

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

## Inžinierske dielo

**Akademický rok:** 2016/2017

**Vedúci práce:** Ing. Karol Rástočný, Phd.

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

**Dátum poslednej zmeny:** 14.12.2016

## Obsah

<b>1</b>	<b>Úvod .....</b>	<b>3</b>
1.1	Prehľad dokumentu .....	3
<b>2</b>	<b>Globálne ciele projektu .....</b>	<b>4</b>
<b>3</b>	<b>Celkový pohľad na systém .....</b>	<b>5</b>
3.1	Pôvodná architektúra v rámci systému.....	5
3.1.1	User Activity.....	5
3.1.2	ITMaintenance.....	5
3.2	Aktuálna architektúra v rámci systému .....	5
3.2.1	CodeCrutches VisualStudio Extension .....	6
3.2.2	CodeCrutches WebManagement .....	6
3.3	Plánovaná architektúra v rámci systému .....	6
3.3.1	DevACTs Client Application .....	7
3.3.2	CodeCrutches rozšírenia pre vývojárske prostredia .....	7
3.3.3	CodeCrutches WebManagement .....	7
3.4	Dátový model.....	8
3.4.1	Predošlý vývoj dátového modelu .....	8
3.4.2	Súčasný dátového modelu.....	8
<b>4</b>	<b>Moduly systému a používateľské príručky .....</b>	<b>9</b>

# **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 0 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 naimplementovanými 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.

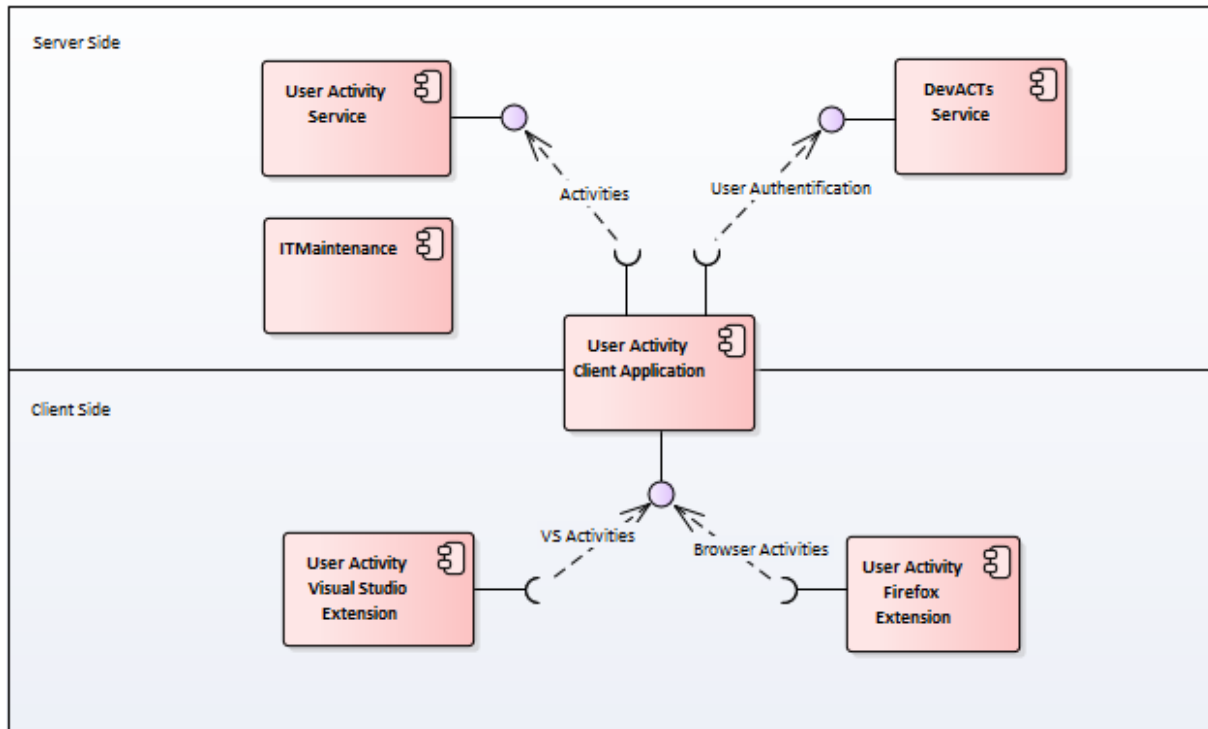
Druhou časťou systému je už spomínané rozšírenie vývojového prostredia. To bude používateľovi priamo v programátorskom rozhraní poskytovať možnosť spravovania úloh, čo zahŕňa vytváranie, časový odhad, editáciu a ukončenie úlohy. Okrem práce s úlohami si vďaka nášmu systému bude môcť vývojár vytvárať 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

##### 3.1.1 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.

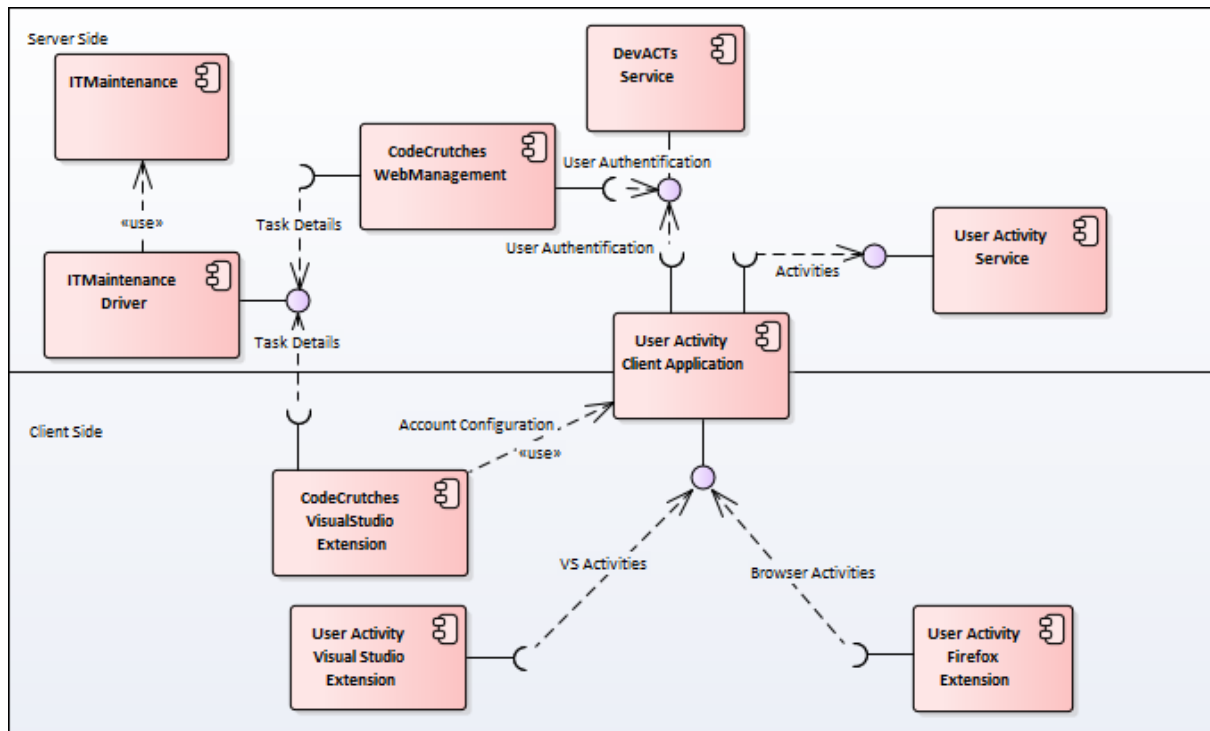
##### 3.1.2 ITMaintenance

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

### 3.2.1 CodeCrutches VisualStudio Extension

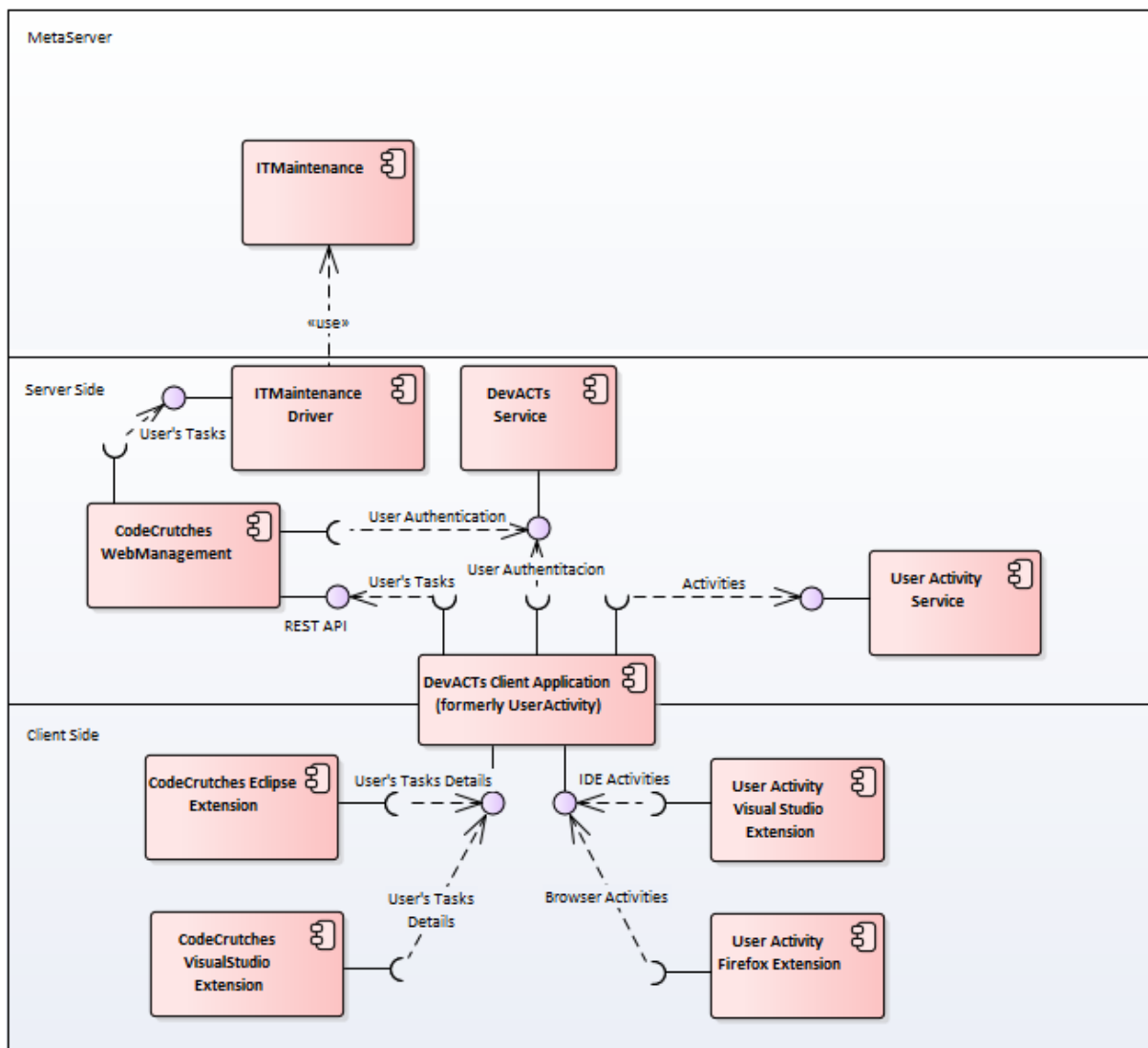
Jedná sa o rožšírenie pre vývojárske prostredie Visual Studio, ktoré vývojárovi poskytuje možnosť vytvárať a upravovať svoje úlohy, vrátane prepínania medzi nimi. Pre identifikáciu používateľa využíva meno používateľa získané z registrov aplikácie *User Activity Client Application*. Dáta o úlohách ukladá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

### 3.2.2 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

### 3.3.1 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.

### 3.3.2 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 technológii (ktorú *ITMaintenance Driver* vyžadoval). Prvým plánovaným je rozšírenie pre vývojárske prostredie Eclipse (*CodeCrutches Eclipse Extension*).

CodeCrutches WebManagement

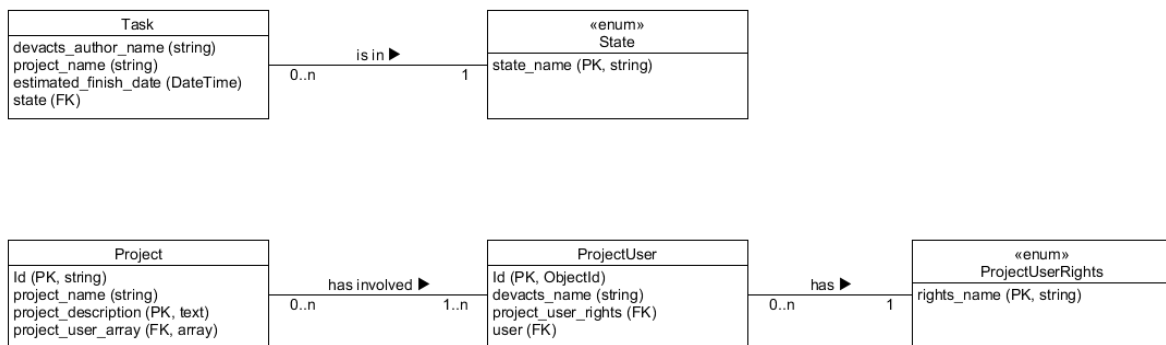
Webová aplikácia CodeCrutches bude v rámci zmeny architektúry systému rozšírená o rozhranie REST API, ktoré bude využívané aplikáciou *DevACTs Client Application* pre zaznamenávanie úloh používateľov na *ITMaintenance* server.

### 3.4 Dátový model

#### 3.4.1 Predošlý vývoj dátového modelu

Dátový model aplikácie CodeCrutches prešiel od začiatku vývoja viacerými zásadnými zmenami, ktoré v tejto kapitole dokumentujeme. Zmeny sa diali zväčša pri pridávaní novej funkcionality do aplikácie.

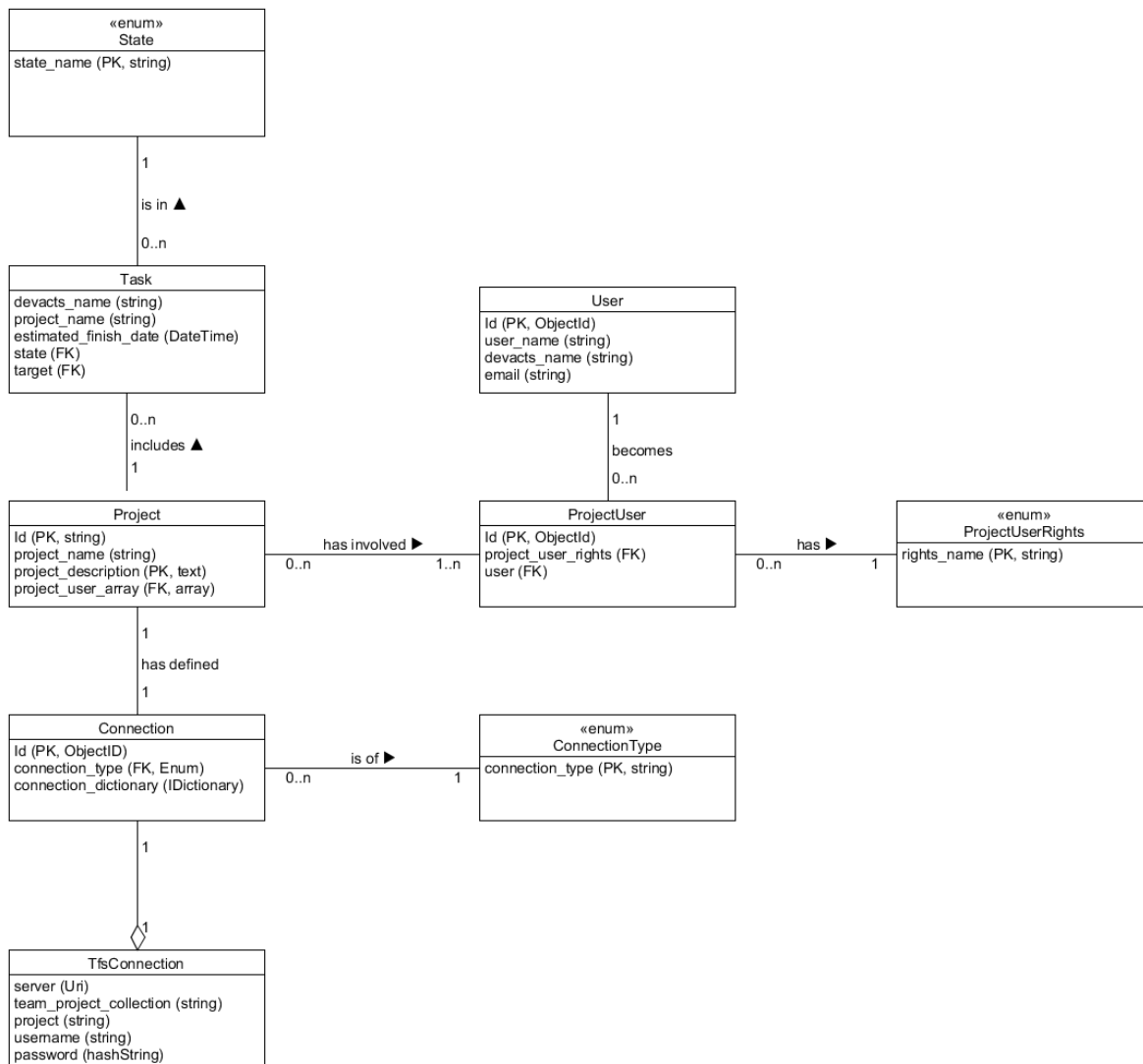
1. Prvotný stav dátového modelu. Aplikácie VSExtension a WebManagement spolu zatiaľ nekomunikujú a nezdieľajú dáta.



#### 3.4.2 Súčasný dátového modelu

Súčasný stav dátového modelu je zobrazený nižšie. Oproti predošlému stavu sme pridali model pre ukladanie generických pripojení na aplikácie pre projekt manažment. Implementovali sme model pre konekciiu do TFS. Momentálne sa pracuje na prepojení Task s Project modelom, aby sme si vedeli ukladať úlohy pod projekty.





## 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

Ilkovičova 2, 842 16 Bratislava 4

## **Modul – DevACT client**

**Akademický rok:** 2016/2017

**Vedúci práce:** Ing. Karol Rástočný, Phd.

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

**Dátum poslednej zmeny:** 9.12.2016

# 1 Úvod

Tento dokument obsahuje opis modulu DevACT client (predchádzajúci názov UserActivity Client), ktorý pôvodne slúžil len na zachytávanie činnosti programátora a posielanie udalostí na server. Modul bol premenovaný a rozšírený o novú funkcionálnosť. Dokument opisuje modul len z hľadiska využitia komunikácie s modulom CodeCrutches WebManagement.

## 2 Komponenty

### 2.1 CodeCrutches služba

Služba je vystavená v podobe REST API a jej úlohou je autentifikovať prichádzajúcu správu a preposielať ich do WebManagementu.

Služba počúva na predvolenom porte 16376 a obsluhuje všetky http správy s metódou POST, GET alebo PUT, ktoré vyhovujú ceste `/codecrutches/{service}`. Port sa dá zmeniť priamo v aplikácii (viď. [Používateľská príručka DevACTs](#)), alebo zmenou registra `code_crutches_port` na ceste `,SOFTWARE\JavaSoft\Prefs\`.

Na vstupe prijíma ľubovoľnú správu vo formáte Json. Do hlavičky správy pridá hodnotu na autentifikáciu používateľa. Následne zavolá službu `service`, uvedenú v uri požiadavky správy, vystavenú modulom WebManagement. Obdržanú odpoveď prepošle naspäť klientovi.

#### Všeobecná url pre prístup REST CodeCrutches

[http://localhost:\[CODE\\_CRUTCHES\\_PORT\]/codecrutches/\[OPERÁCIA\]](http://localhost:[CODE_CRUTCHES_PORT]/codecrutches/[OPERÁCIA])

#### Príklad url

<http://localhost:16376/codecrutches/ProjectTask>

### 2.2 Autentifikácia správ

Do správy pridáme hlavičku s menom 'Authorization' s nasledujúcou hodnotou:

'Basic [autorizačný-reťazec]'

[autorizačný-reťazec] - je zakódovaný reťazec v base64:

'[používateľské-meno]:[heslo]'

### 2.3 CodeCrutches VisualStudio Extension

Zatiaľ jediný modul, ktorý využíva služby codecrutches v DevACTs. Bližšie informácie v [Modul – Visual Studio Extension](#)

### 2.4 CodeCrutches WebManagement

Modul, ktorý vystavuje služby s požadovanou biznis funkciou. Bližšie informácie v [Modul – Web management](#)

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

## **Modul – Visual Studio Extension**

**Akademický rok:** 2016/2017

**Vedúci práce:** Ing. Karol Rástočný, Phd.

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

**Dátum poslednej zmeny:** 9.12.2016

# 1 Úvod

Tento dokument opisuje rozšírenie pre vývojové prostredie Visual Studio, ktoré slúži ako tenký klient integrujúci funkcionality CodeCrutches.

## 2 Komponenty

Hlavný komponent CodeCrutches Visual Studio Extension (skrátene VsExtension) predstavuje rozšírenie pre IDE Visual Studio, ktoré má každý používateľ nainštalovaný lokálne vo svojom prostredí. 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.

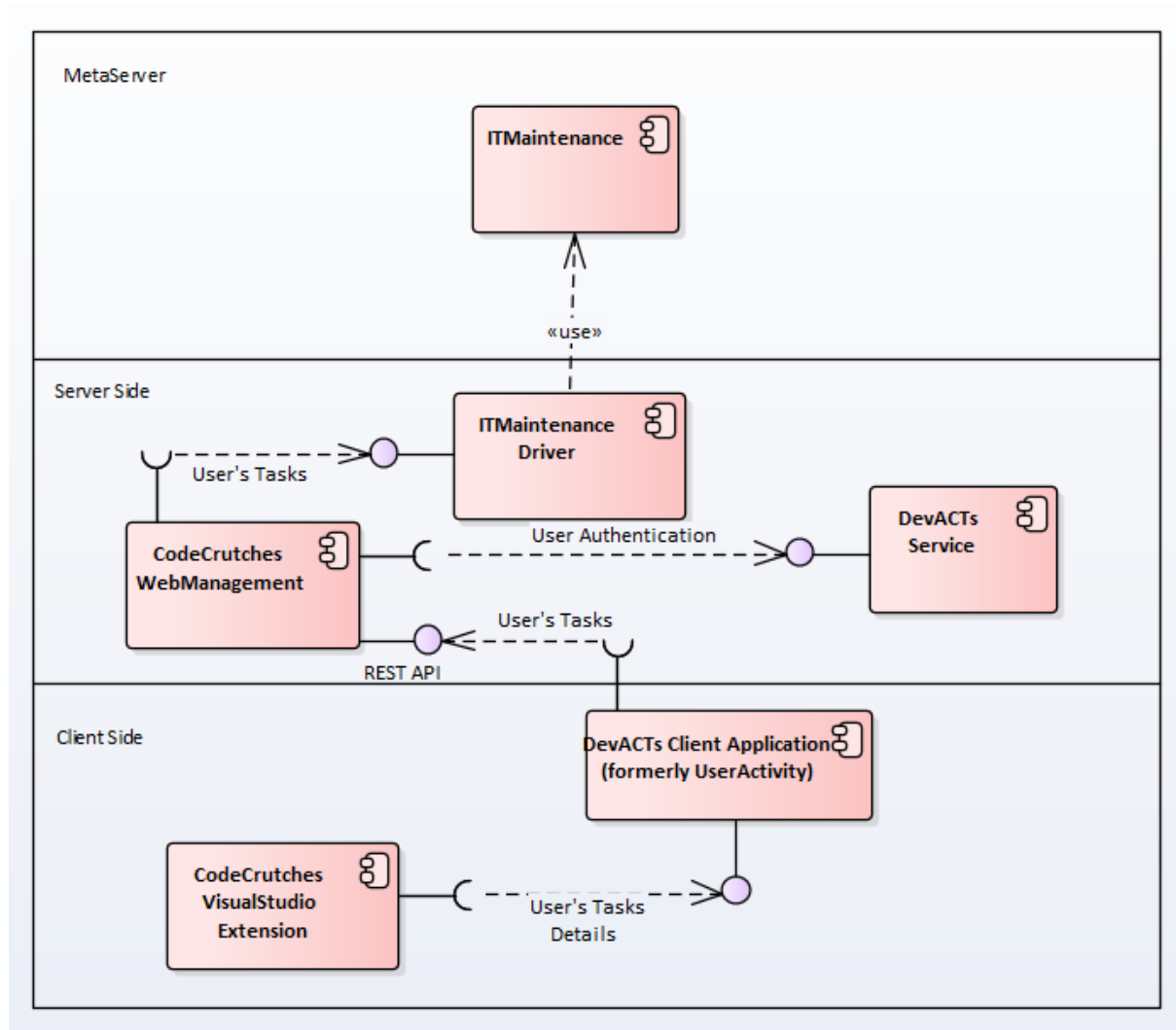


Diagram 1: Komponenty využívané rozšírením CodeCrutches VisualStudio Extension

### 2.1 Komponent DevACTs Client Application

Rozšírenie komunikuje s aplikáciou DevACTs Client (pôvodne nazývaná UserActivity), ktorú musí mať používateľ nainštalovanú. Pri jej nainštalovaní tiež zadáva svoje meno a heslo do platformy DevACTs.

Meno aktuálne prihláseného používateľa sa získava z registra s názvom `user/Name` (umiestnenom

v `HKEY_CURRENT_USER\SOFTWARE\JavaSoft\Prefs\com\gratex\perconik\useractivity\app`). Druhý parameter, ktorý potrebuje VsExtension z registra je port,

na ktorom DevACTs klient počúva jeho požiadavky. Názov registra z ktorého sa port číta je `code_crutches_port`.

Prostredníctvom tohto portu rozšírenie komunikuje s DevACTs aplikáciou, ktorá mu poskytuje metódu prostredníctvom REST API pre autentifikovanie správy a preposlanie do komponentu WebManagement.

## 2.2 Komponent CodeCrutches WebManagement

Tento komponent vystavuje REST API pre prácu s projektami a s úlohami. Volania na toto rozhrania musia byť autentifikované menom a heslom používateľa DevACT, ktoré sa pridáva v komponente DevACT Client. V súčasnosti sa využívajú operácie na:

### Práca s projektami:

- Načítanie zoznamu projektom, ku ktorým má používateľ prístup

### Práca s úlohami:

- Vytvorenie novej úlohy v ITM
- Aktualizácia stavu úlohy
- Načítanie úloh

Pre bližšie informácie o tomto module a práca ITM Maintenance viď. [Modul - WebManagement](#)

## 3 Dátový model

### 3.1 Úloha v ITM reprezentácií

Model úlohy je uložený v *ProjectTask*, pre ktorý máme implementovaný vlastný konvertor *ProjectTaskConvertor*. Táto trieda nám poskytuje štruktúru na uloženie potrebných informácií o úlohe.

#### 3.1.1 Opis polí

- *Id* – má len `get` metódu. Je to posledná časť *TaskUri*.
- *Name* – názov úlohy
- *EstimatedEndDate* = predpokladaný dátum ukončenia úlohy
- *State* - stav úlohy (bližšie opísaný v 3.1.2)
- *ProjectId* - identifikátor projektu, ku ktorému je úloha priradená
- *AssignedTo* – meno, komu je priradený task
- *RemainingWork* – koľko hodín treba do ukončenia úlohy
- *CompletedWork* – koľko hodín sa pracovalo na danej úlohe
- *OriginalEstimateWork* – odhadovaný počet hodín na začiatku
- *TaskUri* – identifikátor, ktorý linkuje issueracker
- *Author* – meno programátora, ktorý má úlohu pridelenú

#### 3.1.2 Životný cyklus

Úloha sa môže nachádzať v troch stavoch:

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

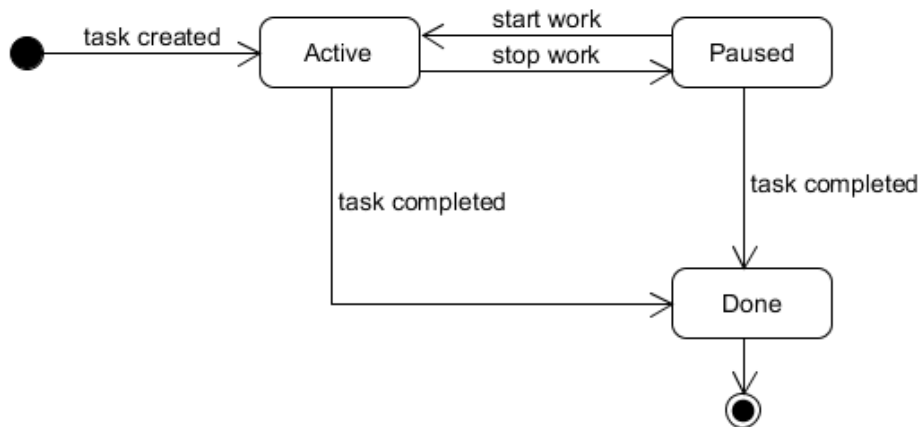


Diagram 2 Životný cyklus úlohy

## 4 Rozšírené používateľské rozhranie

### 4.1 Rozšírenie pre vzhľad okraja editora

Definovanie vlastného znaku na okraji editora v triede *CodeMarkerGlyphFactory*, ktorá implementuje rozhranie *IGlyphFactory* s metódou *GenerateGlyph*, ktorá vracia nadefinovaný výzor nového elementu používateľského rozhrania.

V triede *CodeMarkerGlyphFactoryProvider* zdefinujeme vzťah medzi našim vytvoreným znakom a značky zobrazujúcej sa na okraji editora.

*CodeMarkerTag* zabraňuje model *CodeTag*, ktorý obsahuje informácie získané z ITM značky opisujúce kód v danom súbore.

*CodeMarkerTagger* zabezpečuje že sa dané značky pre daný súbor správne zobrazia na požadovanom riadku.

*CodeMarkerTaggerProvider* potom vystavuje novo vytvorený tagger.

### 4.2 Rozšírenie pre zvýraznenie textu editora

Toto rozšírenie spolupracuje s predošlým. Po vybraní zvolenej značky má za úlohu toto rozšírenie zvýrazniť text, na ktorý sa viaže daná značka.

*CodeMarkerTagTextHighlighting* obsahuje funkcionality tohto rozšírenia; výpočet oblasti na zvýraznenie, komunikáciu s taggerom a vykresľovanie zvýraznenia.

*CodeMarkerTagTextHighlightingTextViewCreation* slúži na vytváranie highlightingu pri každom otvorení dokumentu.

*HighlightingManager* vystavuje dostupné highlightre. Každý novovytvorený highlighter sa zaregistruje u *HighlightingManager*a. *HighlighterManager* si drží informácie o každom highlighteri a zároveň vie poskytnúť aktuálny highlighter k danému aktívnemu dokumentu.

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

## **Modul – Web management**

**Akademický rok:** 2016/2017

**Vedúci práce:** Ing. Karol Rástočný, Phd.

**Členovia tímu:** Bc. Ondrej Čičkán, Bc. Šimon Dekrét, Bc. Dušan Javorník,  
Bc. Miroslav Laco, Bc. Anton Ján Vrban

**Dátum poslednej zmeny:** 13. 5. 2017



# 1 Úvod

Tento dokument obsahuje modul, ktorý je zameraný na webovú aplikáciu. Jedná sa o manažment riadenia projektov, na ktorých pracujú programátori. Tento modul vystavuje webové služby, ktoré implementujú logiku pre prácu s úlohami a projektami.

## 1.1. 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.

## 2 Komponenty

Komponent CodeCrutches WebManagement, ktorý je predmetom tejto dokumentácie, poskytuje komponentom DevACTs Client Application a CodeCrutches VisualStudio Extension REST API rozhranie pre správu úloh a projektov.

Úlohy používateľov sú ukladané na ITMaintenance serveri, ku ktorému sa pristupuje pomocou ITMaintenance Driveru. Pre autentifikáciu používateľa sa využíva komponent DevACTs Service. V aplikácii je taktiež integrovaný issue tracker – TFS, ktorý je využívaný na získavanie nových taskov, ale taktiež do ktorého sa nové tasky ukladajú a taktiež updatujú.

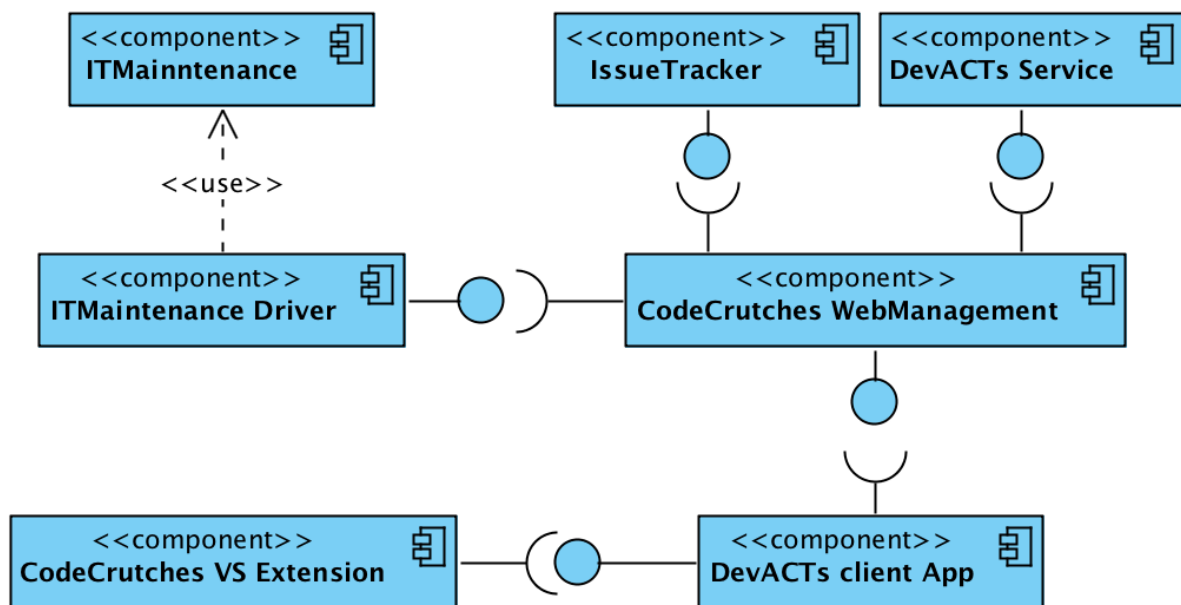


Diagram 1: Komponenty využívané a využívajúce CodeCrutches WebManagement

### 2.1 Komponent ITMaintenance Driver

Pre zobrazovanie detailov úloh používateľa, ako aj ich modifikáciu sa využíva komponent ITMaintenance Driver. Sprostredkováva komunikáciu s ITServerom, kde sú údaje o úlohách uložené. 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:

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

## 1.2 Komponent DevACTs Client Application

Aplikácia DevACTs Client komunikuje s webovým portálom prostredníctvom REST API rozhrania. Využíva metódu pre získanie úloh pre daný projekt, pridanie novej úlohy a updatovanie úlohy.

## 1.3 Komponent CodeCrutches VisualStudio Extension

Rozšírenie CodeCrutches VisualStudio Extension komunikuje s webovým portálom prostredníctvom komponentu DevACTs Client Application.

# 3 REST API rozhranie

Webový portál CodeCrutches WebManagement poskytuje REST API rozhranie pre prístup k jeho funkciám z iných komponentov.

## 1.2. api/ProjectTask

**Controller:** CodeCrutches.WebManagement.Controllers.ProjectTaskController

Toto rozhranie poskytuje metódy pre správu úloh a projektov. Momentálne je dostupné na adrese <https://devacts.fiit.stuba.sk:8443/codecrutches/>. Výsledky vracia vo formáte JSON. Momentálne poskytuje metódy pre:

- **Načítanie všetkých úloh podľa projektov** (vyžadovaná autentifikácia):

request: GET api/ProjectTask/GetTasks/[projectId]

odpoveď (príklad) – podľa modelu CodeCrutches.Common.Model.ProjectTask:

```
{
  "Id": "5860",
  "Author": "Pavol Mrkva",
  "Created": "2016-11-29T22:27:17.063Z",
  "EstimatedEndDate": "2016-11-29T00:00:00",
  "Name": "HelloTask1",
  "ProjectId": "5860",
  "State": "1",
  "URI": "http://perconik.fiit.stuba.sk/ITM/c6e32505-63c5-46c3-8bcc-c0de68c015ce",
  "AssignedTo": "xpavol",
  "Type": "itm:ProjectTaskInformationTag",
  "HasVersion": "0",
  "RemainingWork": "3.0",
  "CompletedWork": "1.0",
  "OriginalEstimateWork": "4.0",
  "TaskUri": "https://tfs.fiit.stuba.sk:8443/tfs/Students/Projects/DevACTs/_workitems/edit/5860"
}
```

- **pridanie novej úlohy** (vyžadovaná autentifikácia):

request: POST api/ProjectTask – podľa modelu

CodeCrutches.Common.Model.ProjectTask:

```

{
  "Author": "xdekret",
  "EstimatedEndDate": "2016-11-29T00:00:00",
  "Name": "HelloTask1",
  "ProjectId": "111111222223333",
}

```

Task sa najprv ulozi to Issue trackera a v prípade správneho uloženia sa uloží aj do ITM.  
odpoveď:

- 200 OK v prípade pridania úlohy,
- 401 Unauthorized v prípade neprihlásenia,
- 403 Forbidden v prípade nedostatočného oprávnenia na Issue tracker

- **aktualizovanie stavu úlohy**

request: PUT api/ProjectTask – podľa modelu

CodeCrutches.Common.Model.ProjectTask:

Metóda dostane v parametri objekt ProjectTask. Najprv sa updatuje task v Issue trackeri a v prípade úspechu sa updatuje aj v ITM.

Odpoveď:

- 200 OK v prípade aktualizovania úlohy,
- 401 Unauthorized v prípade neprihlásenia,
- 403 Forbidden v prípade nedostatočného oprávnenia

- **načítanie projektov, ku ktorým má používateľ prístup:**

request: GET api/ProjectTask/UserProjects/[username] (kde [username] predstavuje používateľské meno používateľa

odpoveď (príklad) – podľa modelu CodeCrutches.Common.Model.ProjectModel:

```

{
  {
    "Id": "58e62a5902ca821d48b897a0",
    "Name": "ProjectTest1",
    "Description": "",
    "ProjectUserList": [
      {
        "DevactsName": "xpavol",
        "Role": 0
      },
      {
        "DevactsName": "xlaco",
        "Role": 2
      }
    ]
  },
  "MongoConnectionModel": "58e62a9602ca8219d4930234"
},
{
  "Id": "5900b2aa02ca81132cdc9faa",
  "Name": "ProjectTest2",
  "Description": null,
  "ProjectUserList": [
    {
      "DevactsName": "xpalo",
      "Role": 0
    }
  ]
},
  "MongoConnectionModel": "5900b3d902ca8125488ed9a6"
}

```

- **načítanie úloh patriacich všetkým používateľom**  
request: GET api/ProjectTask  
odpoveď (príklad) – podľa modelu CodeCrutches.Common.Model.ProjectTask

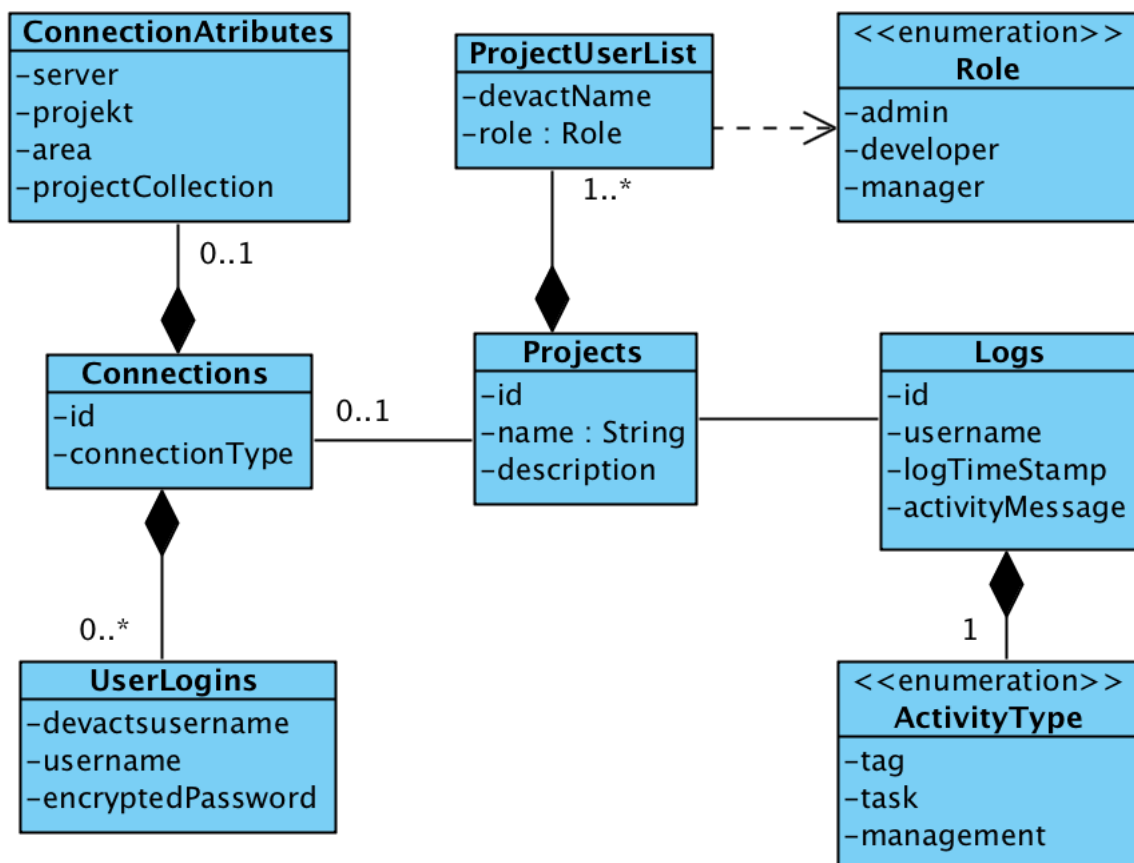
## 4 Dátový model

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

- Admin
- Manager
- Developer

### 1.3. Projekt

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



Obrázok 1: Dátový model projektu

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. Každý participant na projekte má pridelenú rolu.

Ku každému projektu je možné definovať pripojenie k issue trackeru (*Connections*). Podporované pripojenie je zatiaľ len k systému TFS (*TfsConnection*). Heslo pre každého používateľa je zahashované. Kód na zahashovanie bol prevzatý z webovej stránky: <http://www.selamigungor.com/post/7/encrypt-decrypt-a-string-in-csharp>

Projekty majú v priradené aj logy, ktoré sa zobrazujú vo web managemente. *Logs* môže byť typu:

- tag,
- task,
- management

## 5. Funkcie

### 2.1. 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

**Services:** CodeCrutches.WebManagement.Services

Pre načítavanie zoznamov úloh a prácu so štruktúrou *ProjectTask* 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

### 2.2. Prihlasovanie pomocou DevACTs účtu

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

*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*.

### 2.3. **ITM Proxy**

Úlohou proxy služby je umožňovať pripojenie na ITM prostredníctvom URL načítanej z konfigurácie DevActs projektu.

*Obslužné triedy*

**Service:** CodeCrutches.WebManagement.Services.DevActsService,  
CodeCrutches.WebManagement.Services.Proxy

Prístup ku konfigurácii DevActs poskytuje externý projekt Core.CentralServices. Na načítanie ITM EndPointu sa využíva metóda triedy DevActsService:

- GetItmEndPoint()

Pripojenie na ITM je implementované v triede Proxy pomocou funkcie:

- ConnectToITM()

### 2.4. **Pripojenie do TFS**

Úlohou funkcie je poskytnúť možnosť napojenia projektu, vytvoreného v module WebManagement na projektový manažment TFS.

*Obslužné triedy*

**Model:** CodeCrutches.Common.Model.ConnectionModel,  
CodeCrutches.Common.Model.TfsConnectionModel

**Data:** CodeCrutches.Common.Data.ConnectionManager

**Service:** CodeCrutches.Common.Service.IssueTrackerConnector

**Controller:** CodeCrutches.WebManagement.Controllers.ProjectController

**View model:** CodeCrutches.WebManagement.Model.SettingsIssueTrackersModel

**View:** CodeCrutches.WebManagement.Views.SettingsIssueTrackers

Pre uloženie dát, potrebných pre pripojenie do TFS poskytuje trieda CodeCrutches.Common.Service.IssueTrackerConnector metódu pre pridanie (príp. aktualizovanie) týchto dát:

- NewProjectConnectionToMongo()

Pre prístup k dátam pre pripojenie do TFS poskytuje pre id projektu táto trieda metódu:

- GetConnection()

Pre otestovanie konekcie na TFS slúži metóda:

- TestConnection()

Daná metóda sa vyskúša pripojiť na TFS za pomoci poskytnutých údajov a vráti výsledok pokusu - či sú všetky údaje v poriadku.

Ďalšia metóda, ktorú poskytuje táto funkcia slúži na získanie všetkých neukončených úloh v TFS týkajúcich sa nášho tímu:

- GetPendingTasks()

Metóda, ktorá vracia úlohy z TFS pre zadaného užívateľa (resp. užívateľov) má názov:

- GetTasksForUsers()

## **6. Knižnice a externé projekty**

### **3.1. Core.CentralServices**

Balíček služieb Core.CentralServices je importovaný prostredníctvom projektu DevActs a poskytuje informácie o jeho konfigurácii.

### **3.2. 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

Ilkovičova 2, 842 16 Bratislava 4

## Používateľská príručka – DevACT client

**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

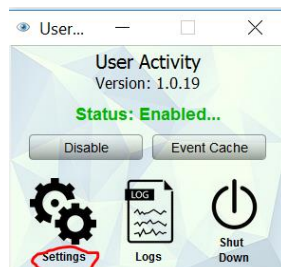


# Úvod

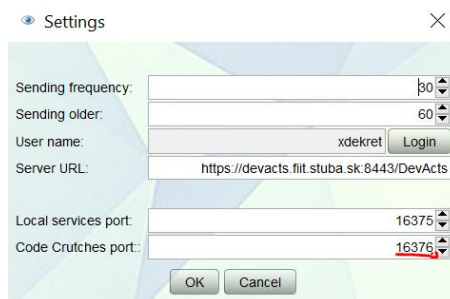
Tento dokument obsahuje opis používateľskej príručky pre DevACTs klienta.

## Zmena portu pre prijímanie požiadaviek z VS Extension

1. Používateľ po otvorení aplikácie klikne na položku Settings



2. Používateľ klikne zmení číslo v políčku Code Crutches port a stlačí OK.



Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

# Používateľská príručka – Visual Studio Extension

**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 Opis používania Visual Studio Extension

## 1.1 Predpoklady

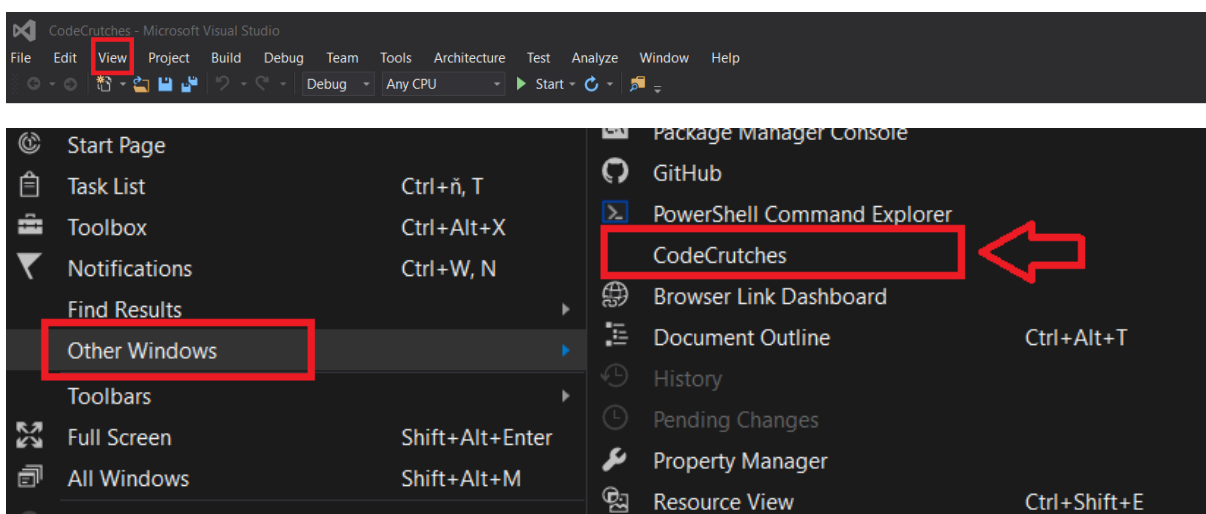
Pre plnohodnotné používania rozšírenia musíte vykonať nasledujúce kroky:

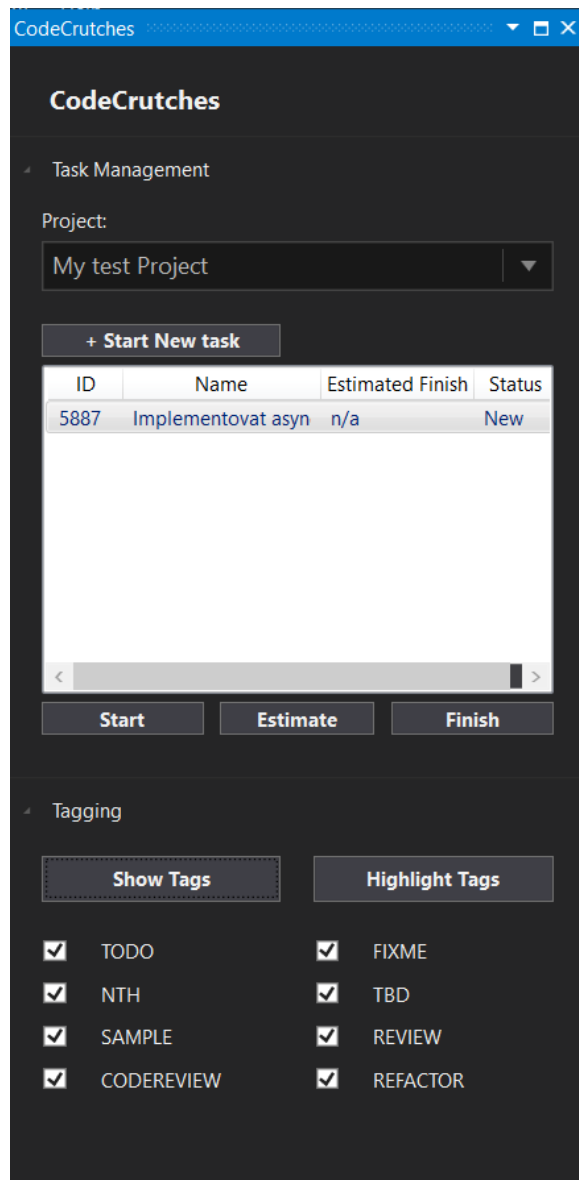
1. Nainštalovať program User Activity (nemusíte si inštalovať jeho plugin do Visual Studia alebo do chromu) – [stiahnuť TU](#)
2. Prihlásiť sa na stránku CodeCrutches <https://devacts.fiit.stuba.sk:8443/CodeCrutches/> Bude potrebné vytvoriť projekt a nastaviť prístupové údaje na pripojenie na TFS (pre viac informácií pozrite Používateľskú príručku pre WebManagement)
3. Nastaviť alias pre TFS na <https://devacts.fiit.stuba.sk:8443/DevACTs/Manage>. Ísť na sekciu **Aliases**. Vyplňte nasledujúce polia:  
*New URL:* adresu tfs napr. (<https://tfs.fiit.stuba.sk>)  
*New Alias:* alias vo vašom profile v tfs (napr. Jozef Mrkva)  
Potom stlačte **Add**.



## 1.2 Opis používateľského rozhrania

Po nainštalovaní doplnku sa objaví v **View** -> **Other Windows** nová položka **CodeCrutches**. Po kliknutí na túto možnosť sa vám zobrazí panel CodeCrutches (Formulár 1), ktorý si môžete pripevniť kdekoľvek vo Visual Studiu.





Formulár 1 CodeCrutches panel nástrojov

Panel obsahuje dve hlavné časti:

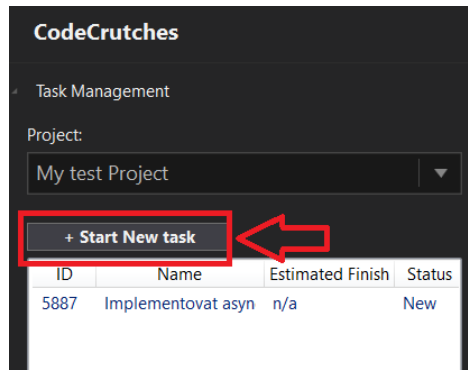
- **Task Management** - slúži na manažment úloh a na evidovanie odpracovaných hodín.
- **Tagging** – slúži na zapnutie/vypnutie meta-značiek a zvýraznenia textu. Ponúka dodatočné možnosti filtrovania typov meta-značiek.

## 2 Opis používateľských scenárov

### 2.1 Vytvorenie novej úlohy

Uistite sa, že aplikácia User Activity je korektne spustená a máte nainštalovaný plugin User Activity a CodeCrutches vo Visual Studio. Projekt, pre ktorý chcete vytvoriť úlohu musí byť cez CodeCrutches korektne napojený na Issue Tracker. V DevACTs nastaveniach musíte mať pre daný Issue Tracker nastavený správny Alias pre Vaše DevACTs menu. Pre zaevidovanie začiatku práce na novej úlohe vykonajte tieto kroky:

1. Vyberte v panely CodeCrutches možnosť „Start New task“:



2. Zobrazí sa formulár pre vytvorenie novej úlohy (Formulár 2).

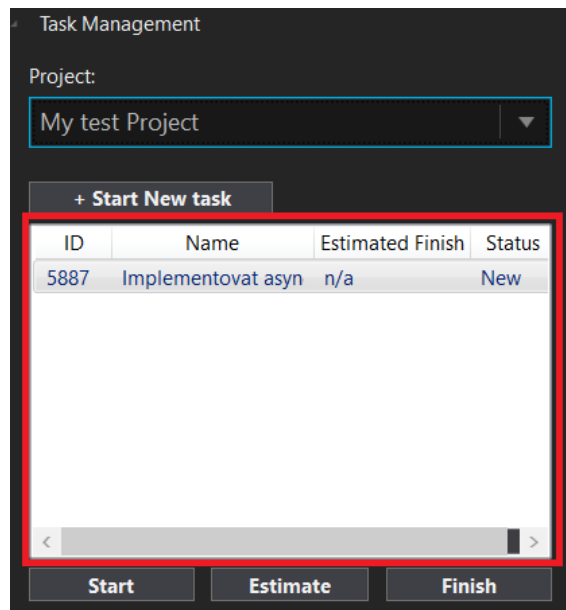
Formulár 2 Zaevidovanie novej úlohy

3. Vyplňte polia:

- a. **Task Name** – čo najvýstižnejší samo-opisný názov novej úlohy
  - b. **Project** – projekt, pod ktorý sa má novo vytvorená úloha zaradiť
  - c. **Estimated task finishu** – predpokladaný dátum ukončenia úlohy (predvolený je aktuálny dátum)
  - d. **Original time estimate** – predpokladaný počet hodín na dokončenie tejto úlohy. Poznámka- odhad predpokladaného ukončenia má iba informatívny charakter a je možné ho dodatočne meniť.
4. Kliknite na tlačidlo „**Submit**“. Pozor- nová úloha sa vytvorí v Issue Tracker-i napojenému k zvolenému projektu.
  5. Budete informovaný o výsledku zaevidovania začatia novej úlohy pomocou okna so správou o úspechu/neúspechu začatia novej úlohy

## 2.2 Zobrazenie a manažment Vašich úloh

Zoznam vašich a nepriradených úloh z TFS sú zobrazené v CodeCrutches panely v časti Task Management.



1. V zozname sú zobrazené úlohy pre projekt vybraný v poli Project, ktoré máte priradené. Môžete kliknúť na ľubovoľnú úlohu a previesť s ňou jednu z aktivít nižšie. Úlohy môžu mať nasledujúce stavy:
  - New – nezačatá úloha priradená Vám v Issue Tracker-i napojenom na zvolený projekt
  - Active – aktívna úloha na ktorej práve pracujete
  - Paused – prerušená úloha, ktorá už bola v minulosti v stave Active
  - Done – ukončená úloha (nezobrazuje sa v zozname)
2. Pre zobrazenie úloh k inému projektu vyberte tento projekt z poľa Project a počkajte na načítanie úloh.

### 2.2.1 Začatie úlohy v stave „New“

1. Po zvolení úlohy kliknite na tlačidlo „Start“ .
2. Zobrazí sa formulár (Formulár 2) pre vytvorenie novej úlohy s vyplneným názvom úlohy a projektom, pod ktorý úloha spadá.
3. Vyplňte polia „Estimated task finish“ a „Original time estimate (hours)“.
4. Kliknite na tlačidlo „Submit“.
5. Budete informovaný o výsledku operácie prostredníctvom dialógového okna. V prípade neúspechu skontrolujte nastavenia popísané na začiatku podkapitoly a pokus opakujte. V prípade úspechu bude aktuálne aktívna úloha pozastavená a začatá úloha bude v stave Active.

### 2.2.2 Pozastavenie úlohy v stave „Active“

1. Po zvolení úlohy kliknite na tlačidlo „Pause Task“.
2. Zobrazí sa formulár (Formulár 3) pre zadanie hodín spojených s úlohou s predvyplneným názvom úlohy, stráveným časom prebratým z Issue Tracker-a a s časom potrebným do ukončenia úlohy prebratým z Issue Tracker-a. Časy sú editovateľné.

Formulár 3 Okno na zmenu zostávajúceho a stráveného času na úlohe

3. Upravte časy podľa potreby.
4. Kliknite na tlačidlo „Submit“.
5. Budete informovaný o výsledku operácie prostredníctvom dialógového okna. V prípade neúspechu skontrolujte nastavenia popísané na začiatku podkapitoly a pokus opakujte. V prípade úspechu bude vybratá aktívna úloha pozastavená a budú upravené dotknuté hodiny.

### 2.2.3 Aktivovanie úlohy v stave „Paused“

1. Po zvolení úlohy kliknite na tlačidlo „Activate Task“.
2. Úloha sa aktivuje. Pozn. – naraz môžete mať aktívnu nanajvýš 1 úlohu, preto ak máte v tomto momente nejakú úlohu v stave „Active“, tak sa pozastaví.

### 2.2.4 Zmena odhadovaného dátumu ukončenia úlohy

1. Po zvolení úlohy kliknite na tlačidlo „Estimate“.
2. Zobrazí sa formulár (Formulár 4) s predvyplneným súčasným odhadovaným dátumom ukončenia zvolenej úlohy.

Formulár 4 Okno na zmenu predpokladaného dátumu ukončenia úlohy

3. Upravte odhadovaný dátum ukončenia podľa potreby.
4. Kliknite na tlačidlo „Submit“.
5. Budete informovaný o výsledku operácie prostredníctvom dialógového okna. V prípade neúspechu skontrolujte nastavenia popísané na začiatku podkapitoly a pokus opakujte. V prípade úspechu bude odhadovaný dátum ukončenia úlohy aktualizovaný

### 2.2.5 Ukončenie úlohy

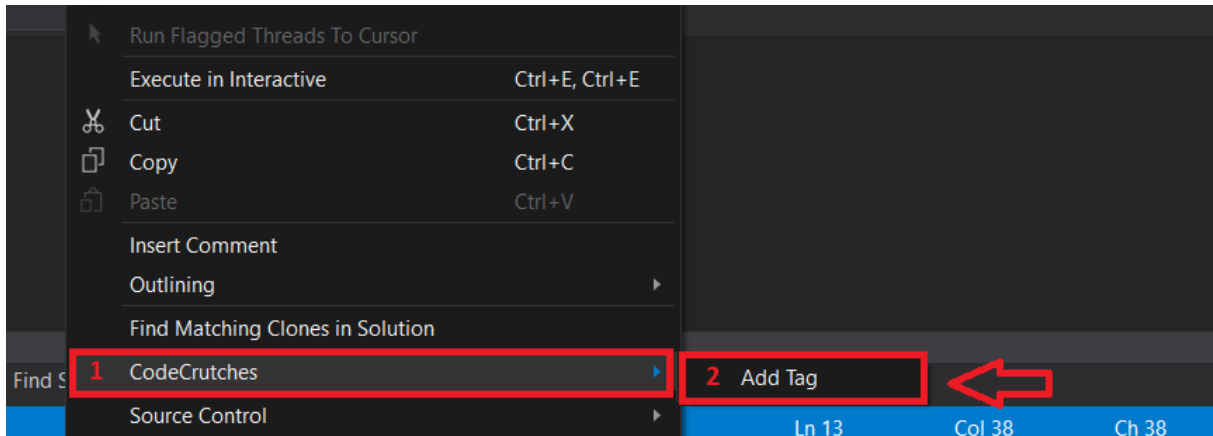
1. Po zvolení úlohy kliknite na tlačidlo „Finish Task“.
2. Zobrazí sa formulár (Formulár 3) pre zadanie hodín spojených s úlohou s vyplneným názvom úlohy, stráveným časom prebratým z Issue Tracker-a a s časom potrebným do ukončenia úlohy prebratým z Issue Tracker-a. Strávený čas na úlohe je editovateľný.
3. Upravte strávený čas podľa potreby.

4. Kliknite na tlačidlo „Submit“.
5. Budete informovaný o výsledku operácie prostredníctvom dialógového okna. V prípade neúspechu skontrolujte nastavenia popísané na začiatku podkapitoly a pokus opakujte. V prípade úspechu bude vybratá úloha ukončená a odobratá zo zoznamu.

## 2.3 Manažment meta-značiek k zdrojovému kódu

### 2.3.1 Vytvorenie meta značky

1. Označte požadovanú časť zdrojového kódu a stlačte pravé tlačidlo pravej myši
2. Otvorí sa kontextové menu s položkou CodeCrutches



3. Kliknite na položku CodeCrutches -> Add Tag
4. Zobrazí sa Formulár 5, na pridanie textu a typu meta značky. V časti Code je zobrazený označený zdrojový kód.



**Add Tag**

Code:

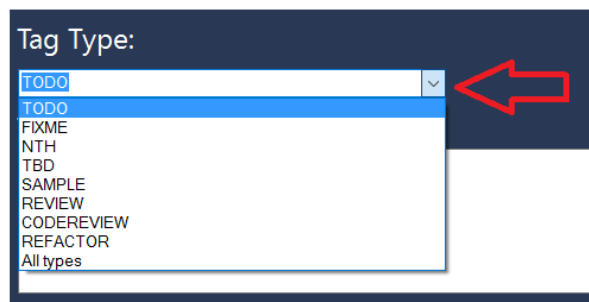
```
16 public const string TaskState = "TaskState";
```

Tag Type:

Text:

Formulár 5 Okno na vytvorenie novej meta značky

- Kliknutím na výberové okno pod Tag type, zvolíte typ značky. A do textového pola Text napíšete komentár ku značke.



- V poli *Text* môžete zadať vlastný komentár k označenému zdrojovému kódu. Je implementovaný automatické dopĺňanie textu pri zadaní znaku „@“ pre dopnenie mena člena priradeného v danom projekte, alebo pri zadaní znaku „#“ na zadanie artefaktu ako napríklad *#Task(cislo\_tasku)*.
- Stlačte tlačidlo Submit

## 2.4 Zobrazenie meta-značiek

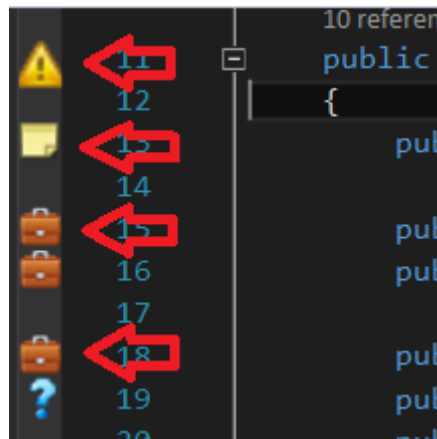
### 2.4.1 Vizuálna reprezentácia typov značiek

 *TODO*

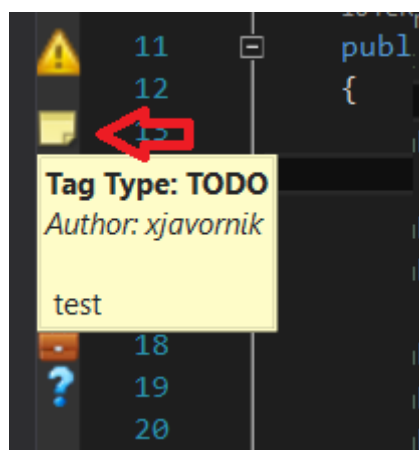
-  *FIXME*
-  *NTH (Nice to have)*
-  *TBD (To Be Done)*
-  *SAMPLE*
-  *REVIEW*
-  *CODEREVIEW*
-  *REFACTOR*
-  *TAG GROUP*


### 2.4.2 Zobrazenie obsahu viacerých meta-značiek na jednom riadku

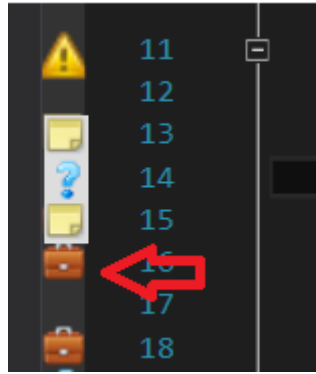
1. Súbory, ktoré obsahujú značky majú zobrazené ikony na ľavom okraji textového editora pre Visual Studio



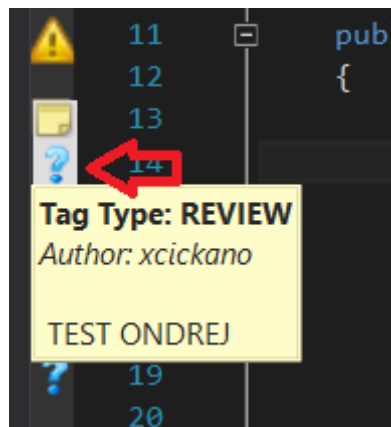
2. Kliknutím na ikonu značky sa zobrazí okno obsahujúce text meta-značky. Toto okno zmizne po druhom kliku, alebo po 5 sekundách.



3. Po kliknutí na typ značky TAG GROUP () sa zobrazí pri značke kontextové okno, ktoré obsahuje všetky značky vyskytujúce sa na danom riadku.

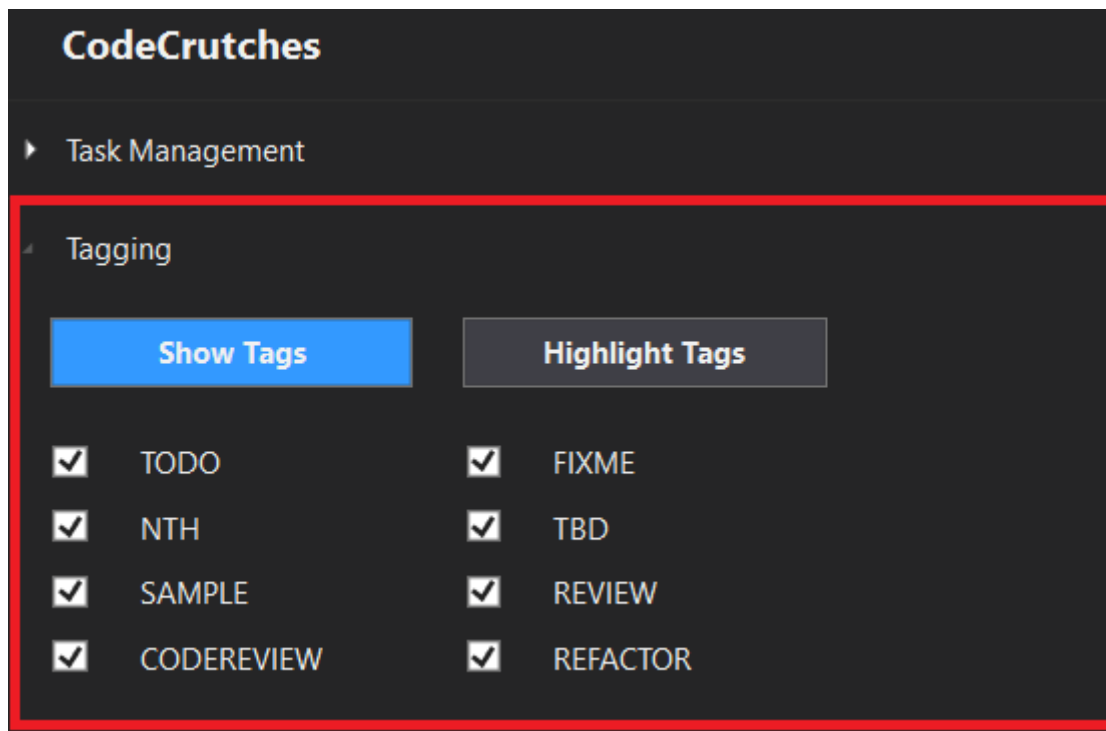


4. Po kliknutí na značku v kontextovom okne sa zobrazia informácie o meta-značke.



## 2.5 Nastavenie zobrazovania značiek

V CodeCrutches panely nástrojov v sekcii **Tagging** sú možnosti pre filtráciu a zobrazovanie značiek a zvýrazňovania textu.



### 2.5.1 Vypnutie/Zapnutie zobrazovania značiek

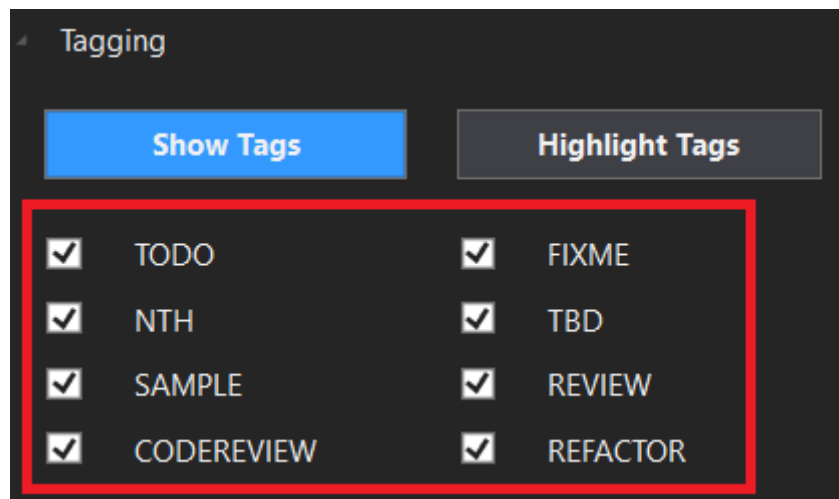
1. Stlačením tlačidla *Show Tags* vypnete alebo zapnete zobrazovanie ikoniek na okraji textového editora vo Visual Studiu.

### 2.5.2 Vypnutie/Zapnutie zvýrazňovania textu

1. Stlačením tlačidla *Highlight Tags* vypnete alebo zapnete zvýraznenie textu v textovom editore. Zvýraznený je kus textu ku ktorému je priradená daná meta-značka.

### 2.5.3 Filtrovanie značiek

1. V sekcii *Tagging* v panely CodeCrutches sú checkboxy obsahujúce podporované značky.



2. Kliknutím na check-box povolíte alebo zakážete daný typ značky.
3. Po kliknutí na akýkoľvek check-box sa automaticky zobrazia len povolené značky (platí pre ikony na okraji obrazovky ako aj pre zvýraznenie textu)

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

# Používateľská príručka – Web management

**Akademický rok:** 2016/2017

**Vedúci práce:** Ing. Karol Rástočný, Phd.

**Členovia tímu:** Bc. Ondrej Čičkán, Bc. Šimon Dekrét, Bc. Dušan Javorník  
Bc. Miroslav Laco, Bc. Anton Ján Vrban

**Dátum poslednej zmeny:** 15. 5. 2017

# 1 Úvod

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

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

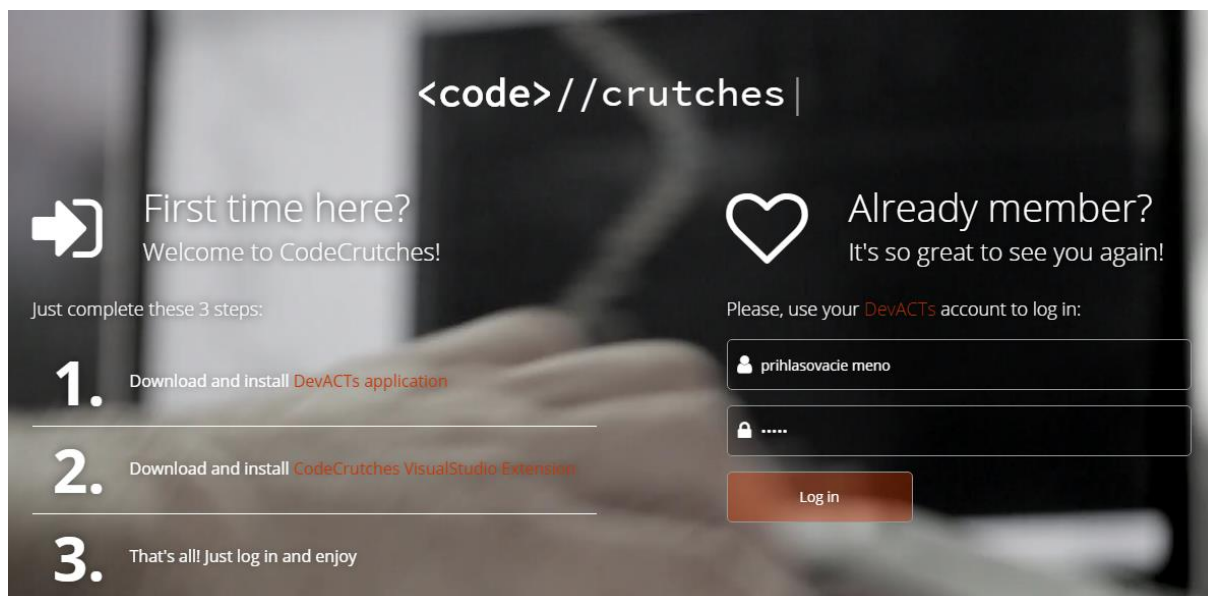
Je dostupné na adrese <http://devacts.fiit.stuba.sk:8080/CodeCrutches>, prípadne <https://devacts.fiit.stuba.sk:8443/CodeCrutches>.

## 2 Predpoklady

Pre prístup do webového rozhrania CodeCrutches je potrebné mať vytvorené konto na platforme DevACTs s povoleným prístupom do modulu CodeCrutches. Pre jeho nastavenie vykonajte tieto kroky:

1. Prihláste sa do platformy DevACTs s využitím prihlasovacích údajov z AIS na adrese: <https://devacts.fiit.stuba.sk:8443/DevACTs/Account/Login> (pozn.: Pre nastavenie prístupu do modulu CodeCrutches musíte mať pridelené administrátorské práva na platforme DevACTs. Ak ich nemáte, prosím kontaktujte správcu platformy DevACTs.)
2. V hornom menu vyberte možnosť „User management“
3. V ľavom menu vyberte možnosť „CodeCrutches“
4. V tabuľke vpravo nájdite svoje prihlasovacie meno a v pravom stĺpci vyberte z možností zobrazených po kliku na šípku možnosť „admin“.

V prípade, že konto DevACTs s prístupom na CodeCrutches už máte vytvorené, môžete sa prihlásiť na adrese <http://devacts.fiit.stuba.sk:8080/CodeCrutches>, prípadne <https://devacts.fiit.stuba.sk:8443/CodeCrutches>.



Na prihlasovacej stránke tiež nájdete odkazy na stiahnutie DevACTs aplikácie a CodeCrutches VisualStudio Extension, ktoré je pre plnohodnotné používanie webovej aplikácie CodeCrutches potrebné (viď. Používateľskú príručku pre CodeCrutches VisualStudio Extension).

## 2.1 Základný popis používateľského rozhrania

### 2.1.1 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.

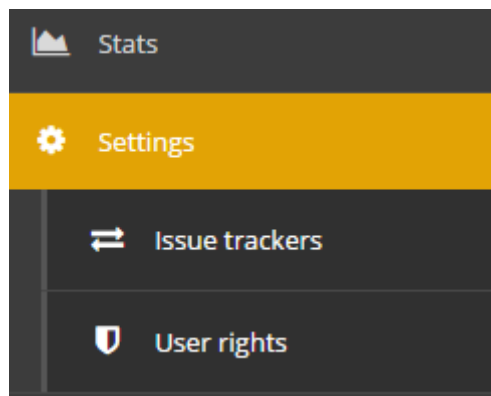
### 2.1.2 Hlavné zobrazenie

The screenshot displays a dashboard interface. At the top, there is a navigation bar with a logo (5), a breadcrumb trail (4: Home > Project 1 > Dashboard), and a user profile dropdown (2: xrbana). On the left, a sidebar menu (1) contains options for Dashboard, Tasks, Team Members, Stats, and Settings. The main content area features three summary cards: '20/50 Hours completed', '6/14 Tasks finished', and '5/7 Members active last 24h'. Below these is a section titled 'Active Tasks' with a search bar and a table of task entries.

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	xickano	06.11.2016 20:53:00	piatok, 11. novembra 2016	Paused
najnovsi task	xickano	06.11.2016 21:15:04	nedela, 6. novembra 2016	Paused
new Task	xickano	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

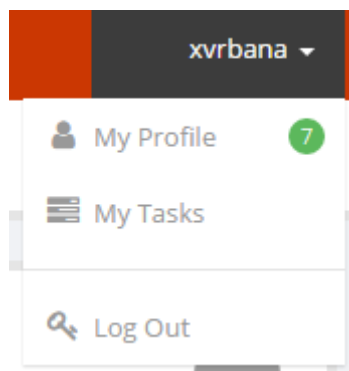
#### Pod-položky v hlavnom menu

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



#### Používateľské menu

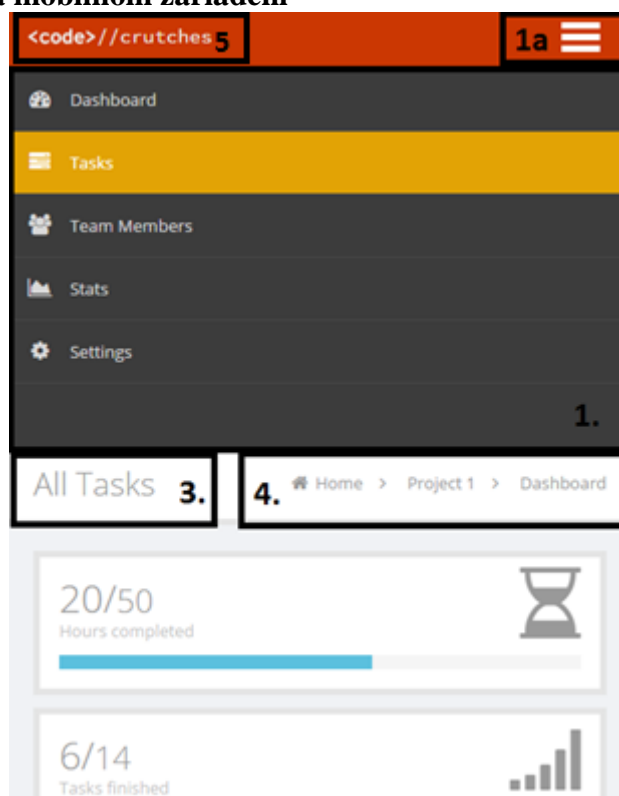
Používateľské menu (označené číslom 2) je prístupné po kliknutí na prihlasovacie meno používateľa.



*Návrat na úvodnú stránku*

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

### 2.1.3 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.

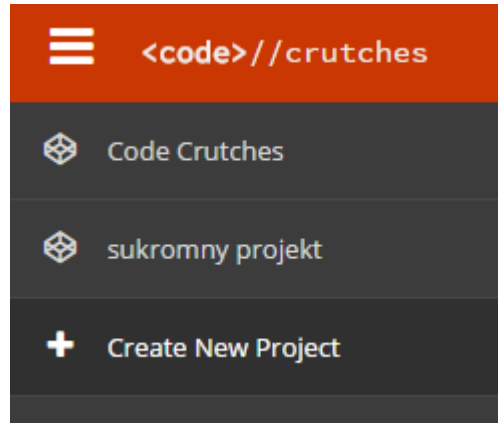


## 3 Projekt

### 3.1 Vytvorenie nového projektu

Pre pridanie nového projektu do aplikácie CodeCrutches je potrebné byť vo webovej aplikácii CodeCrutches prihlásený. 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. Názov projektu („Name“) môže byť ľubovoľný, no nesmie zostať nevyplnený. Vyplnenie položky popis projektu („Description“) 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 jedného prideleného používateľa - Vás. Práva k projektu máte predvolene nastavené ako administrátor.

---

## Create New Project

Enter your new project details

Name

**test project**

Description

**test project description**

**Create Project**

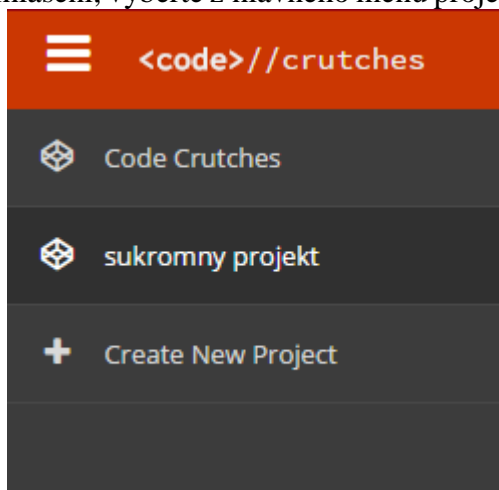


## 4 Manažment projektu

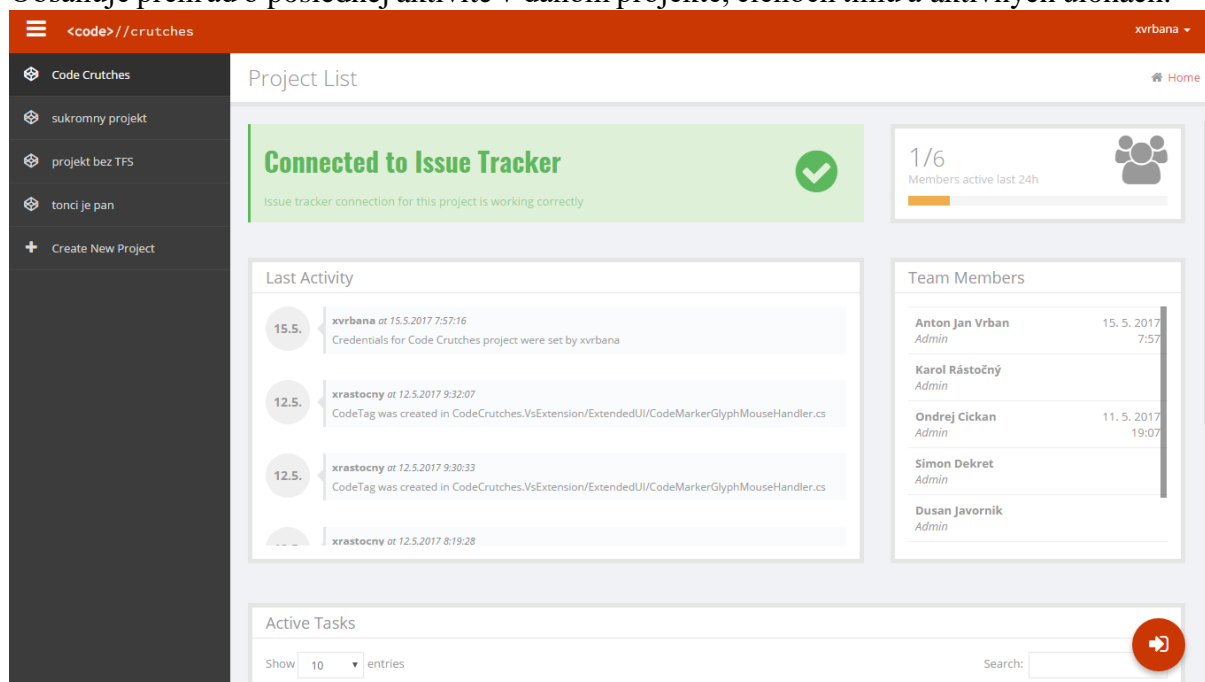
### 4.1 Výber aktívneho projektu a jeho manažment

#### 4.1.1 Výber projektu a zobrazenie jeho detailov

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



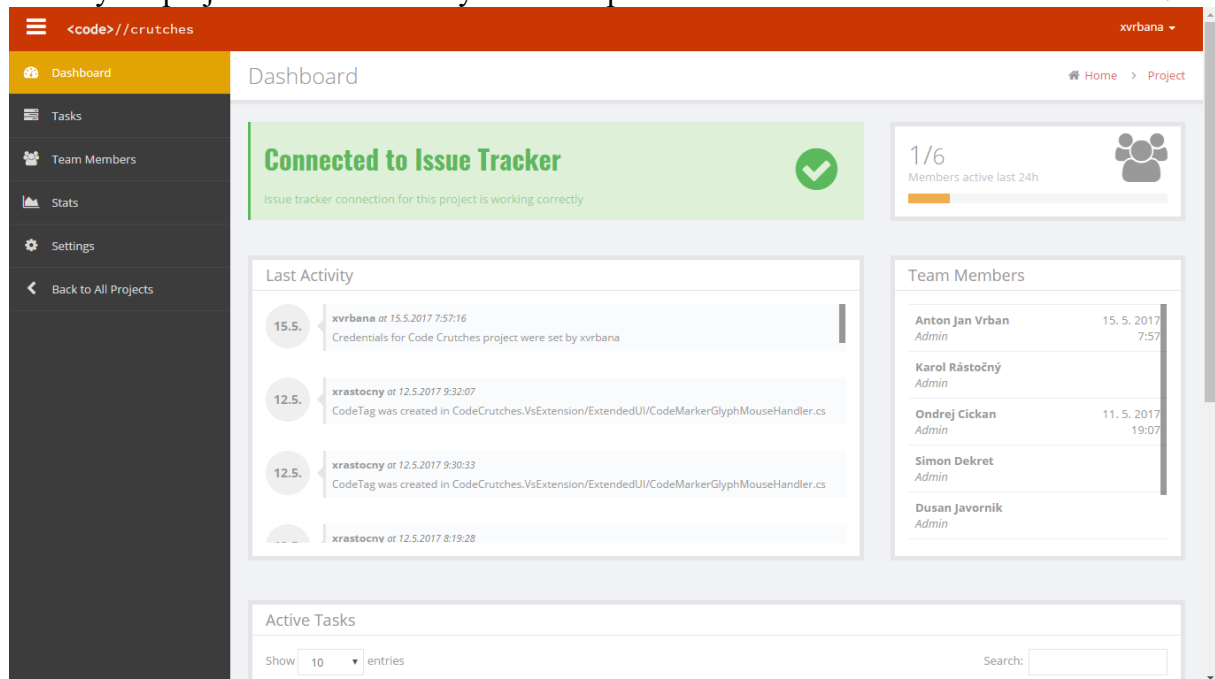
2. Vybráním projektu z menu sa zobrazí v okne vpravo jeho nástenka („Dashboard“). Obsahuje prehľad o poslednej aktivite v danom projekte, členoch tímu a aktívnych úlohách.



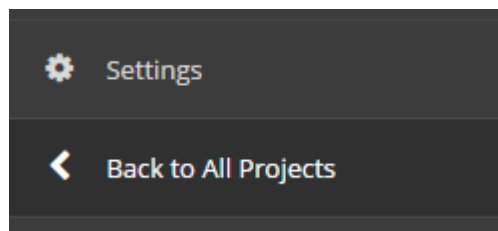
3. Projekt označíte ako aktívny kliknutím na kruhovú ikonku v pravom dolnom rohu.



- Po výbere aktívneho projektu vstúpite do jeho správy. Jednotlivé akcie súvisiace so zvoleným projektom môžete vykonávať prostredníctvom hlavného menu vľavo:



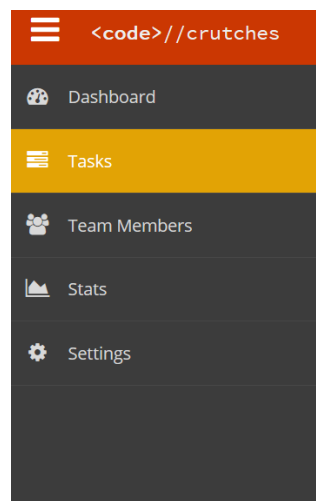
- Pre vrátenie sa na zoznam projektov vyberte z hlavného menu možnosť „Back to All Projects“.



#### 4.1.2 Zobrazenie úloh projektu

Webová aplikácia CodeCrutches umožňuje sledovať stav riešenia úloh k projektu, pričom tieto úlohy môžu byť vytvorené prostredníctvom rozšírenia CodeCrutches VisualStudio Extension, prípadne priamo v používanom Issue Trackeri, ktorý je spárovaný s aktívnym projektom (viď časť 4.3).

- V hlavnom menu kliknite na položku *Tasks*.



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

3. V menu nad tabuľkou s úlohami je možné filtrovať úlohy podľa ich stavu:

- **In Progress** – zobrazí úlohy so stavom „Active“ (v momentálnom čase sa na nich pracuje) a „Paused“ (úlohy sú v stave riešenia – boli začaté a ešte neukončené)
- **Finished** – zobrazí úlohy so stavom „Done“ (vyriešené, uzavreté úlohy)
- **All** – zobrazí všetky úlohy

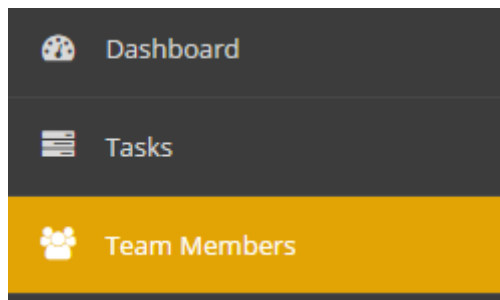
4. Zoznamy úloh je možné filtrovať pomocou vyhľadávania umiestneného nad zoznamom. Vyhľadávať je možné akýkoľvek reťazec (aj jeho časť) a vyhľadávanie prebieha nad všetkými stĺpcami tabuľky priamo pri písaní vyhľadávacej frázy:

## 4.2 Používatelia projektu

Projekt môže mať viacero používateľov, pričom títo používatelia môžu mať rôzne oprávnenia (administrátor, developer, manažér).

### 4.2.1 Zobrazenie zoznamu používateľov projektu

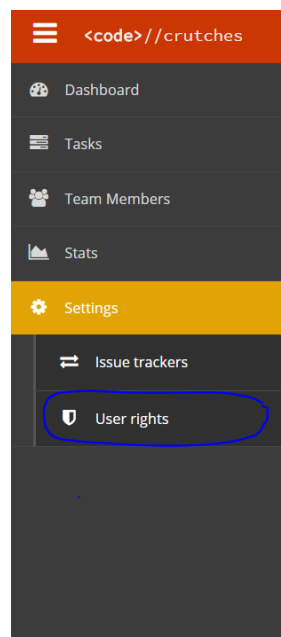
Zoznam používateľov projektu, spolu s ich kontaktnými údajmi a záznamom aktivity nájdete po kliknutí na položku Team Members v hlavnom menu.



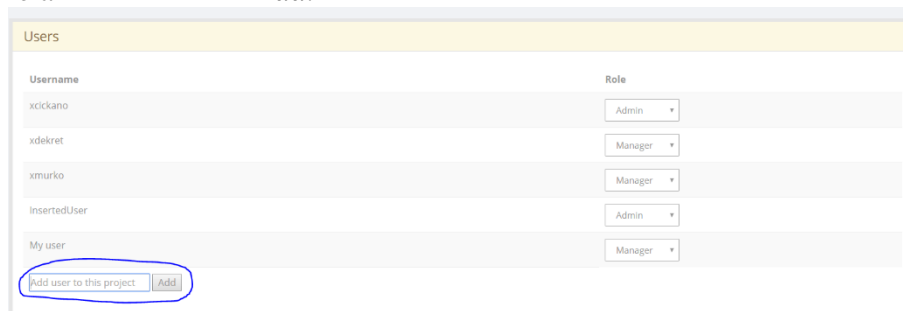
#### 4.2.2 Pridanie nového používateľa k projektu

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*.



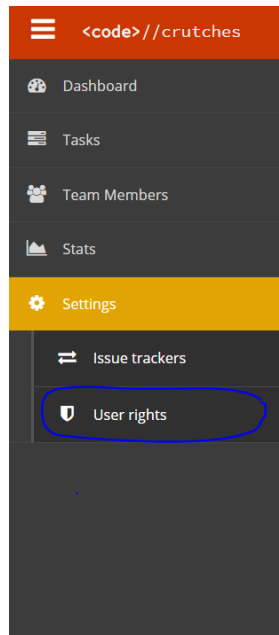
3. Na konci zoznamu používateľov je možnosť napísania mena nového používateľa. Toto pole vyplne a stlačte tlačidlo *Add*.



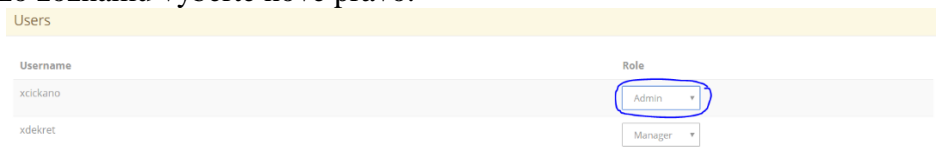
#### 4.2.3 Zmena roly používateľa

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.



#### 4.3 Nastavenie prepojenia na aplikáciu pre projektový manažment

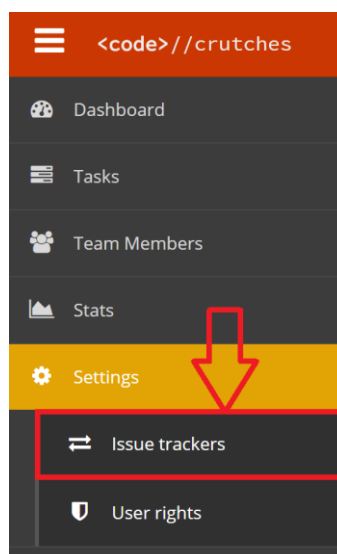
Webová aplikácia CodeCrutches umožňuje spárovanie projektu s používaným issue trackerom a zabezpečiť tak synchronizáciu úloh.

V súčasnosti je podporované prepojenie na issue tracker Microsoft Team Foundation Server (TFS). Projekt môže byť prepojený iba s jedným issue trackerom.

##### 4.3.1 Nastavenie prepojeného issue trackera

Nastavenie prepojeného issue trackera môže urobiť len administrátor projektu. Pre pridanie údajov k prepojeniu vykonajte nasledujúce operácie:

1. Kliknite na položku *Settings* a vyberte *Issue trackers*.



2. Pokiaľ ste mali uložené údaje k prepojeniu na aplikáciu pre manažment projektov, uložené údaje sa Vám zobrazia a môžete ich meniť pokračovaním v kroku 5. Pokiaľ neboli doposiaľ uložené žiadne takéto údaje, polia sú prázdne.
3. Zobrazí sa Vám sprievodca pre nastavenie pripojenia k aplikácií pre manažment projektu. Vyberte jej typ (podporované TFS) a zadajte URL adresu servera a prístupové údaje k nemu.

**Pozn.:** Používateľské meno (pole Username) a heslo (pole Password) je jedinečné pre každého používateľa patriaceho k projektu. Ostatní používatelia projektu si budú musieť pred používaním webovej aplikácie CodeCrutches nastaviť vlastné prihlasovacie údaje pre zvolený issue tracker.

Po vyplnení údajov kliknite na tlačidlo „Check connection and continue »“ – systém overí funkčnosť pripojenia a v prípade, že bolo pripojenie úspešné, pokračuje krokom 4.

Connection Settings Wizard

Issue Tracker: TFS

TFS Server URL:

Username:

Password:

Check connection and continue »

4. Zobrazí sa druhý krok sprievodcu, v ktorom môžete vybrať projekt z aplikácie pre manažment projektov.



Po výbere položky z „Team Project Collection“ sa automaticky naplní zoznam dostupných projektov v danej kolekcii (položka „Project“). Podobne, po výbere položky z „Project“ sa automaticky naplní zoznam dostupných „Area“. Ak nie je zadaná žiadna „Area“, použije sa hodnota vybraná v položke „Project“.

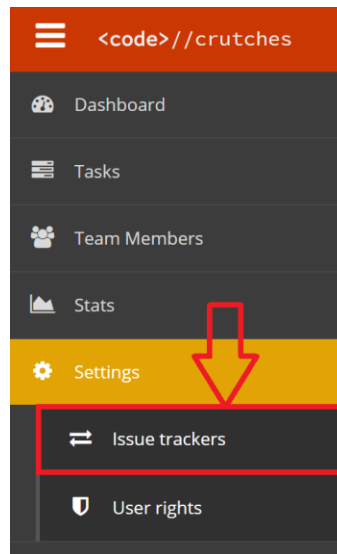
Nastavenia potvrdíte stlačením tlačidla „Save“, prípadne sa môžete vrátiť späť na predchádzajúci krok stlačením tlačidla „« Back“.

5. Zobrazí sa vám obrazovka s nastavenými vlastnosťami prepojenia. Tieto vlastnosti môžete kedykoľvek zmeniť.

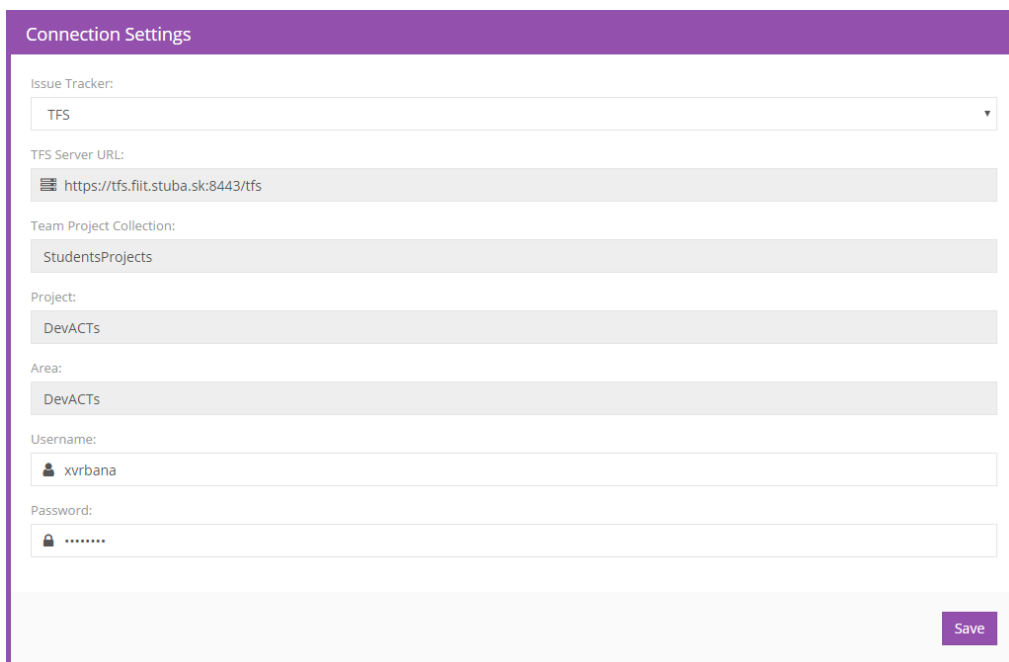
#### 4.3.2 Nastavenie prístupových údajov do prepojeného issue trackera

Ak administrátor projektu, ku ktorému máte prístup, už nastavil údaje potrebné pre prepojenie s issue trackerom (viď. časť 4.3.1), môžete si nastaviť vlastné prístupové údaje nasledovne:

1. Kliknite na položku *Settings* a vyberte *Issue trackers*.



2. V zobrazenom okne vidíte konfiguračné údaje issue trackera vyplnené administrátorom projektu.

A screenshot of a web application window titled 'Connection Settings'. The window has a purple header. Below the header, there are several form fields: 'Issue Tracker' (a dropdown menu with 'TFS' selected), 'TFS Server URL' (a text input field containing 'https://tfs.fiit.stuba.sk:8443/tfs'), 'Team Project Collection' (a text input field containing 'StudentsProjects'), 'Project' (a text input field containing 'DevACTs'), 'Area' (a text input field containing 'DevACTs'), 'Username' (a text input field containing 'xvrbana'), and 'Password' (a text input field with masked characters). A purple 'Save' button is located at the bottom right of the form.

3. Do kolonky *Username* a *Password* vyplňte Vaše prístupové údaje, ktoré používate pre prihlásenie do daného issue trackera.
4. Kliknite na tlačidlo uložiť. Pripojenie sa automaticky otestuje a budete informovaný o úspešnosti uloženia údajov.