

# **Maturitní práce z informatiky**

Gymnázium Jana Keplera, Parlérova 2, Praha 6

7941K41 Čtyřleté gymnázium pro absolventy ZŠ

Autor: Martin Koreček

Vedoucí práce: Karel Jílek

Praha 2018

*Prohlašuji, že jsem jediným autorem této maturitní práce a všechny citace, použité literatury a další zdroje jsou v práci uvedené.*

*Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium Jana Keplera, Praha 6, Parléřova 2 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.*

# Obsah

Anotace.....	4
Úvod.....	5
Analýza existujících podobných služeb .....	5
Implementace .....	7
Použité technologie.....	7
Postup při implementaci .....	7
Testování.....	8
Návod k instalaci.....	9
Závěr.....	13
Zmíněné odkazy .....	14

## **Anotace**

Výstupem práce je webová aplikace s názvem Colab, prostřednictvím níž může kterýkoli přihlášený uživatel navrhnout projekt, který by chtěl vytvořit a ostatní uživatelé mu mohou s tvorbou pomoci. Funguje to na jednoduchém principu, kdy uživatel s nápadem na projekt svůj nápad slovy popíše (může jednotlivé části svého nápadu rozčlenit do kapitol), případně také připojí související užitečné odkazy na jiné webové stránky a nápad zveřejní. Názvy zveřejněných projektů se zobrazují na úvodní stránce aplikace a dá se jejich prostřednictvím zobrazit popis projektu a komentáře k němu (komentáře k projektům mohou zveřejňovat pouze přihlášení uživatelé).

Cílem aplikace je pouze propojení kreativních lidí, samotnou práci na projektu neusnadňuje.

# Úvod

Tento projekt jsem si zvolil, protože myslím, že podobná aplikace na 'trhu' chybí a líbilo by se mi, kdyby reálně začala být používána větší komunitou lidí. Také projekt беру jako výzvu, jelikož jsem ještě webové aplikace nikdy nevyvíjel a hodně se projektem přiučím. V rámci maturitní práce si za cíl dávám vytvoření zcela základní verze aplikace Colab tak, aby již byla k účelu použitelná, ale nebyla přehnaně komplikovaná.

## Analýza existujících podobných služeb

Nápad na tento projekt vznikl na základě mého vlastního přání, aby taková webová stránka pro mé využití existovala. Proto jsem podobné projekty ještě před úvahou, zda ji nevytvořit sám, nějaký čas hledal (přes Google Search <sup>1</sup>). Marně. Proto jsem přesvědčen, že taková aplikace opravdu ještě neexistuje, nebo alespoň není v provozu.

Něčím podobných stránek je však několik a tak některé z nich proberu v této kapitole a vysvětlím, co Colab přináší nového, popřípadě jak se jimi může inspirovat.

První na mysl přicházejí stránky pro tzv. freelancery (schopné lidi pracující mimo dlouhodobý pracovní poměr na různých placených projektech), pěkným tuzemským příkladem takové služby je třeba Topdesigner <sup>2</sup>. Podobnost s mým projektem je zřejmá, nicméně celá filozofie vydělávání online je od konceptu Colabu odlišná. Sám jsem na podobných stránkách nějaké peníze získal, seznámil jsem se s tímto prostředím a pro drtivou většinu freelancerů jde zkrátka jen o snahu získat nabízené peníze, na čemž mně samotnému chyběla určitá seberealizace. Chtěl jsem pracovat na projektech, ke kterým bych měl alespoň nějaký vztah, byť zcela bez zisku, a zároveň jsem měl plný sešit konceptů, které jsem zkrátka nebyl schopen sám zrealizovat. Kdyby tehdy Colab existoval a měl i solidní komunitu, opravdu bych to ocenil.

Snad největší podobnost s mým projektem se dá najít u služby Kickstarter <sup>3</sup>, která je opravdu výborná pro existující týmy či jednotlivce s dobrým nápadem. Poskytuje jim možnost svůj nápad představit ostatním uživatelům, kteří jej pak mohou podpořit darováním de facto libovolné finanční částky. Důležitá výhoda Kickstarteru je, že týmy často nabízí za dárcovství odměny ve formě produktů a služeb souvisejících s projektem. Tím v podstatě také shání důležité zákazníky. Kickstarter je velkou měrou podobný Colabu (alespoň co se týče jejich konceptu a filozofie) a v době, kdy vznikl, šlo skutečně o průkopnický nápad a svého druhu revoluci pro malé neznámé týmy. Není však vhodný pro jednotlivce s dobrým nápadem bez schopnosti jej samostatně

zrealizovat. Podporuje se zde 'jen' prostřednictvím peněz a uživatelé se v samotném projektu nemohou zapojit.

Protipólem Kickstarteru jsou stránky jako je StackOverflow <sup>4</sup>. Tam uživatelé pomáhají právě naopak s tvorbou projektů (zejména kódu), o podobnosti s Colabem se však skoro nedá mluvit. Pomoc zde totiž najdeme jen s dílčími postupy, například bychom zde mohli položit otázku 'Jak s pomocí této technologie zobrazíme tlačítko uprostřed obrazovky?' a přestože je komunita této služby skvělá a ve valné části případů nám poradí, celý projekt by se postupným dotazováním zde realizoval opravdu těžko a dlouho. StackOverflow tak může mému projektu sloužit spíše jako inspirace.

Za zmínku ještě stojí služba Trello <sup>5</sup>, pomocí níž mohou týmy organizovat, kdo se chopí kterého úkolu. Zmiňuji ji jen jako inspiraci pro případná pozdější rozšíření funkčnosti Colabu, případně jako prostředek využitelný uživateli Colabu (tvůrce projektu může zvlášť vytvořit 'nástěnku' na Trello a odkaz na ni vložit do 'resource linků' u svého projektu).

## Implementace

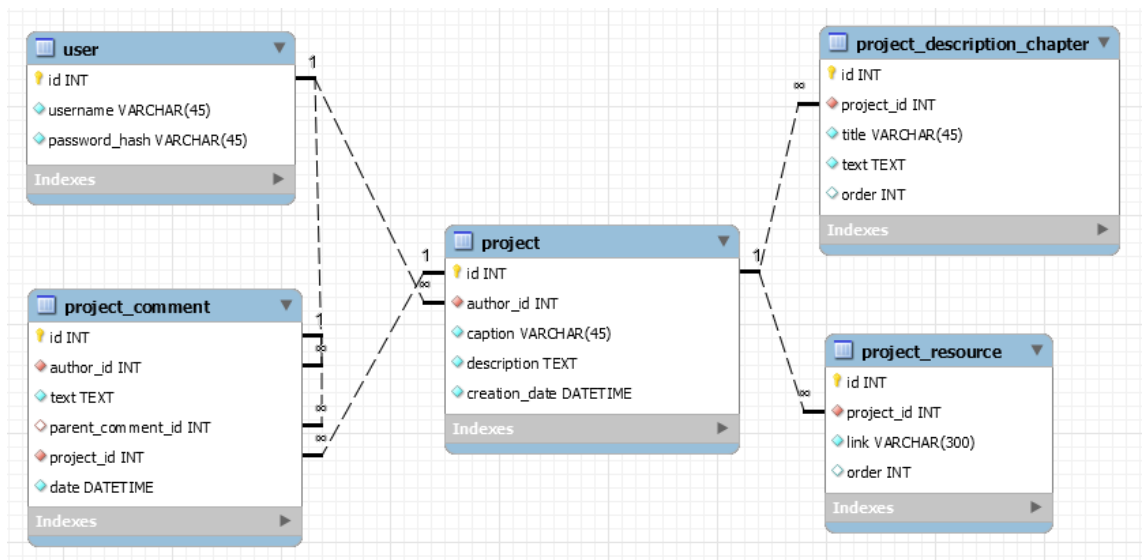
Tato kapitola jednoduše popisuje tvorbu maturitního projektu.

### Použité technologie

- MySQL (databáze)
- Java, Spring framework, Hibernate framework (přístup do databáze, backend)
- Maven (pro správu 'buildů' backendu)
- Angular 2 (frontend)
- Twitter Bootstrap (respektive jeho min-css pro verzi 3; použito k usnadnění tvorby grafického rozhraní (frontendu))

### Postup při implementaci

Základem aplikace je databázový model, jeho schéma lze vidět na **obrázku 1**, navíc se ve zdrojovém kódu můžete podívat na jeho create script, v umístění `backend/Colab/src/main/resources/cz/martinkorecek/colab/create.sql`.



•Obrázek 1

Backend, jak už jsem zmínil výše, jsem psal v Javě a jeho tvorbu mi neskutečně usnadnily frameworky Spring a Hibernate, díky nimž bych de facto (až na create script) nemusel napsat jedinou řádku SQL kódu. Stačilo by ve svých rozhraních (interface) oannotovaných jako `org.springframework.stereotype.Repository` rozšiřovat (extends) rozhraní `org.springframework.data.repository.CrudRepository` a Spring by za mne naimplementoval všechny CRUD (create, update, delete, get) metody. Tuto možnost jsem však nevyužil a některé SQL 'queries' jsem psal sám (viz např.

`backend/Colab/src/main/java/cz/martinkorecek/colab/repository/ProjectRepository.java` ve zdrojových souborech). Další podstatnou výhodou využití Spring frameworku je, že 'v sobě' má implicitně zahrnut Tomcat server a ke spuštění backendu tak stačí pouze spustit hlavní třídu `backend/Colab/src/main/java/cz/martinkorecek/colab/Application.java`.

To bylo stručně k backendu a ještě se zmíním o frontendu. K jeho tvorbě jsem použil platformu Angular 2 (Angular Typescript) a celá webová aplikace je koncipovaná jako jednostránková (single-page), což je možné díky výborně vyřešenému 'routingu' v Angularu. Za pozornost také stojí, jak snadno se dají pomocí Angularu vytvářet komponenty prostřednictvím *ng-templates*. Díky výběru této platformy se i mně jako frontendovému začátečníkovi stránka vytvářela pohodlně.

Ještě musím zmínit, že k implementaci loginu (přihlášení) uživatelů jsem následoval tutoriál (odkaz [6](#)). Díky němu jsem se naučil šifrovat heslo (převést plain-text heslo na *password\_hash* pro tabulku *user*) a vlastně celý proces přihlašování. Login v Colabu funguje tak, že backend po úspěšném přihlášení uživatele vygeneruje token (OAuth2 token) specifický pro každou session, který pak zase frontend posílá v hlavičce post requestů a tím ověřuje 'identitu' uživatele.

## Testování

Ideální by samozřejmě bylo pro všechny metody vstupující do databáze pečlivě vytvořit unit testy. To jsem však zanedbal a celý Colab je testován pouze praktickým zkoušením funkčnosti. Nedbalost je to podstatná, avšak projekt je natolik jednoduchý, že snad unit testy nejsou nezbytně nutné.

Ještě jednu věc jsem při implementaci projektu zanedbal, a to selectování (a zobrazování) komentářů pod projekty. Komentáře se při zobrazování projektu načítají všechny (patřící k danému projektu). To by při velkém množství komentářů mohlo zpomalovat přístup do databáze a při opravdu velkém množství dokonce přehltit paměť prohlížeče.

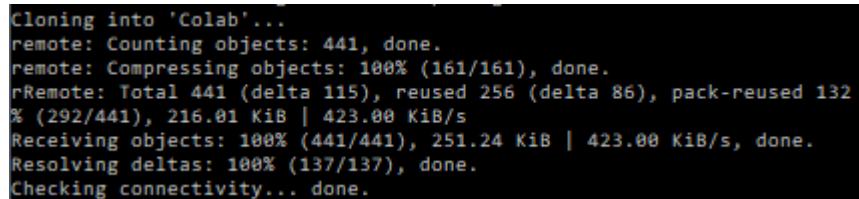


## Návod k instalaci

V této kapitole se dozvíte, jak obdrženy zdrojový kód nainstalovat ve smyslu jeho kompilace a spuštění na lokálním serveru, tedy pouze v rámci svého zařízení. Než s instalací začnete, ujistěte se, že na zařízení máte nainstalovanou aktuální verzi Javy, NodeJS, MySQL a funkční heslem chráněnou MySQL lokální službu [7](#), moderní webový prohlížeč (tzn. novou verzi např. Mozilla Firefox, Google Chrome, Microsoft Edge, atd.) a že jste připojeni k internetu. Dále budete potřebovat mít nainstalovaný MySQL Workbench (k jeho instalaci můžete použít návod v odkazu [8](#)) a Git (odkaz [9](#), pouze pokud ještě nemáte zdrojové kódy Colabu).

Ještě je potřeba mít na paměti, že součástí projektu jsou serverové technologie. Nebýt nich, stačilo by poskytnout spustitelné soubory a žádný návod by nebyl potřeba. Kvůli nim je také, bohužel, potřeba pro instalaci mít nejdříve nainstalovanou Javu, MySQL a MySQL Workbench. Pamatujte však, že pomýšlíme-li o nasazení Colabu 'na produkci,' koncový uživatel by nic z toho nainstalované mít nepotřeboval a stačil by mu moderní webový prohlížeč.

Teď už však k postupu. Pokud jste zdrojové kódy neobdrželi jinak, naklonujte si repositář Colabu z GitHubu. Na to budete potřebovat v příkazovém řádku (cmd / bash) v umístění, kde chcete zdrojové kódy mít, užít příkaz `git clone https://github.com/MartinKorecek/Colab.git`. Vypíše se něco podobného **obrázku 2** a ve zvoleném umístění teď budete mít možnost zdrojové kódy najít.



```
Cloning into 'Colab'...
remote: Counting objects: 441, done.
remote: Compressing objects: 100% (161/161), done.
rRemote: Total 441 (delta 115), reused 256 (delta 86), pack-reused 132
% (292/441), 216.01 KiB | 423.00 KiB/s
Receiving objects: 100% (441/441), 251.24 KiB | 423.00 KiB/s, done.
Resolving deltas: 100% (137/137), done.
Checking connectivity... done.
```

•Obrázek 2

Nyní bude třeba vytvořit novou databázovou 'connection' pro Colab. Proto spusťte MySQL Workbench a connection vytvořte (můžete následovat návod v odkazu [10](#), do kolonky Hostname napište 'localhost' a v okně Specify Commands nechte kolonky prázdné) a údaje z ní doplňte do souboru `backend/Colab/src/main/resources/application.properties`. K tomu poslouží obyčejný textový editor. Při vytváření connection si tedy musíte zapamatovat vámi zadané údaje a doplnit je do zmíněného souboru na místa s otazníkem (viz **obrázek 3**)

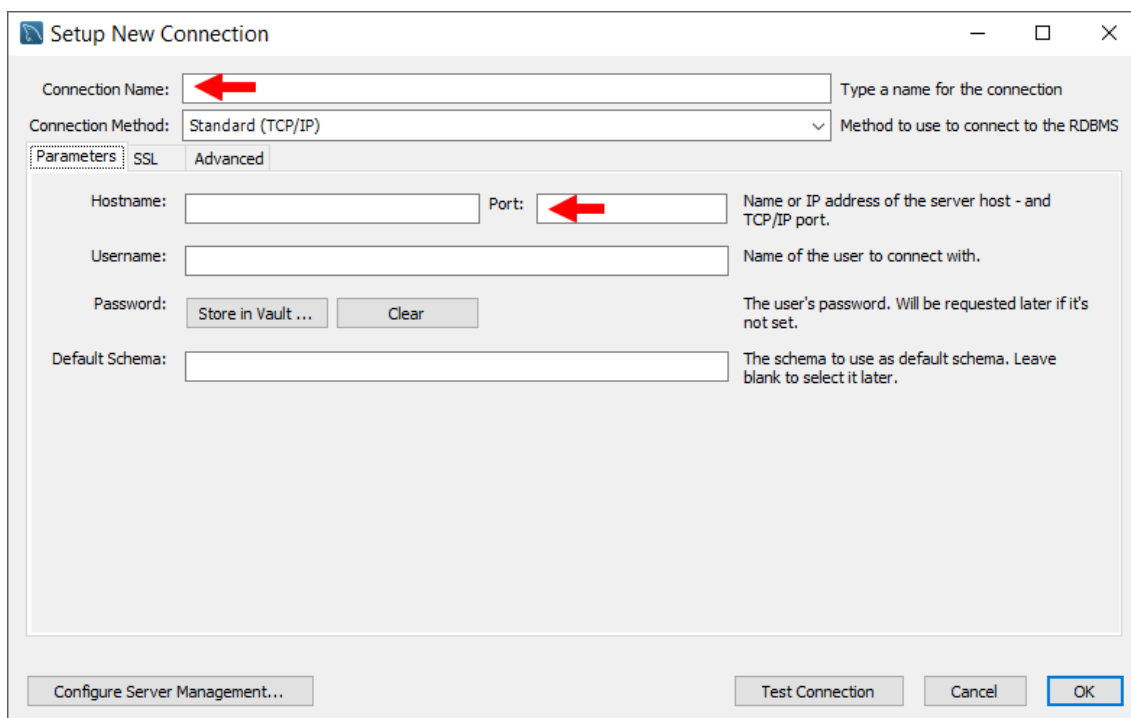
```
# Zde nastavte svůj connection string
spring.datasource.url=?
# Sem patri vami nastavené údaje
spring.datasource.username=?
spring.datasource.password=?
```

•Obrázek 3

Do kolonky `spring.datasource.password` vložte heslo své služby SQL (heslo můžete nastavit podle odkazu [7](#))

Connection string by měl mít formát `jdbc:mysql://localhost:*port*`

`/*Connection name*.` Údaje ve hvězdičkách doplňte podle vámi zadaných dat (viz obrázek 4).



•Obrázek 4

Ted' vytvoříme databázi. Pokud se vám povedlo vytvořit connection, měla by se vám v MySQL Workbench zobrazit mezi 'MySQL Connections.' Klikněte na ni, zadejte svoje heslo a měli byste ted' vidět otevřený editor SQL kódu. Klikněte na *File > Run SQL script...* a vyberte soubor umístěný v `backend/Colab/src/main/resources/cz/martinkorecek/colab/create.sql` a klikněte na Run. Databáze je nyní vytvořena.

Pokud se vám předchozí kroky povedlo následovat, můžeme přejít k samotnému spuštění aplikace. Otevřete si ted' zvlášť dva příkazové řádky. Jedním spustíme backend

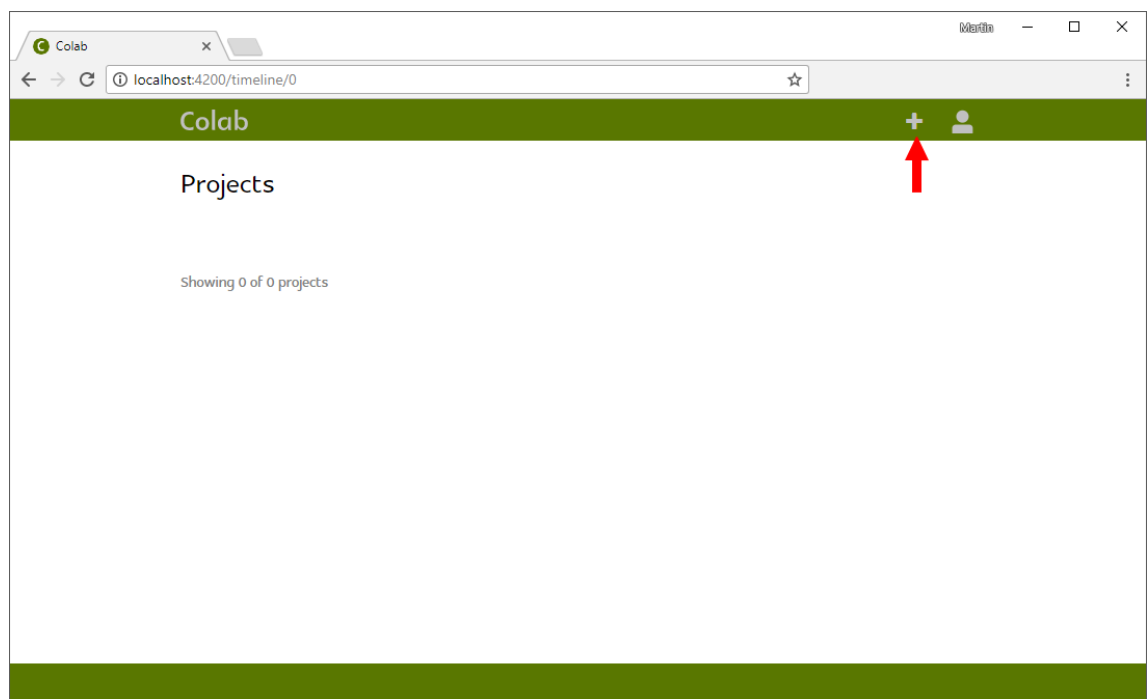
a druhým frontend. První tedy backend. Vstupte do umístění `backend/Colab` a spusťte build backendu příkazem `mvn clean install`. V jednom z posledních řádku se vám vypíše `'[INFO] BUILD SUCCESS.'` Nyní bude stačit příkaz `mvn exec:java`, jímž backend spustíte.

Pokud nyní backend běží (vypsalo se 'Started Application'), je čas spustit frontend. K tomu využijeme druhý spuštěný příkazový řádek a přejdeme do umístění `frontend/Colab`, kde užijeme následující příkazy v tomto pořadí:

- 1: `npm install` -instalace Node modulů; selhání příkazu značí špatně nainstalovaný NodeJS
- 2: `npm install angular2-jwt` -instalace modulu pro zacházení s JSON Web Tokeny, nutnými pro bezpečné přihlašování uživatelů
- 3: `npm install bootstrap3` -instalace Bootstrapu (důležité je nainstalovat třetí verzi, ve čtvrté už nejsou tzv. 'glyphicons,' které v projektu používám)
- 4: `ng serve` -spuštění frontendu

-pokud tento příkaz selže, použijte místo něj `npm run ng serve`

Nyní je aplikace spuštěna a měli byste mít možnost ji v prohlížeči najít na adrese `http://localhost:4200`. Ověřte, že vypadá podobně jako **obrázek 5**, akorát bez přidané červené šipky. Šipka slouží jako vodítko, kam kliknout k vytvoření projektu. Nejdřív se budete muset zaregistrovat jako nový uživatel, ale pak vás stránka hned přesměruje na tvorbu projektu.



•Obrázek 5

Při tvorbě projektu si všimněte intuitivní funkčnosti přidávání kapitol a 'resource linků' zajištěné dynamicky přidávanými (a dynamicky odebratelnými) komponentami.

Pokud byste chtěli vidět, jak úvodní stránka Colabu vypadá s větším množstvím projektů, můžete v MySQL, podobně jako dříve *create.sql*, spustit soubor `backend/Colab/src/test/resources/mock_projects.sql`. Projekty, které vám tento skript vygeneruje slouží čistě k zaplnění místa na úvodní obrazovce Colabu a umožní vám v pravém spodním rohu aplikace vidět komponentu k listování stránkami (**obrázek 6**). Teď už byste měli mít možnost užívat všechny funkce Colabu.

[Previous page](#) [Page 1 of 3](#) [Next page](#)

•Obrázek 6

## **Závěr**

Podarilo se mi vytvořit jednoduchou a přehlednou webovou aplikaci umožňující svým uživatelům najít spolupracovníky na kreativních projektech, případně naopak vyhledávat zajímavé projekty, na kterých se zapojit. V tomto stavu je aplikace připravena ke svému nasazení na 'produkcii,' ale zároveň také dává prostor celé řadě rozšíření a vylepšení k případnému pozdějšímu doplnění.

## Zmíněné odkazy

- 1: <https://www.google.com/>
- 2: <https://www.topdesigner.cz/>
- 3: <https://www.kickstarter.com/>
- 4: <https://www.stackoverflow.com/>
- 5: <https://www.trello.com/>
- 6: <https://medium.com/@juliapassynkova/angular-springboot-jwt-integration-p-1-800a337a4e0>
- 7: <https://dev.mysql.com/doc/refman/5.7/en/resetting-permissions.html>
- 8: <https://dev.mysql.com/doc/workbench/en/wb-installing.html>
- 9: <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>
- 10: <https://dev.mysql.com/doc/workbench/en/wb-getting-started-tutorial-create-connection.html>