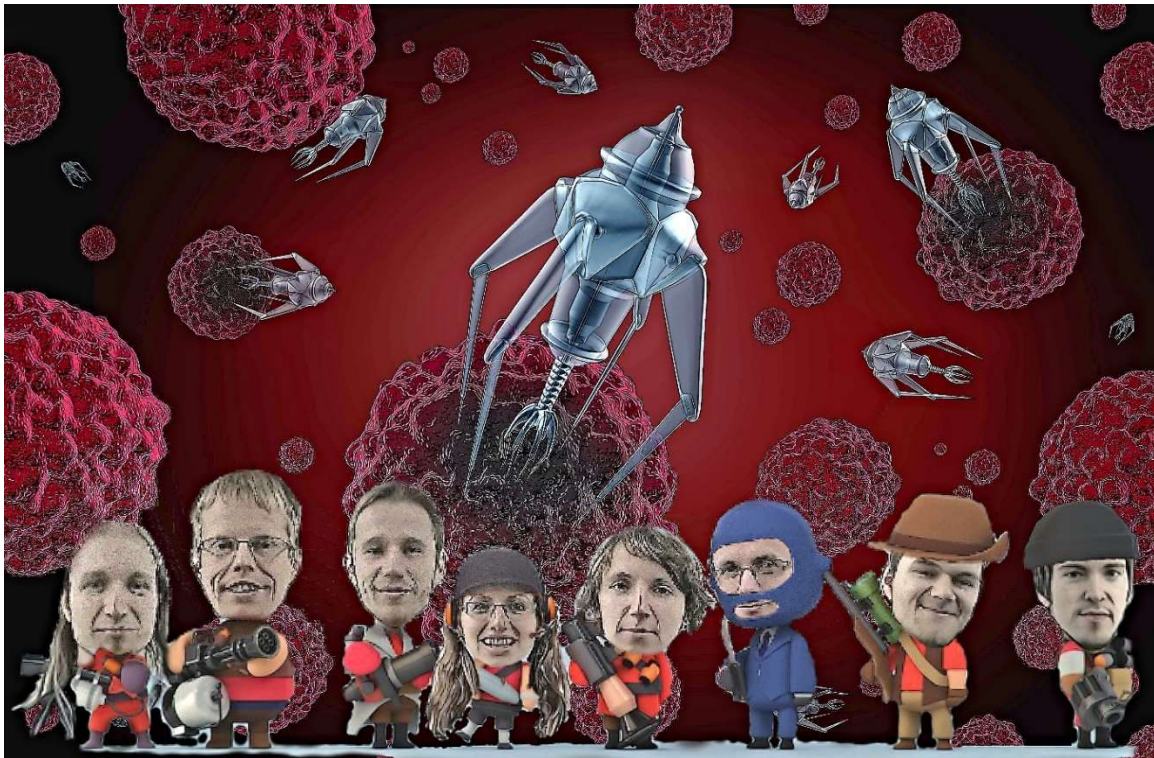


# TÍM č. 8 IS-SI

Blank



**Názov projektu:**

**TRACKS - Tasks, Code Reviews, Activities, Source Code and Knowledge about Software**

**Členovia tímu (študenti):**

Peter Bobovský, Michal Kráľ, Peter Kučera, Marek Mura, Miriama Pomffyová, Lukáš Račko, Michal Slovík

**Ved. tímu (pedagóg):**

Martin Konôpka

**Motto tímu:**

*Spájame nespojiteľné.*

**O ČOM JE NÁŠ PROJEKT?**

Ako už možno z názvu vytušiť, TRACKS je skratka pojmov: „Tasks, Code Reviews, Activities, Source Code and Knowledge about Software“. Programátori, rovnako ako aj my, pri tvorbe softvéru pracujú s veľkým

množstvom podporných systémov, ktoré im pomáhajú pri vývoji. Medzi najznámejšie patria vývojové prostredia (Eclipse, Visual Studio), systémy pre správu zdrojových kódov (Git), úloh a nahlasovania chýb (Bugzilla, Jira, Redmine), prehliadok zdrojového kódu (Gerrit), alebo aj diskusné fóra a dokumentácie (StackOverflow, MSDN). Problém nastáva, keď sa snažíme nájsť súvislosti medzi nimi. Príkladom je prepojenie otázky v StackOverflow od vývojára pracujúceho na úlohe z Bugzilla a jeho commity v Git. Pri následnej prehliadke zmien v Gerrit má kontrolór pred sebou iba výsledný návrh zmeny v Git a nie odpoveď zo StackOverflow, na ktorej vývojár svoje riešenie založil.

Cieľom tohto projektu bolo vytvorenie systému TRACKS pre zber a prepájanie dát z jednotlivých podporných systémov. Ich následné spracovanie tak, aby nadobúdali prínosnú hodnotu ako pre samotného programátora, tak aj pre manažéra projektu. Každý účastník vývoja bude mať dokonalý prehľad o tom, ku ktorému „Tasku“ patrí ktorý „Code Review“, „Activity“ a „Source Code“, poprípade, „Knowledge about Software“. Takýmto spôsobom sa naplní význam názvu projektu – TRACKS.

Zber dát je vykonávaný pomocou botnet klientov na základe zadaní od hlavného uzla distribujúceho sťahovanie dát tzv. balancera. Nasleduje prevedenie dát do spoločnej reprezentácie, pre ktorú sme zvolili RDF štandard - Resource Description Framework, ich samotné prepájanie a sprístupnenie tretím stranám. Jedným z hlavných cieľov bolo navrhnúť infraštruktúru tak, aby bola ľahko nasaditeľná ako na fakulte, tak aj interne v softvérovej firme a bola využitá záverečnými projektmi na fakulte.

### **ČO NÁM DÁVA PRÁCA NA TOMTO PROJEKTE?**

Keď sme prvýkrát počuli o predmete Tímový projekt, poniektorí sa tešili, že na prácu nebudú sami, iní sa naopak obávali spolupráce po dovedajších skúsenostiach. Pokiaľ na projekte pracujete sami, musíte si s problémami poradiť bez zásahu iného programátora. Pokiaľ ste na prácu dvaja, hrozí, že kvôli slabšej zodpovednosti toho druhého, alebo

kvôli problémom s komunikáciou, budete mať nie len horšie hodnotenie, ale aj horší pocit z výsledku.

Počas práce na projekte v tíme sme sa uistili v tom, že pravdu mali obe skupiny. Pokiaľ nastal problém, vždy sa našiel niekto, s kým ste si s “neriešiteľnou” situáciou vedeli poradiť. Programovať sme mohli v pokoji aj vo dvojici, dokonca aj v trojici. Práca na projekte napredovala. Mali sme však obrovské šťastie! Nič by nešlo tak hladko, nebyť dobrého zloženia tímu. Väčšina z nás sa už pred týmto predmetom poznala a pomáhali sme si počas predošlých rokov štúdia. Boli sme na seba zvyknutí a vedeli sme, čo môžeme od koho čakať.

Avšak počas práce na tomto projekte, sme postrehli zmeny. Zlepšila sa nám naša komunikácia. Naučili sme sa napláňovať si vlastné aktivity tak, aby sa našlo viac priestoru pre spoločnú prácu na projekte aj mimo oficiálnych stretnutí. Keď sme sa dostali do problematiky, začali sme lepšie ohodnocovať úlohy, nenaberali sme si na seba všetko naraz, ale iba toľko, koľko za šprint “unesieme”. Vďaka každej zmene sa zvýšila naša rýchlosť napredovania.

Riadili sme sa metódou Scrum so šprintami o dĺžke jedného týždňa. Nakoľko bol každý zvyknutý robiť všetko na poslednú chvíľu, vďaka kratším šprintom sme sa to nepriamo odnaučili a dosiahli sme vyššiu efektivitu práce. Prezentovaním svojich výsledkov a problémov sme sa vzájomne motivovali dosiahnuť viac. Splnením vlastných úloh sme motivovali aj sami seba. Môžeme teda povedať, že na základe týchto skúseností, bude každý z nás schopný stať sa prínosnou súčasťou vývojového tímu aj mimo fakulty.

### **PREČO JE NÁŠ PROJEKT ZAUJÍMAVÝ?**

Na každom väčšom projekte v praxi spolupracujú rôzni členovia tímu. Aby mal každý informácie o súčasnom stave riešenia projektu, využívajú rôzne podporné nástroje. Každú informáciu o úlohách, chybách, zdrojových kódoch, prehliadkach kódov a iných, si musia vyhľadať v samostatných systémoch.

Náš projekt ponúka riešenie, ktoré tento problém odstráni. Jeho cieľom je zozbieranie dát z viacerých zdrojov, ako sú StackOverflow, Bugzilla a Github, s možnosťou ľahkého doplnenia iného zdroja.

Zozbierané dáta našimi klientmi následne ukladáme do databázy v trojiciach: objekt, predikát, subjekt, ktoré zodpovedajú štandardu RDF - Resource Description Framework. Takto sú všetky informácie na jednom mieste. Takto sa otvára priestor pre dolovanie v dátach a vyhľadávanie informácii bez opätovného sťahovania dát.

### **POUŽITÉ TECHNOLOGIE:**

Backend:

C#, .NET Core, asp.net Core, Razor Views, Microsoft SQL Server, RDF databáza Jena, UNIX, Microsoft Windows Server, NuGet, Windows Presentation Foundation

Frontend:

Bootstrap, JavaScript, HTML

### **O ČOM TO VLASTNE JE?**

TRACKS je systém zložený zo skupiny klientov sťahujúcich dáta z repozitárov jednotlivých systémov. Ďalej z centrálného uzla, Balancera, rozdeľujúceho zadania klientom pre sťahovanie. Poslednou časťou je Server, ktorému klienti posielajú stiahnuté dáta a on ich následne ukladá do databázy Jena. Takto sú všetky dáta prístupné na jednom mieste.