobre definované témy

Nepodarilo sa nám / Nepáčilo sa nám

- Neplnenie úloh / Nestihnutie šprintu
- Nepíše sa priebežne dokumentácia
- Neriešime priebežne
 - vykazovať hodiny
 - zdefinovať si kedy kto robí
- Aktivita len posledné dni pred koncom šprintu
- Nedodržiavanie dohôd
- Stráca sa tímový duch – každý si rieši svoje
 - potreba team-buildingu
- Príliš široký záber technológií – rýchla práca bez dodržiavania kvalitatívnych štandardov
 - chýba snaha robiť samostatne a učiť sa
- Nedostatočný opis taskov v TFS
 - naučiť sa, ako si podrobne zanalyzovať task
 - skvalitniť opisy taskov

Plánujeme zlepšiť

- Zlepšiť úspešnosť taskov pri šprintoch
- Uviesť do praxe nápady z predchádzajúcich retrospektív
- Priebežná práca
- Systematika vo vyberaní nových technológií a zohľadňovanie ich potreby učenia
- Mikromanažment v rámci User Story
 - nie celá User Story pre jedného človeka
 - radšej nech celý tím pracuje na jednej User Story, potom riešiť ďalšiu
 - priebeh riešenia User Story tímom:



- 1. Dohodnúť postup na začiatku US (stand-up míting)
- 2. Pracovať na taskoch z US priebežne
- 3. Zanalyzovať US, napísať doku
- 4. Pull request, ďalšia US

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Inžinierske dielo

Autori: Ondrej Čičkán, Šimon Dekrét, Dušan Javorník,
Dušan Jom, Miroslav Laco, Anton Ján Vrban

Vedúci projektu: Ing. Karol Rástočný, Phd

Akademický rok: 2016/2017

Dátum odovzdania: 20.11.2016

Obsah

| | | |
|----------|--|----------|
| 1 | Úvod | 3 |
| 1.1 | Prehľad dokumentu | 3 |
| 2 | Globálne ciele projektu | 3 |
| 3 | Celkový pohľad na systém | 4 |
| 3.1 | Pôvodná architektúra v rámci systému | 4 |
| 3.2 | Aktuálna architektúra v rámci systému | 4 |
| 3.3 | Plánovaná architektúra v rámci systému | 5 |
| 3.4 | Dátový model | 7 |
| 3.4.1 | Súčasný stav dátového modelu | 7 |
| 3.4.2 | Krátkodobý plán stavu dátového modelu | 7 |
| 4 | Moduly systému a používateľské príručky | 7 |

1 Úvod

Tento dokument bol vytvorený v rámci predmetu Tímový projekt I, II. Obsahuje časti, ktoré zachytávajú systém, ktorý sme robili z architektonického hľadiska. Sú tu rozobrané jednotlivé moduly z ktorých je systém zložený.

1.1 Prehľad dokumentu

V kapitole 2 sú opísané globálne ciele projektu. Kapitola 3 rozoberá celkový pohľad na systém z rôznych pohľadov. Všetky ostatné kapitoly patria modulom patriacim nášmu systému.

2 Globálne ciele projektu

Práca na rozsiahlych a komplexných projektoch si vyžaduje znalosť nielen používaných technológií, ale aj nutnosť porozumenia veľkému množstvu procesov, ktoré sú spojené s vývojom daného softvéru.

Predpokladaným výstupom tímového projektu je systém, ktorý má slúžiť ako pomôcka pre softvérových vývojárov, ktorý bude kompatibilný s projektmi nainplementovanými predchádzajúcimi tímami pod vedením Ing. Karola Rástočného, PhD.

Jeho dvoma základnými komponentmi sú webové rozhranie a balíček pre rozšírenie vývojového prostredia.

Webová časť systému bude vďaka svojmu grafickému a hlavne zobrazovaciemu potenciálu slúžiť ako manažérsky prostriedok pre prácu na úrovni projektov. Používateľ tu bude môcť založiť nový projekt, zadeliť role v ňom, vytvárať a pridelovať úlohy. Grafické používateľské rozhranie zabezpečí tiež zobrazenie prehľadu úloh na projekte a ich závislostí. Funkcionalita a služby webového komponentu sa budú líšiť podľa role prihláseného používateľa na danom projekte.

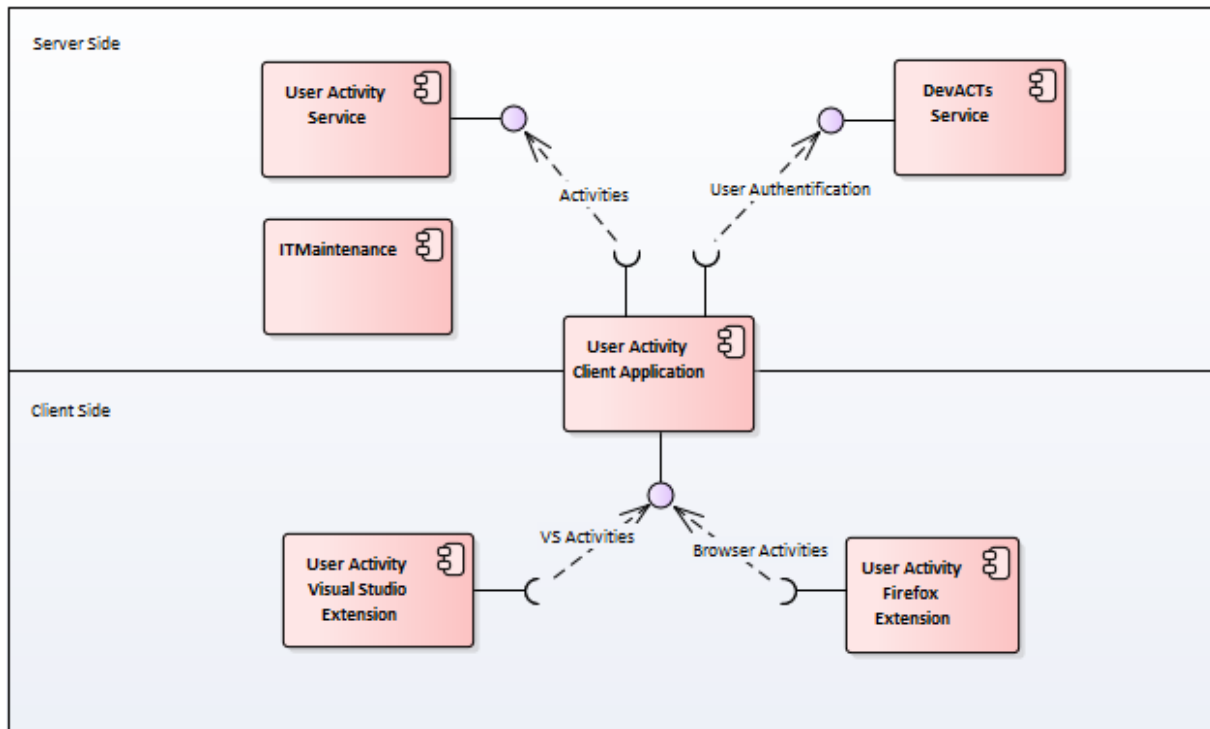
Druhou časťou systému je už spomínané rozšírenie vývojového prostredia. To bude používateľovi priamo v programátorskom rozhraní poskytovať možnosť spravovania úloh, čo zahŕňa vytváranie, časový odhad, editáciu a ukončenie úlohy. Okrem práce s úlohami si vďaka nášmu systému bude môcť vývojár vytvárať šablóny pre zjednodušenie a sprehľadnenie písania kódu, nahlasovať chyby a odosielať ich podľa zvolených preferencií na server, alebo na konkrétneho používateľa. Tento komponent bude poskytovať možnosť voľby projektu, na ktorom sa práve pracuje a prepojenie s webovým rozhraním.

3 Celkový pohľad na systém

Aplikácie vyvíjané v rámci tohto tímového projektu sú vyvíjané ako súčasť systému viacerých aplikácií, ktoré boli vyvíjané v predošlých tímových projektoch vedených Ing. Karolom Rástočným, PhD. Konkrétne sú využívané platformy User Activity, DevACTs a ITMaintenance.

3.1 Pôvodná architektúra v rámci systému

Na diagrame komponentov (obrázok č. 1) sú zobrazené pôvodné prepojenia medzi vybranými komponentmi systému, v rámci ktorého sa realizuje vývoj platformy CodeCrutches ešte pred jej integráciou.



Obrázok 1: Prepojenie doterajších komponentov systému

User Activity

User Activity je platforma, ktorá slúži na zaznamenávanie a zber aktivít používateľov vývojových prostredí (IDE) a prehliadačov, za účelom ich ďalšieho analyzovania. Jej súčasťou je klientská aplikácia *User Activity Client Application*, ktorá zbiera informácie o aktivite používateľa z rozšírenia pre Visual Studio (*User Activity Visual Studio Extension*) a internetový prehliadač Firefox (*User Activity Firefox Extension*), pričom tieto aktivity priebežne ukladá na server (*User Activity Service*). Pre identifikovanie používateľa využíva platformu DevACTs.

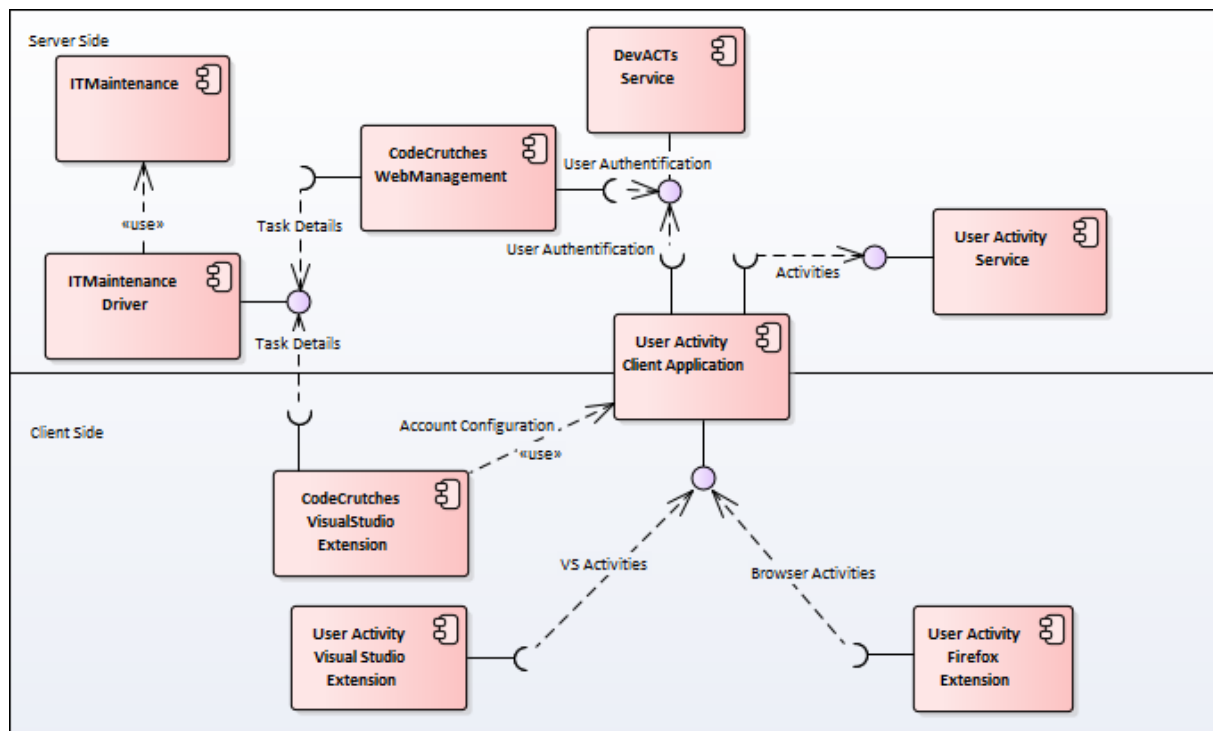
ITMaintenance

Platforma ITMaintenance okrem iného umožňuje ukladať na server dáta s variabilnou štruktúrou a pristupovať k nim.

3.2 Aktuálna architektúra v rámci systému

Nasledujúci diagram komponentov (obrázok č. 2) zobrazuje, ako sa architektúra systému zmenila po integrácii komponentov platformy CodeCrutches vyvíjaných v rámci tohto

tímového projektu – rozšírenia pre vývojárske prostredie Visual Studio (*CodeCrutches VisualStudio Extension*) a webovej aplikácie pre manažment projektov a úloh (*CodeCrutches WebManagement*).



Obrázok 2: Aktuálne prepojenie komponentov v rámci systému po integrácii platformy CodeCrutches

CodeCrutches VisualStudio Extension

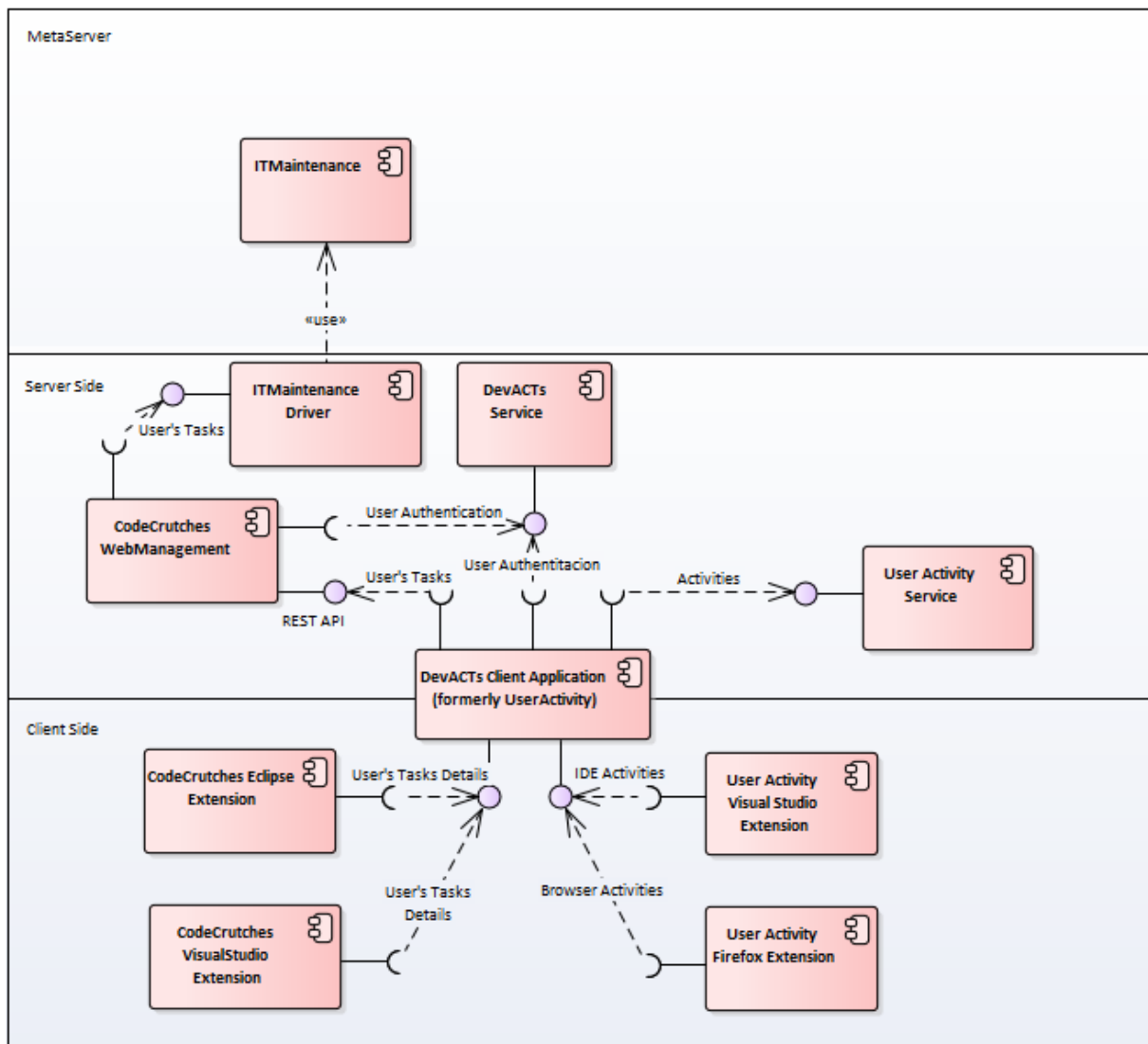
Jedná sa o rozšírenie pre vývojárske prostredie Visual Studio, ktoré vývojárovi poskytuje možnosť vytvárať a upravovať svoje úlohy, vrátane prepínania medzi nimi. Pre identifikáciu používateľa využíva meno používateľa získané z registrov aplikácie *User Activity Client Application*. Dáta o úlohách ukladá na server *ITMaintenance* prostredníctvom knižnice *ITMaintenance Driver*.

CodeCrutches WebManagement

Webová aplikácia CodeCrutches poskytuje používateľovi po prihlásení sa používateľskými údajmi platformy *DevACTs* možnosť vytvárať nové projekty, priradovať k nim používateľov a zobrazovať ich úlohy zaznamenané prostredníctvom rozšírenia CodeCrutches pre Visual Studio. Jednotlivé úlohy sú načítavané zo serveru *ITMaintenance* prostredníctvom knižnice *ITMaintenance Driver*. Informácie o projektoch a používateľoch k nim priradených sa ukladajú do databázy *MongoDB*.

3.3 Plánovaná architektúra v rámci systému

S cieľom zjednodušiť aktuálnu architektúru systému a prepojenie komponentov (zobrazené na obrázku č. 2) a centralizovať zdieľanú logiku bol vytvorený návrh budúcej architektúry, na ktorého realizácii aktuálne pracujeme. Je znázornený na nasledujúcom diagrame komponentov (obrázok č. 3).



Obrázok 3: Plánované prepojenie komponentov systému po zmene architektúry

DevACTs Client Application

Pôvodná aplikácia *UserActivity Client Application* bude v rámci zmeny architektúry premenovaná, pričom bude rozšírená o priamu komunikáciu s CodeCrutches službami (rozšíreniami na klientskej strane a webovou aplikáciou na serverovej strane) a stane sa tak centrálnym bodom komunikácie medzi klientskou a serverovou stranou systému.

CodeCrutches rozšírenia pre vývojárske prostredia

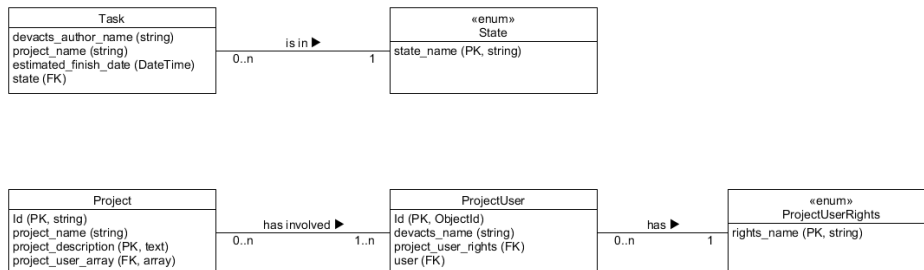
Rozšírenie CodeCrutches pre vývojárske prostredie Visual Studio (*CodeCrutches VisualStudio Extension*) už nebude komunikovať priamo s *ITMaintenance Driverom*, ale s aplikáciou *DevACTs Client Application* prostredníctvom jej rozhrania. To umožní vývoj rozšírení aj pre iné platformy, ktoré nefungujú na .NET technológií (ktorú *ITMaintenance Driver* vyžadoval). Prvým plánovaným je rozšírenie pre vývojárske prostredie Eclipse (*CodeCrutches Eclipse Extension*).

CodeCrutches WebManagement

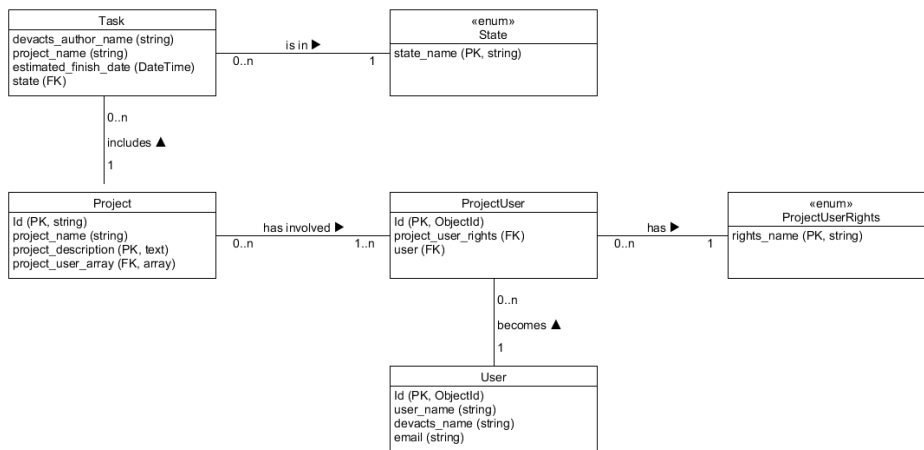
Webová aplikácia CodeCrutches bude v rámci zmeny architektúry systému rozšírená o rozhranie REST API, ktoré bude využívané aplikáciou *DevACTs Client Application* pre zaznamenávanie úloh používateľov na *ITMaintenance* server.

3.4 Dátový model

3.4.1 Súčasný stav dátového modelu



3.4.2 Krátkodobý plán stavu dátového modelu



4 Moduly systému a používateľské príručky

V nasledujúcej časti máme priložené moduly systému, ktoré tvoria našu technickú dokumentáciu. Taktiež tu sú používateľské príručky.

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Modul - Visual Studio Extension

Autori: Ondrej Čičkán, Šimon Dekrét, Dušan Javorník,
Dušan Jom, Miroslav Laco, Anton Ján Vrban

Vedúci projektu: Ing. Karol Rástočný, Phd

Akademický rok: 2016/2017

Dátum odovzdania: 20.11.2016

1 Úvod

Tento dokument bol vytvorený v rámci predmetu Tímový projekt I, II. Obsahuje časti, ktoré zachytávajú systém, ktorý sme robili z architektonického hľadiska. Sú tu rozobrané jednotlivé moduly z ktorých je systém zložený.

2 Komponenty

Hlavný komponent VsExtension predstavuje rozšírenie pre IDE Visual Studio, ktorý má každý používateľ nainštalovaný lokálne vo svojom prostredí. Tento komponent využíva model ITMaintenance.Driver na zápis, aktualizovanie a čítanie informácií o úlohách používateľa zo servera. Modul komunikuje aj s UserActivity, odkiaľ zisťuje používateľské meno.

Rozšírenie nástroja Visual Studio, ktoré vyvíjame komunikuje s rôznymi komponentmi pri zabezpečení správnej funkcionality. Na obrázku Diagram 1 Komponenty je znázornený diagram, ktorý zahŕňa všetky relevantné komponenty.

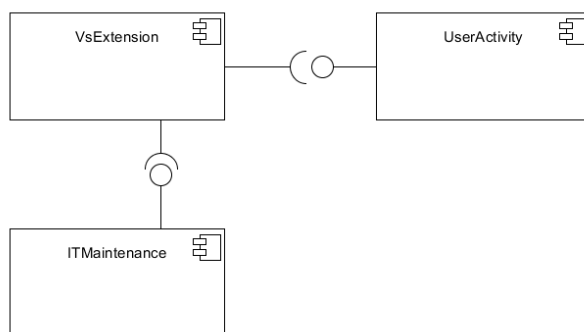


Diagram 1 Komponenty

Pri nainštalovaní aplikácie UserActivity, zadáva používateľ aj svoje meno a heslo. Toto meno je využívané aj VsExtension komponentom. Práve preto je potrebná komunikácia medzi VsExtension a UserActivity komponentom, ktorý nám poskytne meno aktuálne prihláseného používateľa. Toto meno sa berie z registra s názvom user/Name (umiestnenom v HKEY_CURRENT_USER\SOFTWARE\JavaSoft\Prefs\com\gratex\perconik\useractivity\app).

Komponent ITMaintenance nám zase pomáha sprostredkovať komunikáciu s ITServerom. Pre vytvorenie spojenia s ITServerom používame `itmservice.svc` službu. Po pripojení na server, môžeme využívať funkcionality, ktorú nám ponúka. Z tejto funkcionality aktívne používame nasledovné operácie:

- Pridanie novej úlohy
- Vyhľadanie úlohy podľa zadaných parametrov
- Aktualizovanie zvolenej úlohy

3 Dátový model

3.1 Úloha

Model úlohy je uložený v *GeneralInformationTag*, ktorý poskytuje komponent *ITMaintenance.Driver*. Táto trieda nám poskytuje štruktúru na uloženie potrebných informácií o úlohe. Väčšinou atribútov v tejto triede nebudeme potrebovať – nižšie sú opísané len pre nás relevantné polia Diagram 2.

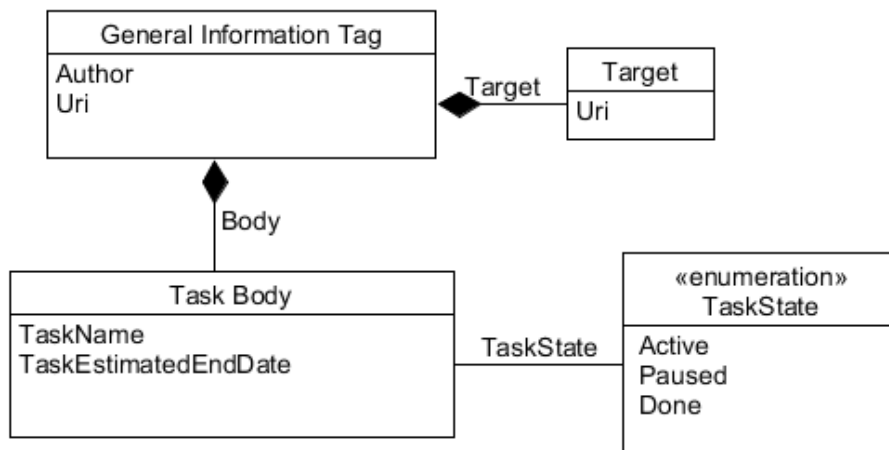


Diagram 2 Dátový model Task

- `taskName` - názov úlohy
- `taskEstimatedEndDate` - predpokladaný dátum ukončenia úlohy
- `taskState` - stav úlohy (bližšie opísaný nižšie)

Okrem nich *GeneralInformationTag* má vlastnú štruktúru, ktorá umožňuje uložiť meno autora a čas vytvorenia úlohy, resp. poslednej modifikácie.

3.1.1 Životný cyklus

Úloha sa môže nachádzať v troch stavoch:

- *Active* – úloha bola vytvorená a programátor na nej pracuje
- *Paused* – programátor v súčasnosti nepracuje na danej úlohe
- *Done* – programátor označil danú úlohu za ukončenú

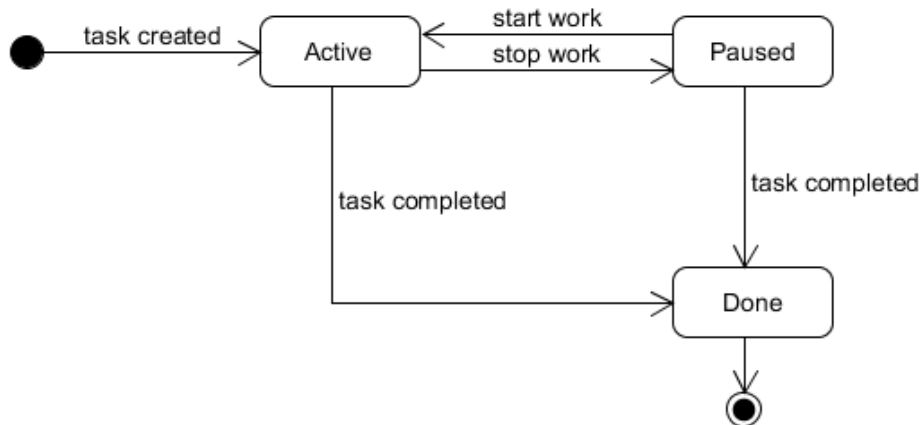


Diagram 3 Životný cyklus úlohy

4 Knižnice a externé projekty

4.1 MongoMigration

Import z DevActs projektu. Zahŕňa knižnice na prácu s Mongo databázou a rozširuje ho o migrácie.

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Modul - Web Management

Autori: Ondrej Čičkán, Šimon Dekrét, Dušan Javorník,
Dušan Jom, Miroslav Laco, Anton Ján Vrban

Vedúci projektu: Ing. Karol Rástočný, Phd

Akademický rok: 2016/2017

Dátum odovzdania: 20.11.2016

1 Úvod

Tento dokument obsahuje modul, ktorý je zameraný na webovú aplikáciu. Jedná sa o manažment riadenia projektov, na ktorých pracujú programátori.

1. Základný popis aplikácie

Webová aplikácia je postavená na platforme ASP.NET MVC a ako databázový server využíva MongoDB.

Používateľské rozhranie je vytvorené v jazykoch HTML5 a CSS3, pričom pre zvýšenie interakcie s používateľom využíva knižnicu jQuery 1.10.2 a framework Bootstrap 3.0.0.

Komunikácia medzi už vytvoreným rozšírením pre Visual Studio IDE (CodeCrutches VS Extension) a touto webovou aplikáciou je momentálne riešená prostredníctvom ITM servera.

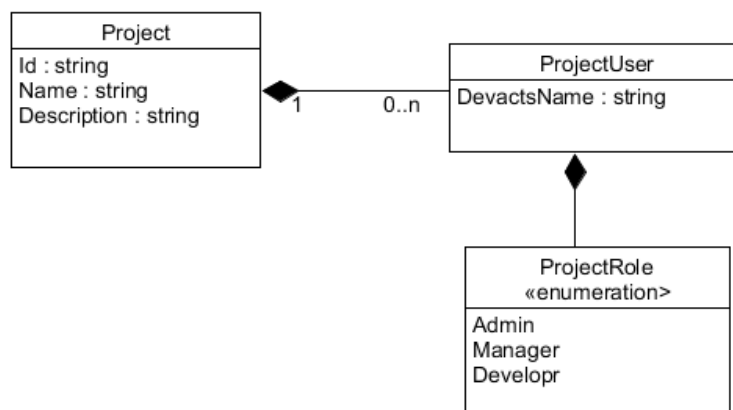
2 Časti modulu

Každý používateľ, ktorý sa prihlási do systému má nastavené vlastné používateľské práva. Používateľ môže mať nasledujúce roly:

- Admin
- Manager
- Developer

1. Dátový model

Používateľ môže vytvoriť nový projekt, na ktorom sa bude pracovať. Projekty sú uložené v Mongo databáze RavensTeamDatabase. Na obrázku Obrázok 1: Dátový model je zobrazený dátový model položky projektu v databáze.



Obrázok 1: Dátový model

Každý projekt má vlastné id, ktorým je jednoznačne identifikovateľné a používateľ o ňom nevie, pretože je generované automaticky v systéme. Pri vytváraní projektov zadáva používateľ

meno (name) projektu, opis (description) projektu a zoznam používateľov (ProjectUserList), ktorí na danom projekte pracujú. Developeri sú k projektu asociovaní manažérmi. Na obrázku Obrázok 2 - Ukážkový JSON záznam v databáze zobrazený ukážkový JSON záznam v databáze MongoDB.

```
{
  "_id": "582baf5e02ca811d34d78538",
  "Name": "Test Project",
  "Description": "First Test Project",
  "ProjectUserList": [
    {
      "DevactsName": "xcickano",
      "Role": "Admin"
    },
    {
      "DevactsName": "xdekret",
      "Role": "Developer"
    },
    {
      "DevactsName": "xvrban",
      "Role": "Manager"
    }
  ]
}
```

Obrázok 2 - Ukážkový JSON záznam v databáze

3 Funkcie

1. Zobrazenie úloh k projektu

Táto funkcia poskytuje načítavanie úloh všetkých používateľov.

Obslužné triedy

Controller: CodeCrutches.WebManagement.Controllers.ProjectController

Akcia controllera: Tasks

Model: CodeCrutches.WebManagement.Models.ProjectViewModels

Akcia modelu: TasksViewModel

View: CodeCrutches.WebManagement.Views.Project.Tasks

Pre načítavanie zoznamov úloh a prácu so štruktúrou *GeneralInformationTag* ktorou sú tieto zoznamy reprezentované, aplikácia využíva spojenie s ITM serverom prostredníctvom projektu *CodeCrutches.Common*. Pre načítavanie zoznamov sú využívané tieto metódy:

- TaskTagHandler.GetAllUnfinishedTasks() – pre získanie zoznamu aktívnych úloh
- TaskTagHandler.GetAllFinishedTasks() – pre získanie zoznamu ukončených úloh

2. Prihlasovanie pomocou DevACTs účtu

Táto funkcia umožňuje autentifikovať sa voči systému a tak sprístupniť väčšinu ostatných funkcionality.

Obslužné triedy

Controller: CodeCrutches.WebManagement.Controllers.AccountController

Akcia controllera: Login, Logout

Model: CodeCrutches.WebManagement.Models.AccountViewModels,
CodeCrutches.WebManagement.Models.User

Akcia modelu: LoginViewModel, User

View: CodeCrutches.WebManagement.Views.Account.Login,
CodeCrutches.WebManagement.Views.Account._LoginLayout

Na autentifikáciu aplikácia využíva triedu DevActsVerification z projektu Core.

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

Používateľská príručka Visual Studio Extension

RavensTeam

Akademický rok: 2016/2017

Vedúci práce: Ing. Karol Rástočný, Phd.

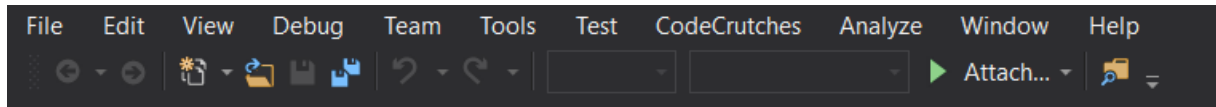
Členovia tímu: Ondrej Čičkán, Šimon Dekrét, Dušan Javorník
Dušan Jom, Miroslav Laco, Anton Ján Vrban

Úvod

Pre použitie doplnku pre Visual Studio je potrebné mať nainštalovanú aktuálnu verziu Visual Studio IDE od Microsoft v akejkoľvek konfigurácii. Taktiež je potrebné mať nainštalovaný program User Activity, a rozšírenie User Activity pre Visual Studio. Program User Activity musí byť počas používania rozšírenia pre Visual Studio zapnutý a je potrebné mať vytvorený účet DevACTs.

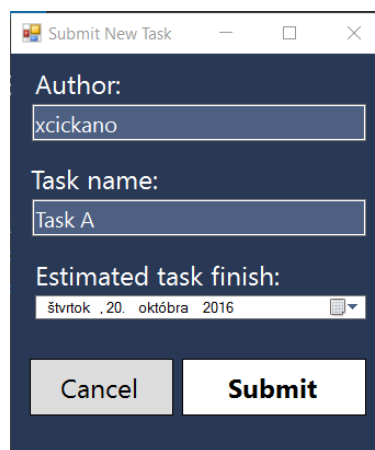
Opis používateľského rozhrania

Po nainštalovaní doplnku sa v ovládacom paneli objaví nové menu s názvom *CodeCrutches*.



Menu *CodeCrutches* ponúka nasledovné možnosti:

- **Start New Task** – akcia na zaevidovanie novej úlohy. Po kliknutí sa zobrazí Formulár 1.
 - *Author* - zobrazený identifikátor súčasného autora UserActivity
 - *Task name* - (meno) úlohy
 - *Estimated task finish* - odhadovaný čas dokončenia tejto úlohy

A screenshot of a dialog box titled 'Submit New Task'. It has three input fields: 'Author' with the text 'xcickano', 'Task name' with the text 'Task A', and 'Estimated task finish' with a date picker showing 'štvrtok . 20. októbra 2016'. At the bottom, there are two buttons: 'Cancel' and 'Submit'.

Formulár 1 Zaevidovanie novej úlohy

- **Manage Tasks** – otvorí okno pre manažment všetkých nedokončených úloh (Formulár 2), ktoré sú priradené na aktuálne prihláseného používateľa. V okne môžeme vidieť list úloh s informáciami o ich stave a predpokladaného času ukončenia. Po zvolení úlohy je možné danú úlohu zastaviť/aktivovať, zmeniť predpokladaný čas ukončenia (po stačení sa objaví Formulár 3) alebo ukončiť úlohu – po tomto kroku daná úloha zmizne zo zoznamu.

| Name | State | Estimated End Time |
|------|--------|--------------------------|
| test | Paused | sobota, 29. októbra 2016 |

Formulár 2 Okno na manažment úloh

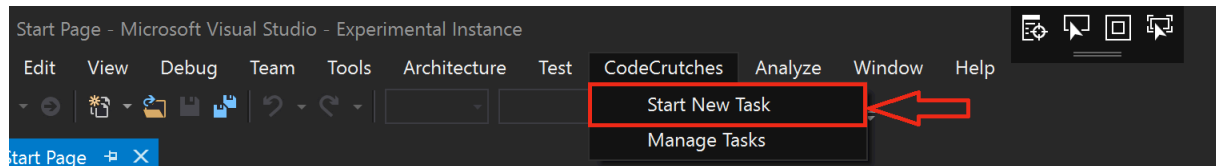
Formulár 3 Okno na zmenu predpokladaného ukončenia úlohy

Úloha

Zaevidovanie začatia novej úlohy

Uistite sa, že aplikácia User Activity je korektné spustená a máte nainštalovaný plugin User Activity a CodeCrutches vo Visual Studio. Pre zaevidovanie začiatku práce na novej úlohe vykonajte tieto kroky:

1. Vyberte v menu CodeCrutches možnosť „Start new task“:

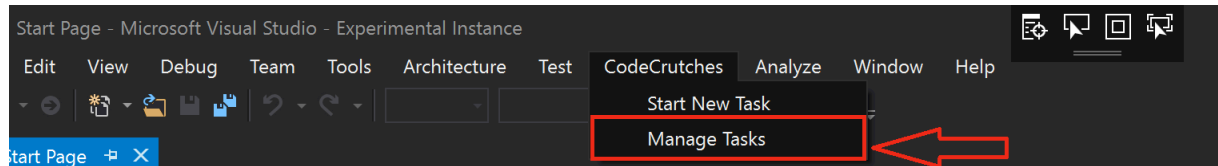


2. Zobrazí sa formulár pre začiatok novej úlohy s predvyplneným menom z DevACTs (1). Pokiaľ predvyplnené meno nemáte, nepokračujte v ďalších krokoch. Aplikácia User Activity nepracuje správne. Uistite sa, že je aplikácia korektné nainštalovaná, spustená a že ste prihlásený.

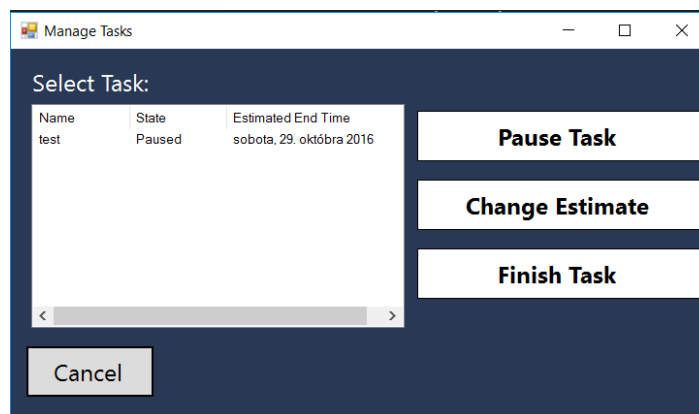
3. Vyplňte polia „Task name“ a „Estimated task finish“. Obe polia sú povinné.
Poznámka- odhad predpokladaného ukončenia má iba informatívny charakter a je možné ho dodatočne meniť.
4. Kliknite na tlačidlo „Submit“.
5. Budete informovaný o výsledku zaevidovania začatia novej úlohy pomocou okna so správou o úspechu/neúspechu začatia novej úlohy

Úprava odhadovaného dátumu ukončenia úlohy

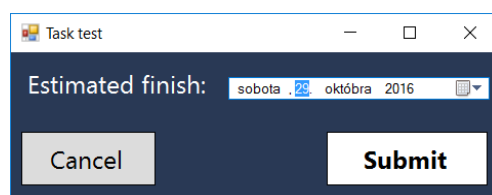
1. Vyberte v menu CodeCrutches možnosť „Manage tasks“:



2. Zobrazí sa formulár pre manažment Vašich úloh:



3. Kliknite na úlohu, ktorej chcete zmeniť odhadovaný dátum ukončenia a s vybranou úlohou kliknite na tlačidlo „Change Estimate“.
4. Zobrazí sa formulár pre zmenu predpokladaného dátumu ukončenia úlohy:



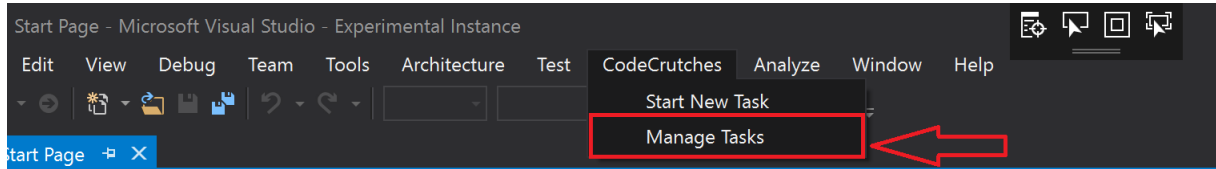
5. Vypíšte nový predpokladaný dátum ukončenia úlohy, alebo vyberte dátum z kalendára, ktorý sa zobrazí po kliknutí na ikonu kalendára vpravo vedľa vypísaného dátumu.
6. Kliknite na tlačidlo „Submit“
7. Budete informovaný o výsledku aktualizácie predpokladaného dátumu ukončenia úlohy.

Poznámka- vytvorenie novej úlohy okamžite priradí novej úlohe stav „aktívna“ a preruší momentálne aktívnu úlohu. Pre zmenu aktívnej úlohy vid' podkapitolu Prerušenie/aktivovanie úlohy

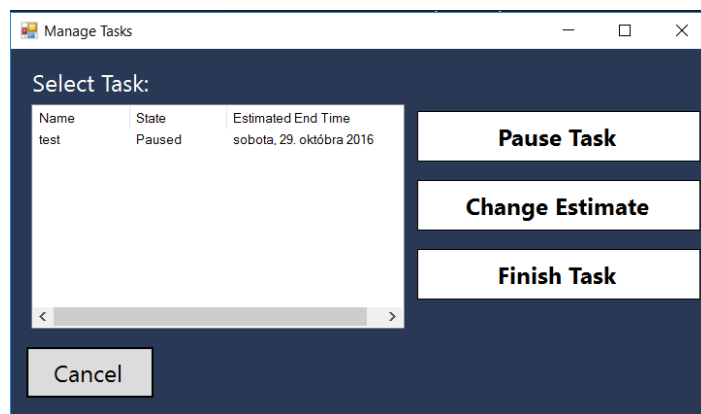
Prerušenie/aktivovanie úlohy

Rozšírenie CodeCrutches dovoľuje mať aktívnu žiadnu, alebo práve 1 úlohu. Pre prerušenie/aktivovanie úlohy prevedte tieto kroky:

1. Vyberte v menu CodeCrutches možnosť „Manage tasks“:



2. Zobrazí sa formulár pre manažment Vašich úloh:

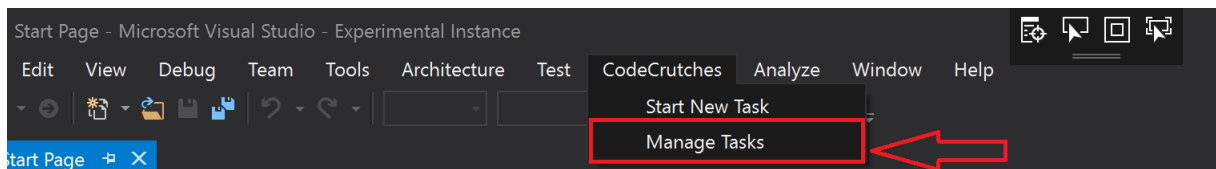


3. Kliknite na úlohu, ktorej stav („state“) si želáte zmeniť.
 - Pokiaľ je úloha aktívna („Active“), tak kliknite na tlačidlo „Pause Task“ pre jej prerušenie. Budete mať následne 0 aktívnych úloh.
 - Pokiaľ je úloha prerušená („Paused“), tak tlačidlo „Pause Task“ bude mať zmenený text na „Activate Task“. Kliknite na toto tlačidlo pre aktivovanie úlohy.

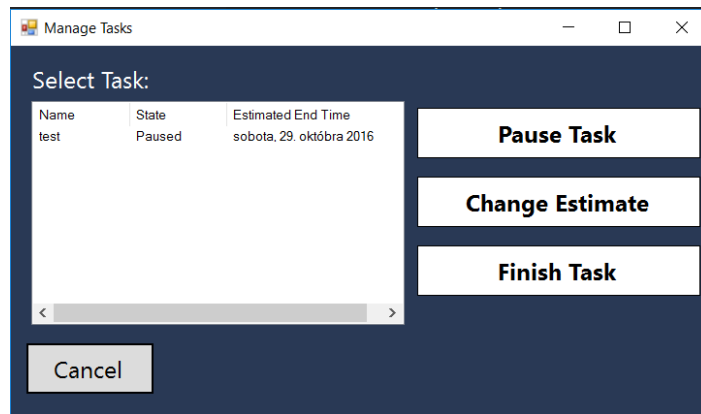
Poznámka- pokiaľ aktivujete úlohu, tak doposiaľ aktívna úloha sa preruší.

Ukončenie úlohy

1. Vyberte v menu CodeCrutches možnosť „Manage tasks“:



2. Zobrazí sa formulár pre manažment Vašich úloh:



3. Kliknite na úlohu, ktorú chcete ukončiť a s vybranou úlohou stlačte tlačidlo „Finish Task“.
4. Budete informovaný o výsledku ukončenia úlohy.

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

Používateľská príručka Web management

RavensTeam

Akademický rok: 2016/2017

Vedúci práce: Ing. Karol Rástočný, Phd.

Členovia tímu: Ondrej Čičkán, Šimon Dekrét, Dušan Javorník
Dušan Jom, Miroslav Laco, Anton Ján Vrban

1 Úvod

Webové rozhranie Code Crutches slúži pre sledovanie a manažment úloh zaznamenaných cez rozšírenie Code Crutches pre vývojové prostredie Visual Studio.

Po prihlásení prostredníctvom účtu DevACTs používateľovi umožňuje vytvárať nové projekty, prípadne prezerať detaily už aktívnych projektov.

1.1 Základný popis používateľského rozhrania

1.1.1 Ovládacie prvky

Používateľské rozhranie sa skladá z nasledujúcich základných ovládacích prvkov:

1. Hlavné menu
2. Menu používateľa
3. Nadpis aktuálnej stránky
4. Navigačná lišta
5. Logo

V nasledujúcich podkapitolách sú tieto prvky vyznačené im prislúchajúcim číslom na snímku obrazovky používateľského rozhrania.

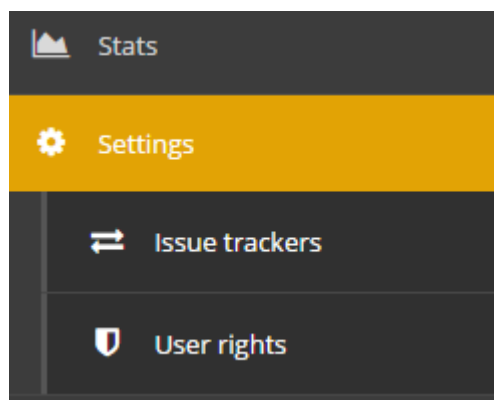
1.1.2 Klasické zobrazenie

The screenshot displays the Code Crutches web interface. At the top, there is a navigation bar with a hamburger menu icon (5), the text '<code>//crutches' (5), and a user profile dropdown (2) for 'xrbana'. On the left, a sidebar menu (1) contains 'Dashboard', 'Tasks', 'Team Members', 'Stats', and 'Settings'. The main content area (3) is titled 'All Tasks' and features three summary cards: '20/50 Hours completed' with an hourglass icon, '6/14 Tasks finished' with a bar chart icon, and '5/7 Members active last 24h' with a group icon. Below these is a table of 'Active Tasks' with columns for Task Name, Author, Created at, Estimated End Date, and State. The table lists 8 tasks, with 5 in a 'Paused' state and 3 in an 'Active' state. A search bar and pagination controls (Previous, 1, Next) are also visible.

| Task Name | Author | Created at | Estimated End Date | State |
|---------------|----------|---------------------|----------------------------|--------|
| 6666 | xlacom | 06.11.2016 21:33:02 | piatok, 18. novembra 2016 | Paused |
| 7777 | xlacom | 06.11.2016 21:32:55 | nedela, 6. novembra 2016 | Active |
| matuis | xrbana | 11.11.2016 19:04:58 | piatok, 11. novembra 2016 | Active |
| my task | xoickano | 06.11.2016 20:53:00 | piatok, 11. novembra 2016 | Paused |
| najnovsi task | xoickano | 06.11.2016 21:15:04 | nedela, 6. novembra 2016 | Paused |
| new Task | xoickano | 06.11.2016 20:51:10 | štvrtok, 10. novembra 2016 | Active |
| novy task | xrbana | 11.11.2016 17:41:39 | piatok, 11. novembra 2016 | Paused |
| Z | xjom | 11.11.2016 9:14:37 | streda, 30. novembra 2016 | Active |

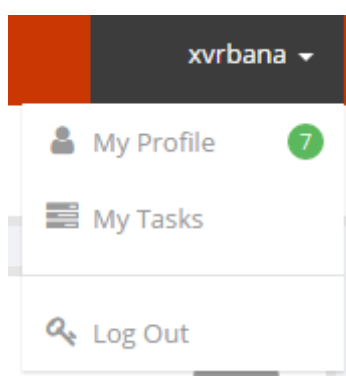
Pod-položky v hlavnom menu

Položka v hlavnom menu (označené číslom 1) môže obsahovať ďalšie pod-položky, ktoré sa zobrazia po kliknutí na hlavnú položku:



Používateľské menu

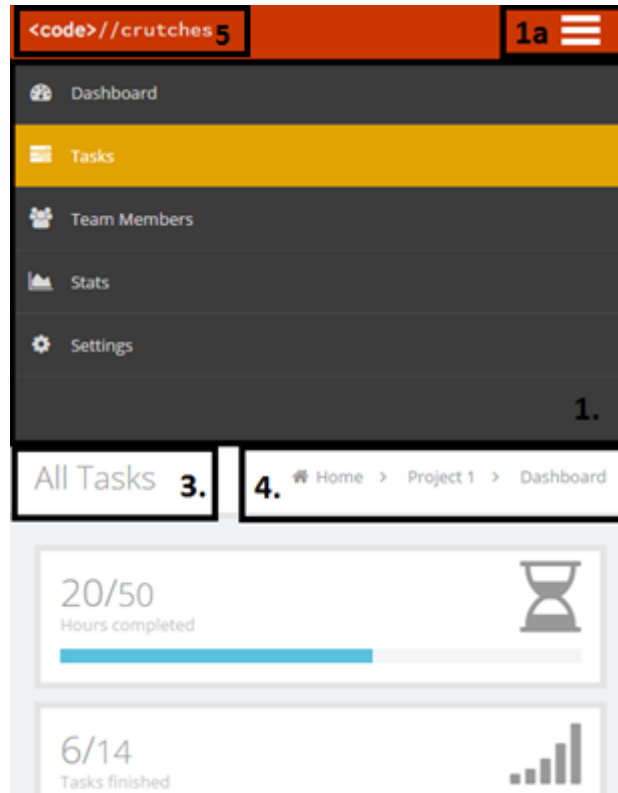
Používateľské menu (označené číslom 2) je prístupné po kliknutí na prihlasovacie meno používateľa.



Návrat na úvodnú stránku

Kliknutím na logo (označené číslom 5) sa používateľ vráti na hlavnú stránku.

1.1.3 Zobrazenie na mobilnom zariadení



Zobrazenie a skrytie hlavného menu

Zobraziť alebo skryť hlavné menu (označené číslom 1) je v mobilnom zobrazení možné kliknutím na tlačidlo umiestnené v hornej navigačnej lište (označené číslom 1a).

Návrat na úvodnú stránku

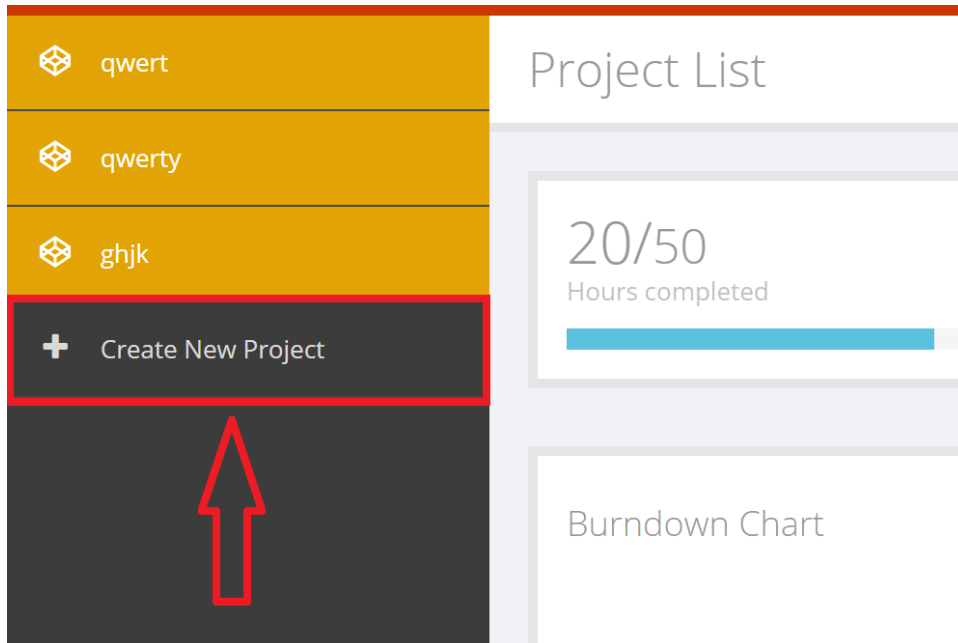
Kliknutím na logo (označené číslom 5) sa používateľ vráti na hlavnú stránku.

2 Projekt

2.1 Vytvorenie nového projektu

Pre pridanie nového projektu do aplikácie CodeCrutches je potrebné byť vo webovej aplikácii CodeCrutches prihlásený pomocou DevACTs účtu (kapitola 4). Pokiaľ ste prihlásený, pre vytvorenie nového projektu vykonajte postupne tieto kroky:


1. Vyberte v menu pre výber aktívneho projektu možnosť "+ Create New Project"




2. Načíta sa obrazovka pre pridanie nového projektu. Vyplňte, prosím, položky názov a popis projektu. Upozornenie- názov projektu môže byť ľubovoľný, no nesmie zostať nevyplnený. Pole popis projektu je dobrovoľné.

Create New Project

Enter your new project details

Name 

Description 

3. Kliknite na tlačidlo "Create project". Projekt sa vytvorí a okamžite pribudne do menu projektov. Nový projekt bude mať 1 prideleného používateľa - Vás. Práva máte predvolene nastavené ako admin.

Create New Project

Enter your new project details

Name

Description

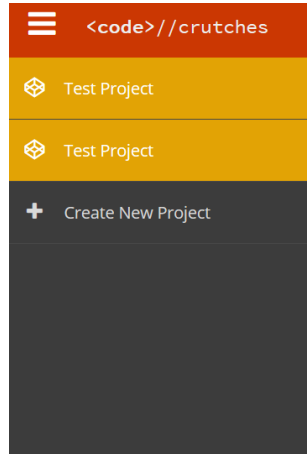
Create Project ←

3 Manažment projektu

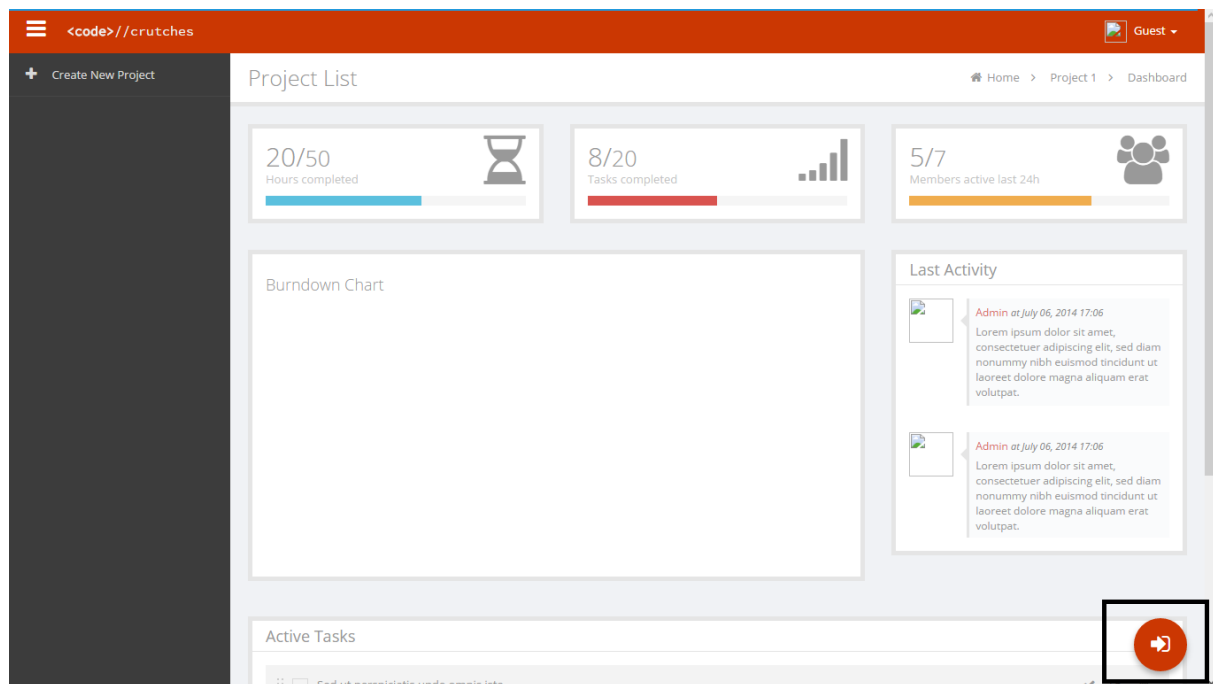
3.1 Výber aktívneho projektu a jeho manažment

3.1.1 Výber projektu a zobrazenie jeho detailov

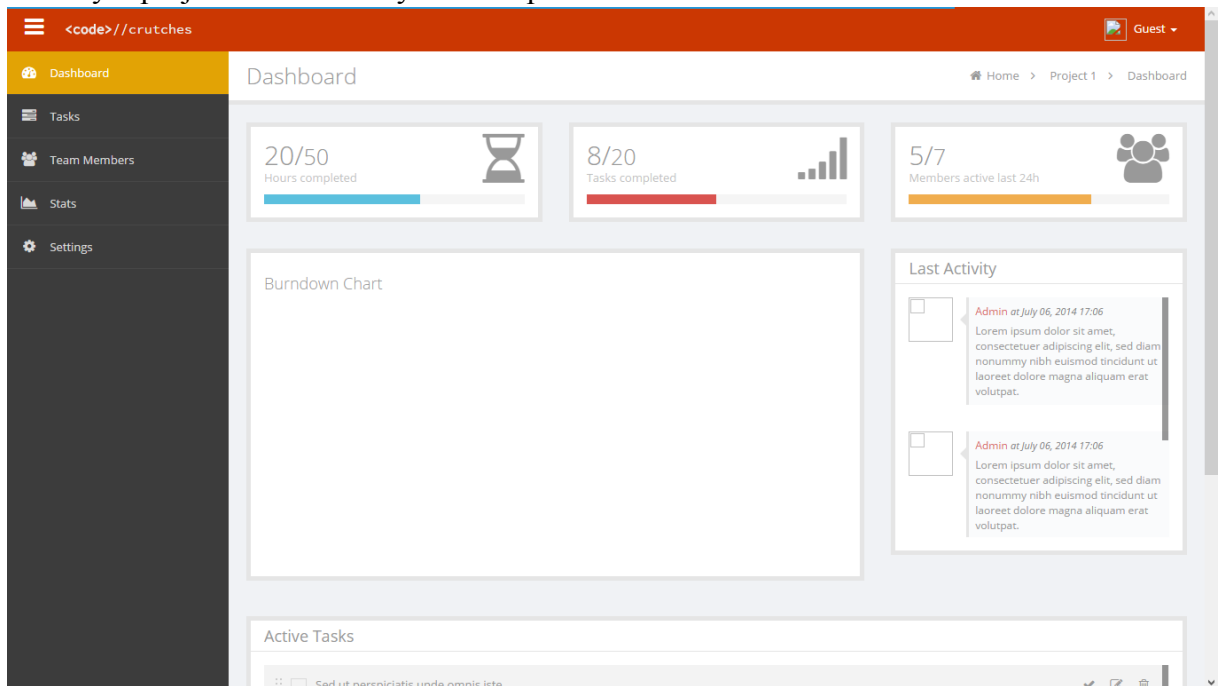
1. Na úvodnej stránke po prihlásení, vyberte z hlavného menu projekt, ktorý chcete spravovať:



2. Zobrazia sa detaily zvoleného projektu. Kliknite na tlačidlo umiestnené v pravom dolnom rohu:

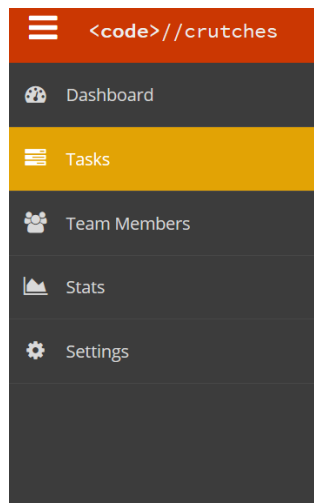


3. Projekt označíte ako aktívny a vstúpite do jeho správy. Jednotlivé akcie súvisiace so zvoleným projektom môžete vykonávať prostredníctvom hlavného menu:



3.1.2 Zobrazenie úloh projektu

1. V hlavnom menu kliknite na položku *Tasks*.



2. Zobrazia sa informácie o aktívnych aj ukončených úlohách patriacich k vybranému projektu:

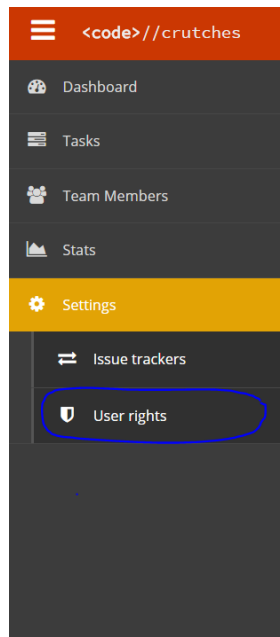
3. Zoznamy úloh je možné filtrovať pomocou vyhľadávania umiestneného nad zoznamom. Vyhľadávať je možné akýkoľvek reťazec (aj jeho časť) a vyhľadávanie prebieha nad všetkými stĺpcami tabuľky priamo pri písaní vyhľadávacej frázy:

3.2 Nastavenie používateľských práv

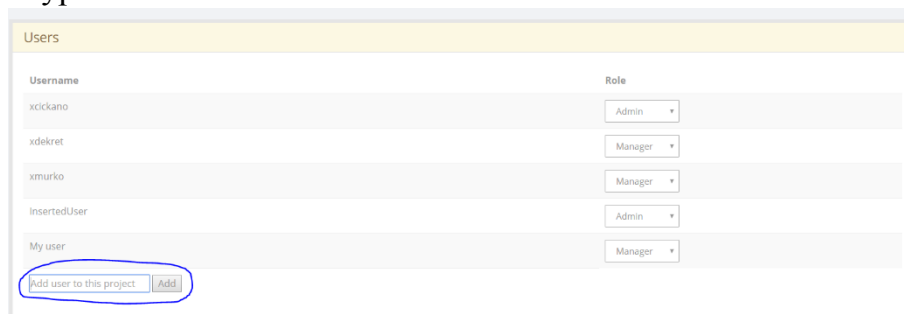
3.2.1 Pridanie nového používateľa

Pridať nového používateľa do projektu môže len používateľ s právami manažéra.

1. Vyberte projekt, do ktorého chcete pridať nového používateľa
2. Kliknite položku *Settings* a vyberte *User rights*.



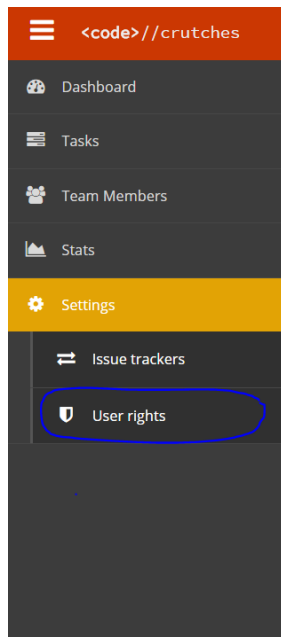
3. Na konci zoznamu používateľov je možnosť napísania mena nového používateľa. Toto pole vyplňte a stlačte tlačidlo *Add*.



3.2.2 Zmena roly používateľa

Zmeniť rolu používateľa môžu taktiež len používatelia, ktorí majú práva manažera.

1. Vyberte projekt v ktorom chcete zmeniť práva používateľa.
2. Kliknite položku *Settings* a vyberte *User rights*.



3. Na konci každého riadku je zobrazené právo daného používateľa. Kliknite na aktuálne právo a zo zoznamu vyberte nové právo.

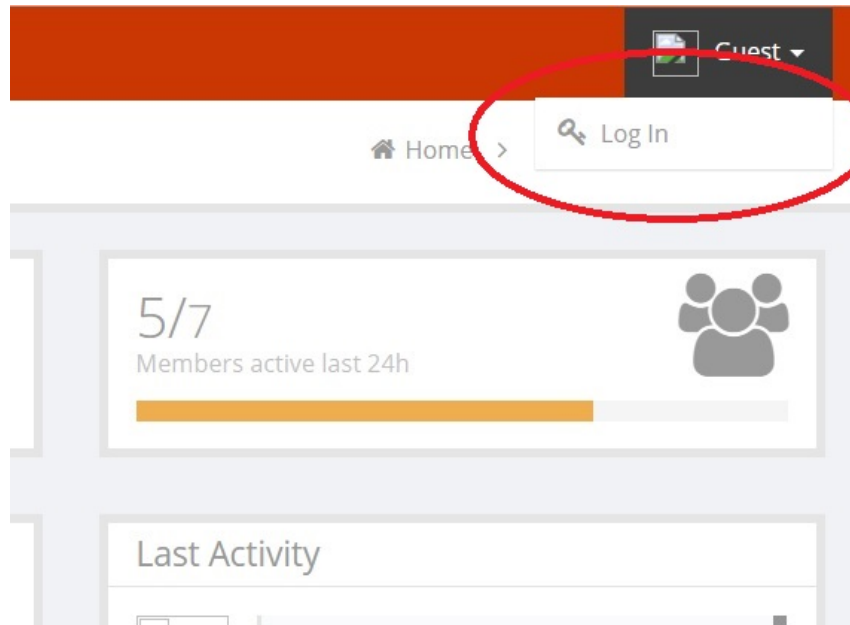
| Users | |
|----------|---------|
| Username | Role |
| xcickano | Admin |
| xdekret | Manager |

4 Prihlásenie

Pre použitie veľkej časti systému je nutné byť prihlásený do systému. Táto časť popisuje, ako je možné sa do systému prihlásiť

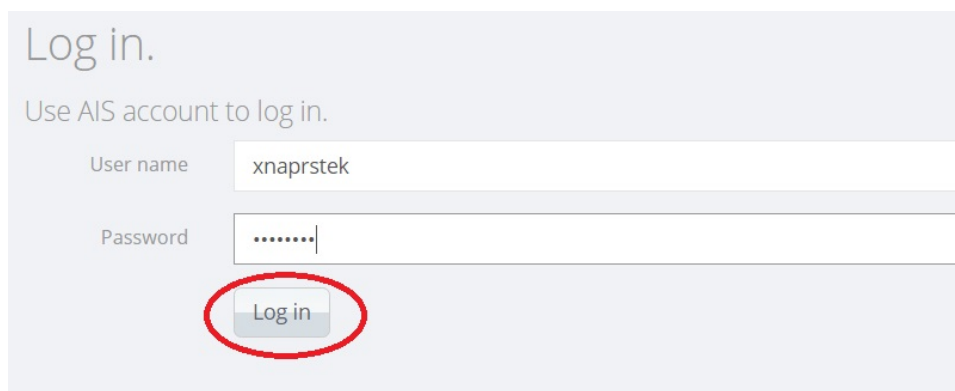
4.1 Prihlasovacia stránka

Na stránku s formulárom pre prihlásenie sa do systému sa dostaneme kliknutím na tlačidlo "Log In" výsuvnom menu nachádzajúcom sa v pravom hornom rohu úvodnej webstránky.



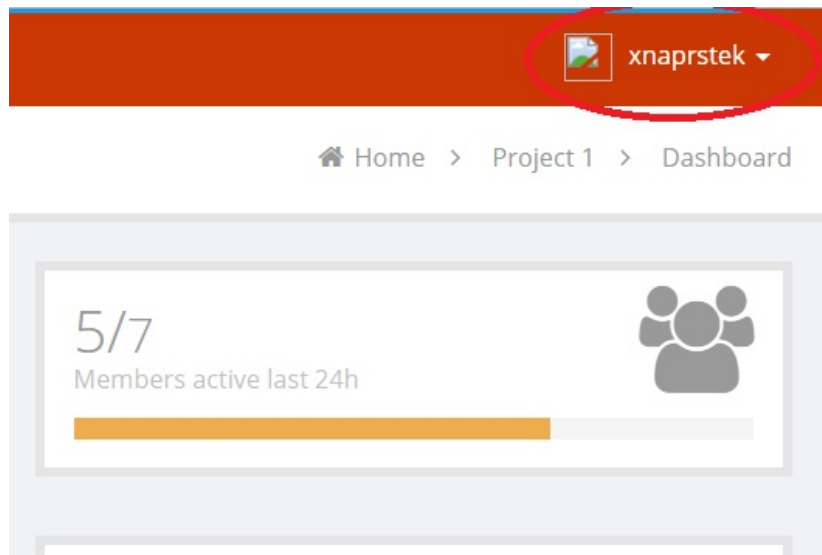
4.2 Prihlásenie

Do formuláru vyplňte požadované údaje (prihlasovacie meno a heslo) a kliknite na tlačidlo "Log In".

A screenshot of a login form. The form has a light gray background. At the top, it says 'Log in.' followed by 'Use AIS account to log in.' Below this are two input fields: 'User name' with the text 'xnaprstek' and 'Password' with a masked password '.....'. At the bottom of the form, there is a 'Log in' button, which is circled in red.

4.3 Kontrola

Pokiaľ zadané údaje boli správne, stránka bude presmerovaná na úvodnú a v pravom hornom rohu bude uvedené vaše meno. Od tohto momentu ste prihlásený.



4.4 Odhlásenie

Ak sa chcete zo systému odhlásiť, kliknite na tlačidlo "Log Out" schované v rozbaľovacom menu v pravom hornom rohu pod vašim menom. Od tohto okamihu ste odhlásený. Stránka vás po kliknutí presmeruje na stránku prihlasovania.

