

**Gymnázium Arabská, Praha 6, Arabská 14**

Programování

## **Ročníkový projekt**



**Gymnázium, Praha 6, Arabská 14**

Arabská 14, Praha 6, 160 00

## **ROČNÍKOVÝ PROJEKT**

**Předmět:** Programování

**Téma:** Sociální síť pro studenty

**Autor:** Martin Krátký, Ema Heřmánková, Natálie Hutníková

**Třída:** 3.E

**Školní rok:** 2023/2024

**Vedoucí práce:** Mgr. Jan Lána



Prohlašujeme, že jsme jedinými autory tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů udělujeme bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Praze dne 26. dubna 2024

---

## **Anotace**

Cílem bylo naprogramovat webovou aplikaci, která by fungovala jakožto sociální síť pro studenty, kde by mohli sdílet a číst články. Příspěvky by pak mohl být zařazený do jedné z kategorií podle školních předmětů.

## **Abstract**

The goal was to program a web application that would act as a social network for students to share and read articles. Posts could then be classified into one of the categories according to school subjects.

## **Zadání projektu**

Webová aplikace, která by fungovala jakožto sociální síť určená převážně pro studenty, kde budou moci sdílet různé příspěvky k učení. Student (uživatel) by po přihlášení mohl buďto číst příspěvky, nebo je sám psát. Příspěvek by pak mohl být zařazen do jedné z kategorií podle školních předmětů.



# Obsah

<b>1. Úvod</b>	<b>1</b>
<b>2. Databáze</b>	<b>2</b>
2.1. Student	2
2.2. Article	2
2.3. Comment	3
2.4. Subject	3
2.5. Confirmation Token	3
<b>3. Backend</b>	<b>4</b>
3.1. Anotace	4
3.2. Uživatel	5
3.3. Články, komentáře a předměty	6
3.4. Security	7
3.5. Propojení s Aplikací React	7
<b>4. Frontend</b>	<b>9</b>
4.1. App.js	9
4.2. NavBar	9
4.3. Login	10
4.4. Register	11
4.5. Articles	12
4.6. Create Article	14
4.7. Account	15
<b>5. Použité knihovny a nastavení aplikace</b>	<b>18</b>
<b>6. Závěr</b>	<b>19</b>
<b>7. Citace a zdroje</b>	<b>20</b>



# 1. Úvod

Cílem této ročníkové práce bylo naprogramovat webovou aplikaci, kde uživatelé mohou přidávat a číst příspěvky ohledně učiva. Uživatel může zařadit článek dle předmětu. Aplikace se skládá z frontendu, backendu a databázi.

Backend byl naprogramován v Javě za použití Spring verze 2.6.7. Spring je open-source framework založený na Javě. Spring byl vyvinut Rodem Johnem v roce 2003. Více informací naleznete na stránkách: <https://spring.io/>.

Frontend byl naprogramován v Reactu verzi 18.2.0. React je JavaScriptová knihovna vytvořená Facebookem. Více o této knihovně naleznete na stránce: <https://react.dev/>.

Aplikace používá databázi postgresql 15.

Pro funkčnost programu se musí zavolat příkaz `.\mvnw spring-boot:run` a připojit postgres databáze. Pro připojení se musí změnit první tři řádky v `resources/application.properties`. Dále se musí nastavit mailový server.

## 2. Databáze

Pro použití této aplikace je potřeba připojit program k databázi postgres. Program po prvním spuštění po připojení k této databázi vytvoří pět tabulek s názvy Student, Article, Comment, Subject a Configuration Token. Toto nastavení se nachází v souboru application.properties a pro zprovoznění je potřeba změnit první tři řádky.

Tabulky jsou označeny anotacemi Table a Entity, které značí tabulku. Každý sloupec je pak anotován anotací column. Id je anotováno Id, SequenceGenerator a GeneratedValue, které slouží k funkčnosti id.

### 2.1. Student

Tabulka Student se skládá ze sedm sloupců a obsahuje uživatele. Id slouží jako id uživatele a je generováno sequenčně. Email slouží jako unikátní uživatelské jméno, Heslo se ukládá kryptovaně a enabled slouží k rozpoznání ověřeného uživatele. Dalšími sloupce jsou name, studentRole a locked.

### 2.2. Article

Tabulka Article se skládá z šesti sloupců a obsahuje články. Id slouží jako id článku a je generováno sequenčně. Sloupec Autor je povinný a je to id uživatele, který jej vytvořil. Title je unikátní název článku. Article je text článku. Subject je id předmětu, jenž byl přiřazen ke článku. Article také obsahuje sloupec date, jenž je časem, kdy článek byl přidán.

## 2.3. Comment

Tabulka Comment se skládá z pěti sloupců a obsahuje komentáře uživatelů, které byly napsány k článku. Id slouží jako id komentáře a je generováno sequenčně. Sloupec autor je id uživatele, který jej vytvořil, a sloupec article je id článku, pod kterým byl napsán. Text je textem, který napsal uživatel pod komentář a date je časem, kdy byl článek vytvořen.

## 2.4. Subject

Tabulka Subject se skládá ze dvou sloupců a obsahuje předměty. Tyto dva sloupce jsou Id, sloužící jako id generující se sequenčně, a name, které slouží jako název předmětu.

## 2.5. Confirmation Token

Tabulka Confirmation Token se skládá z šesti sloupců a slouží k uchování tokenu, který je zapotřebí k ověření účtu při založení. Dalšími sloupce jsou id, které funguje stejně jako u ostatních tabulek, createdate, neboli datum vytvoření tokenu, expiresat, který určuje do kdy token platí, confirmedat, neboli čas potvrzení tokenu, a sloupec student.

## 3. Backend

Backend byl naprogramován Javě z použití frameworku Spring. Cílem backendu je zpracovávat požadavky frontendu. Backend zahrnuje propojení s databází a odchyťování endpointů.

### 3.1. Anotace

Spring pro práci s databází používá anotace, ani program věděl, co jaká třída dělá, proto bych rád v následující kapitole popsal některé z těchto anotací.

Pro práci s tabulkami jsou vytvořeny třídy Controller, které mají anotace mapping, jenž udává příkaz, který se má volat. Mappingy použité v programu jsou celkem čtyři a to delete, pro smazání dat z databáze, put, pro změnu dat v databázi, post, pro přidání záznamu do databáze, a get, pro dostání dat z databáze. V programu se také můžete setkat s RequestMapping, který slouží jako cest na stránce.

Dalšími použitými anotacemi jsou RestController, Repository, Service a Configuration. RestController označuje řídicí třídu, která obsahuje metody s anotací mapping. Repository je vždy anotováno u interfacu, který rozšiřuje JpaRepository. JpaRepository obsahuje metody základních příkazů k práci s databází. Anotace Query pak přidává jeden příkaz pro práci s databází. Service slouží na označení třídy, kde je samotná logika programu. Configuration slouží k nastavení, konkrétně v této práci na nastavení zabezpečení endpointů a přidání prvků do tabulky subject. S Configuration se velice často používá anotace Bean, která specifikuje vrácení.

Kromě anotací ze Spring jsou použity i anotace z knihovny Lombok. Tyto anotace mají za cíl zkrátit kód a zlepšit jeho čitelnost. Z této knihovny pochází například anotace

Getter a Setter, které automaticky generují gettery a settery pro proměnné z dané třídy.

Více se můžete dozvědět zde: <https://projectlombok.org/>.

## 3.2. Uživatel

Aplikace vlastní uživatele pod názvem Student. Uživatele je možné registrovat, přihlásit, smazat a změnit heslo nebo jméno. Dále je také možné získat všechny uživatele z databáze. Ve třídách ve složce validators můžete nalézt metody, které testují nějaké parametry studenta. Pokud nějaký test neprojde, vypíše chybu v konzoli.

Nejzajímavější ze všech funkcí je registrace, kterou je možné zavolat jak z StudentController tak z RegisterController. Registrace z StudentController je mnohem méně komplexní, jelikož jen zkontroluje zda se v databázi nenachází uživatel se stejným emailem a pokud ne tak jej přidá do databáze.

Registrace z RegisterController je mnohem komplikovanější. Než se uživatel zaregistruje, nemůže přihlásit. Aplikace mu pošle email, pomocí kterého uživatel potvrdí svoje přihlášení. Tento email je možné potvrdit jen po dobu 15 minut. Pokud se uživateli nepovede potvrdit email, tak se musí znovu zaregistrovat. Uživatel potvrdí email tím, že klikne na link, který ho přesune na jinou stránku. Po potvrzení emailu se uživatel smí přihlásit na stránku. Tento kód byl částečně použit z videa <https://www.youtube.com/watch?v=QwQuro7ekvc>. Email můžete vidět na obrázku

1.

## Confirm your email

---

Hi martin,

Thank you for registering. Please click on the below link to activate your account:

[Activate Now](#)

Link will expire in 15 minutes.

See you soon

Obrázek 1: Mail poslaný programem

### 3.3. Články, komentáře a předměty

Pro propojení frontendových příkazů a ostatních databází jsou používány metody s názvem Controller, které pak volají metody z názvy Service.

ArticleController umí získat články dle parametru title a předmět. Dále umí získat všechny články přihlášeného uživatele. Přihlášený uživatel se získá díky proměnné Principal. Dále umí smazat článek z databáze, přidat článek do databáze a změnit text článku. Když je smazán článek, jsou automaticky smazány i všechny komentáře pod článkem. Požadavky ohledně článků jsou zpracovávány na api/v1/article.

CommentController dokáže získat všechny komentáře, které byly napsány pod článkem. Dokáže mazat komentáře i přidávat nové komentáře. Požadavky ohledně komentářů jsou zpracovávány na api/v1/comment.

SubjectController umí získat všechny předměty, nebo jen předmět podle id. Jelikož jsou všechny předměty již vytvořeny, nejsou zde žádné metody na přidávání předmětů do databáze a mazání předmětů z databáze. Všechny předměty můžete nalézt v SubjectConfig. Požadavky ohledně komentářů jsou zpracovávány na `api/v1/subjects`.

## 3.4. Security

Kromě propojení frontendu s databází se backend stará i o zabezpečení endpointů. Ve springu ve verzi 2.6.7. je třída `WebSecurityConfigurerAdapter`, která umožňuje zabezpečovat endpointy, pokud není uživatel zaregistrován. Tuto metodu již nelze použít od verze 3.0.0, kde endpointy zabezpečují díky anotaci `Bean`. Aby jsme mohli nějakou třídu použít jako uživatele musíme ji implementovat interface `UserDetail` a přepsat některé jeho metody. Pro získání parametrů přihlášeného uživatele můžeme požadovat v metodách o parametr `Principal`. Konfigurace samotných endpointů můžete nalézt v souboru `WebSecourityConfig` v metodě `config`. Tento kód byl použit z videa <https://www.youtube.com/watch?v=QwQuro7ekvc>.

Security automaticky vytváří svůj vlastní login. Pro použití vlastní stránky na login se muselo použít `.formLogin().loginPage("/login")`, aby náš login byl vytvořen na stránce `/login`.

## 3.5. Propojení s Aplikací React

Pro propojení HTML souboru se spring aplikací se používají template, které by za použití čistého HTML byly ve složce `resources`. Jelikož pro frontend používáme React, musely se použít pluginy, které nahrají Reaktové soubory do souborů `target`. Tyto pluginy se nachází v `pom.xml`.

Pro každé změně ve složce reactapp, se musí zavolat příkaz `.\mvnw spring-boot:run`, jinak změny nejsou vidět. Tento příkaz se musí volat i v případě zkopírování z githubu.

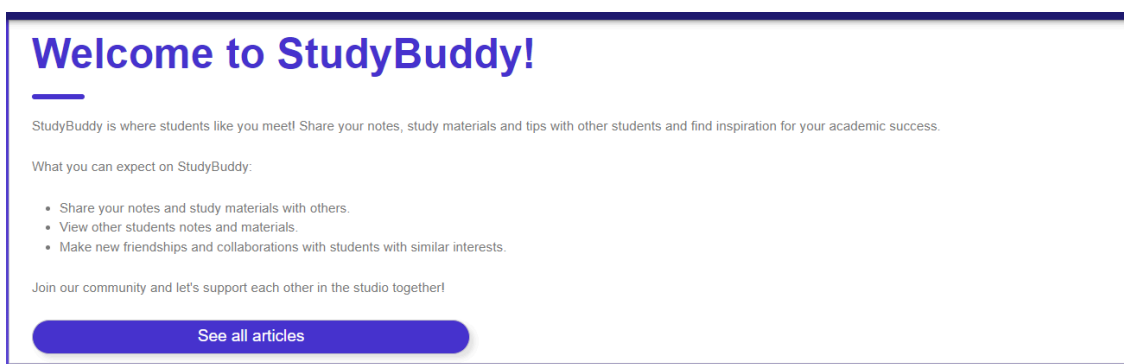
Propojení ještě zajišťuje třída `AppController`, která všechny cesty co nemají písmena `c`, `x`, `q` a `p` nasměrovává na reactapp v target.



## 4. Frontend

Frontend, který je napsán v Reactu, je ve složce reactapp. Pro lepší orientaci v projektu jsou komponenty rozděleny do složek dle názvu dané route. Pro komunikaci frontendu s backendem je použit axios.

Všechny styly najdete ve složce css, kde card slouží k nastavení vzhledu článků, CreateUpdateArticlesAccount slouží k stylování stránky s články, homepage, stránky account a stránky pro vytváření a úpravu článků a RegisterLogin slouží k stylování. Homepage je stránka, kde jsou základní informace o stránce. Tuto stránku můžete vidět na obrázku 2. Po zmáčknutí na see all articles se stránka přesměruje na stránku se všemi články.



Obrázek 2: homepage

### 4.1. App.js

App.js je hlavní funkce, která vždy složí stránku dle aktuální cesty. K zjištění cesty App.js používá komponent Route z react-router-dom. Stará se tedy o cesty k home, loginu, změně hesla, registraci, účtu, článků a úpravě a vytváření článku. Kromě Route a s ní souvisejících komponentů obsahuje App.js NavBar, který je vždy bez ohledu na cestu přidán.

## 4.2. NavBar

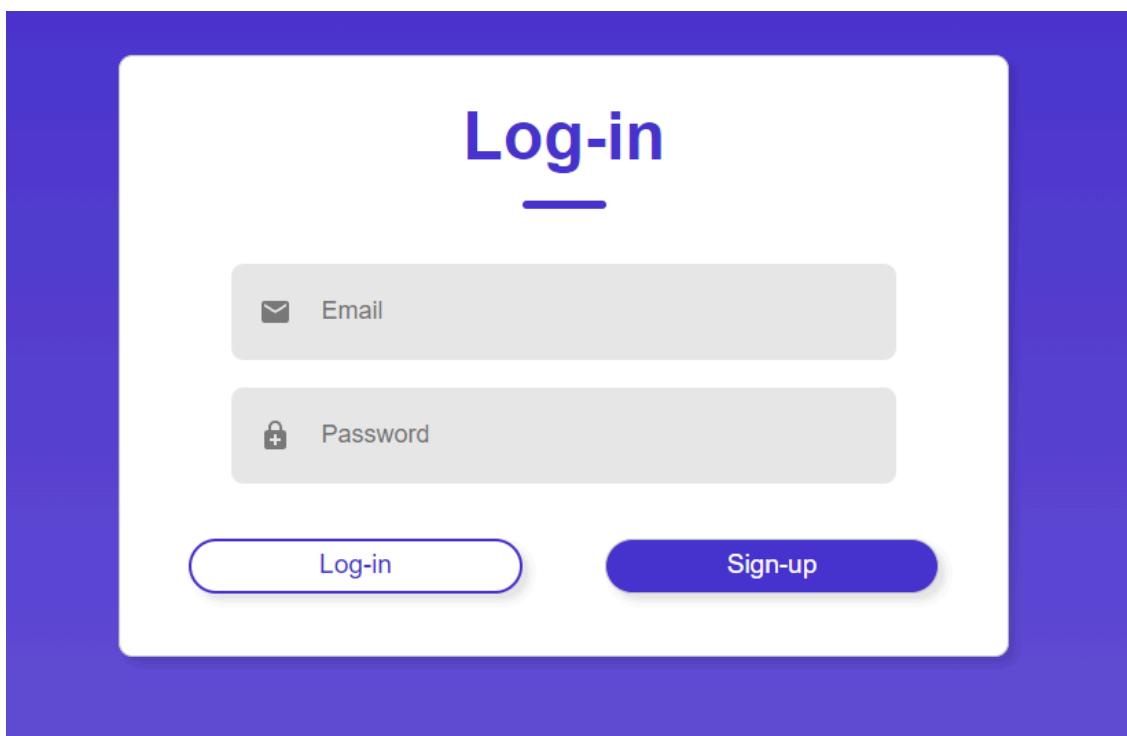
Pro rychlou orientaci na stránce je použit komponent NavBar. Po zmáčknutí na jakýkoliv text nebo tlačítko v navbaru se stránka automaticky přesune na jinou cestu.

Na mobilu jsou všechny tlačítka a text, kromě názvu, přesunut do menu. Po kliknutí na menu, se menu otevře, a uživateli nabídne všechny možnosti. Breakpoint pro změnu je md. NavBar byl vytvořen díky knihovně mui.

## 4.3. Login

Komponent login se zavolá pod cestou /login. Při otevření jakékoliv stránky, kromě samotného loginu, registru a home, na uživatele vyskočí login. Aby se uživatel mohl dostat dál musí se přihlásit. O zabezpečení se stará spring security. Pokud uživatel navštíví stránku po druhé, tak email, kterým se přihlásil nebo ho jen zadal, se zobrazí v kolonce email. Tato hodnota je uložena na stránce pokaždé když se změní.

Po kliknutí na Login se pošle přihlašovací dotaz, který bude zpracován spring security. Po kliknutí na tlačítko sign up se zobrazí stránka register. Login můžete vidět na obrázku 3.



Obrázek 3: Login

## 4.4. Register

Komponent register se zavolá pod cestou /register. Pro zaregistrování uživatele musí být vyplněno heslo, jméno i email. Pokud uživatel navštíví stránku po druhé, tak email a jméno, kterými se zaregistroval nebo jen zadal, se zobrazí v kolonce email a username. Email se také zobrazí v login v kolonce email. Tyto hodnoty jsou uloženy na stránce pokaždé když se změní.

Po kliknutí na Sign up se pošle příkaz na registraci a uživateli se zobrazí stránka s přihlášením. Po kliknutí na tlačítko login se zobrazí stránka s přihlášením. Register můžete vidět na obrázku 4.

The image shows a sign-up form titled "Sign-up" in a large, bold, blue font. Below the title is a horizontal line. The form consists of three input fields, each with a gray background and a light gray border. The first field is labeled "Username" with a person icon. The second field is labeled "Email" with an envelope icon. The third field is labeled "Password" with a lock icon. Below the input fields are two buttons: a solid blue "Log-in" button and a white "Sign-up" button with a blue border. The entire form is centered on a solid blue background.

Obrázek 4: Register

## 4.5. Articles

Komponent `articles` se zavolá pod cestou `/user/articles`. Úkolem tohoto komponentu je zobrazit články. Jaké články zobrazí je určeno dle `select`. Komponenty `select` jsou předměty, které jsou v databázi `Subject`, a jejich `value` je `id` daného předmětu. Tyto předměty jsou požadavkem, který vytvoří `useEffect` po otevření této stránky. Výchozí nastavení `selectu` jsou všechny články.

Kromě požadavku na předměty vytvoří `useEffect` i požadavek na všechny články. Kdykoli však uživatel změní `select`, tak jsou články automaticky přepsány podle toho,

jaký předmět uživatel zvolí. Pokud pod nějakým předmětem není žádný článkem, tak program vypíše none articles found viz. obr. 5.

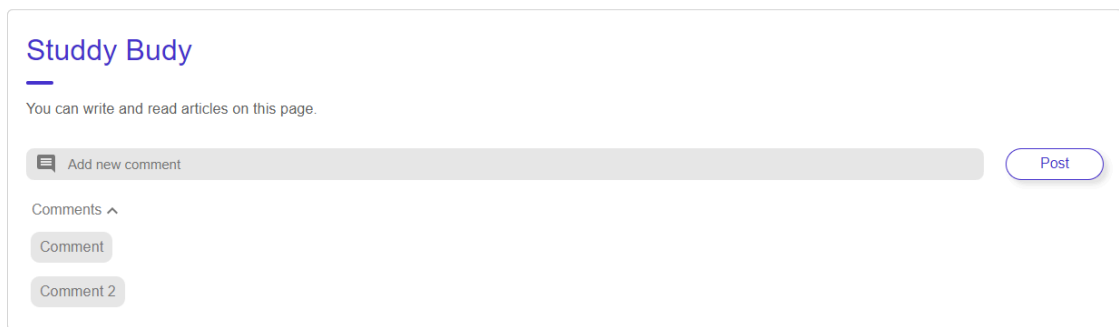


Obrázek 5: komponent articles

Články se vypisují jako komponent Card. Tento komponent potřebuje parametry title, text a id. title se zobrazí jako nadpis a text se zobrazí jako tělo článku. Jelikož text je je html překonvertované na string, tak musíme říct, že tento string je html.

Pod článkem je input pro napsání komentáře a tlačítko pro přidání komentáře. Id slouží jako id článku pod kterým je komentář napsán. Po zmáčknutí tlačítka se zavolá příkaz, který přidá článek do databáze.

Pod inputem se nachází tlačítko pro čtení komentářů. Po zmáčknutí se objeví všechny komentáře k danému článku. Pokud pod článkem nejsou žádné komentáře, tak program vypíše none comments found. Po přidání komentáře pod článek se automaticky zavolá příkaz na získání všech komentářů pod článkem. Na obrázku 6 můžete vidět celý komponent Card.

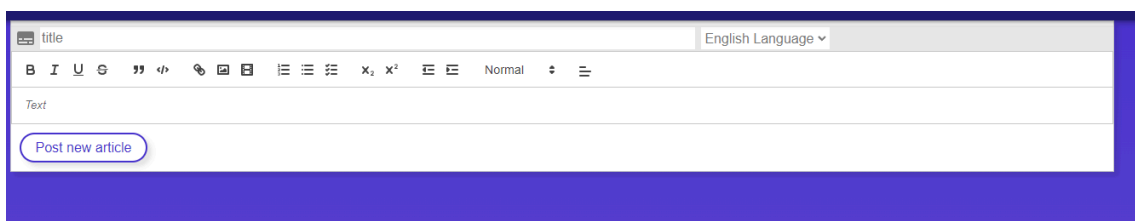


Obrázek 6: Článek s komentáři

## 4.6. Create Article

Komponent `create_article` se zavolá pod cestou `/user/article/create`. Po otevření se ukáže stránka s auditorem článků viz. obrázek 7. Uživatel musí zadat unikátní nadpis do kolonky table, který se bude ukazovat jako nadpis článku. Dále si může zvolit jeden z předmětů. Defaultně je nastaven anglický jazyk. Samotný editor textu umožňuje základní možnosti pro editaci textu, jako například vložit odkaz videa, vložit obrázek z počítače, změnit text, nebo zadat kód. Nastavení editoru najdete v proměnné `toolbarOptions`. Text a nadpis se ukládají, takže pokud by jste článek nedopsali, tak se vám na stránce uloží. Text z této metody je datového typu string, který obsahuje html komponenty.

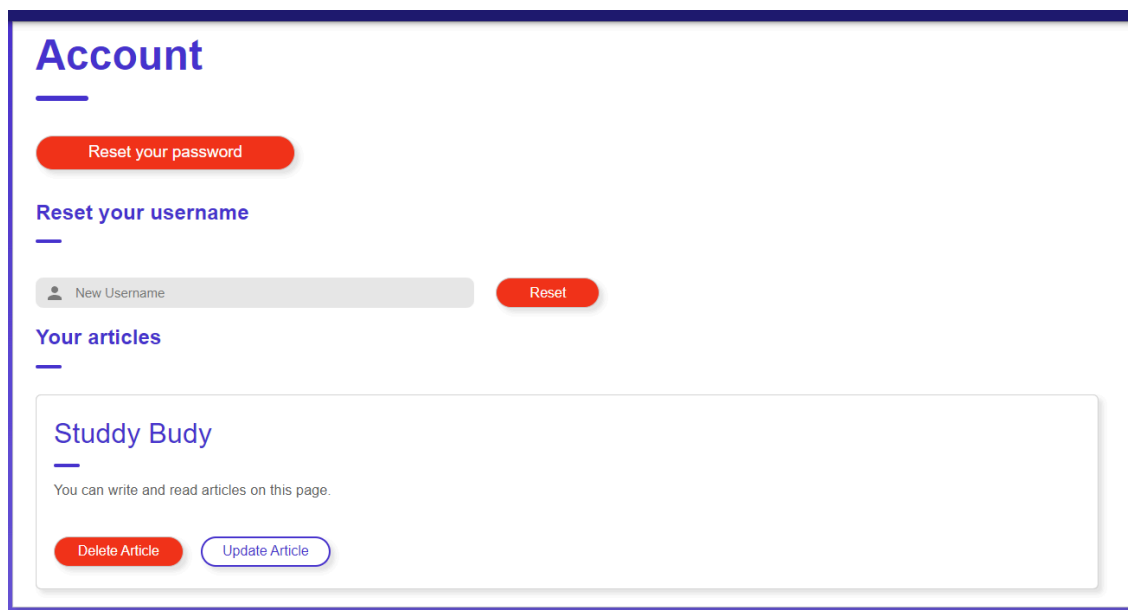
Pro přidání nového článku musí uživatel zmáčknout na tlačítku `Post new Article`. Tlačítko vytvoří požadavek na přidání článku a uživatele přesměruje na homepage.



Obrázek 7: Komponent pro tvorbu článku

## 4.7. Account

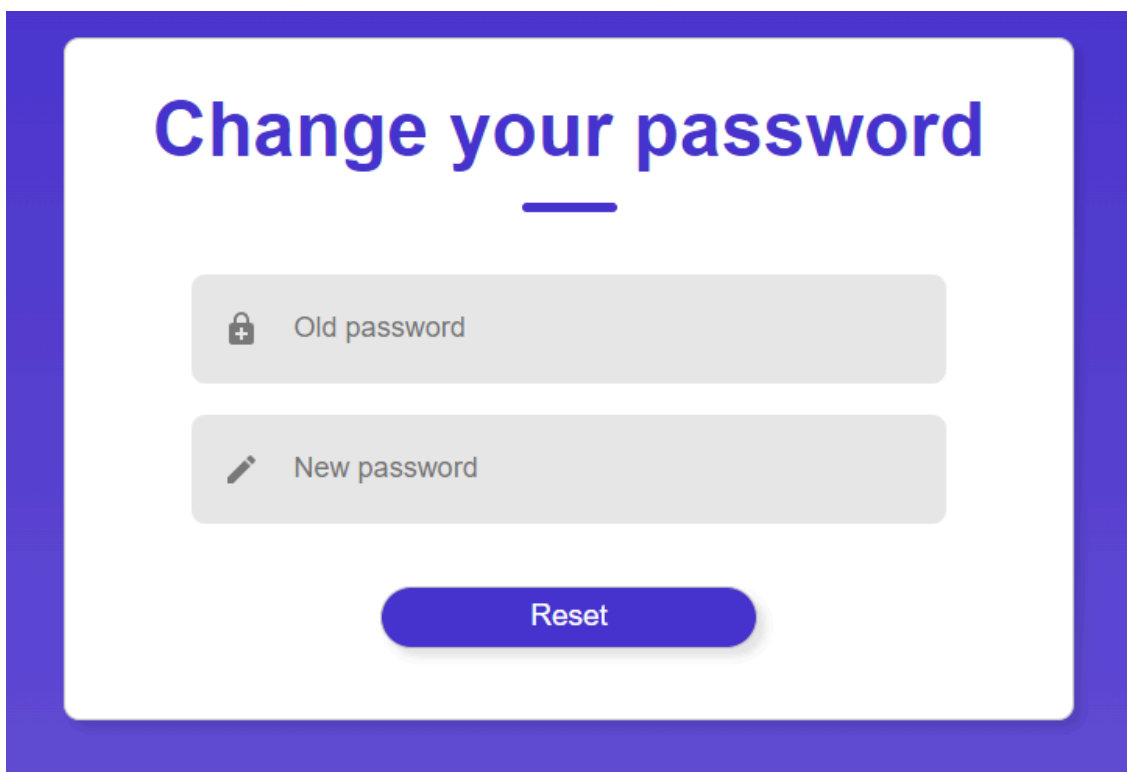
Komponent Account najdete pod cestou `/user/account`. Na stránce naleznete nastavení pro změnu hesla, jména a mazání a úpravu článků, které napsal daný uživatel. Tuto stránku můžete vidět na obrázku 8.



The screenshot displays the 'Account' page with a purple header. Below the header, there are three main sections: 'Reset your password' with a red button, 'Reset your username' with a text input field labeled 'New Username' and a red 'Reset' button, and 'Your articles' which contains a card for 'Studdy Buddy'. The card includes the text 'You can write and read articles on this page.' and two buttons: 'Delete Article' (red) and 'Update Article' (blue).

Obrázek 8: Komponent Account s jedním příspěvkem

Pro změnu hesla musí uživatel kliknout na tlačítko Reset your password. Po zmáčknutí na uživatele vyskočí stránka s dvěma inputy, old password a new password, a tlačítko reset. Pro změnu hesla musí uživatel zadat staré heslo a nové heslo a zmáčknout tlačítko Reset. Tuto stránku můžete vidět na obrázku 9.



## Change your password

---

Old password

New password

Reset

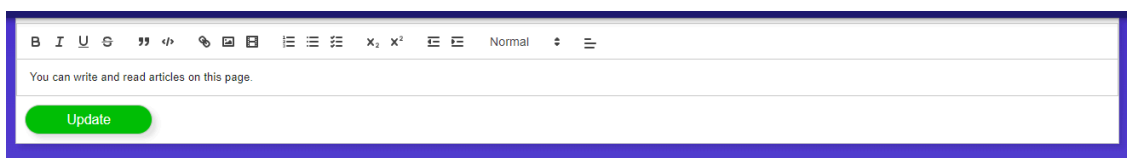
Obrázek 9: Resetování hesla

Změna uživatelského jména je možná udělat na samotné stránce po zadání jména do kolonky New Username a po zmáčknutí tlačítka Reset.

Pod nadpisem Your articles se vypíší všechny příspěvky, které napsal daný uživatel. Jestliže uživatel nenapsal žádné články, zobrazí se Po zmáčknutí na tlačítko delete article se daný článek smaže a po zmáčknutí na tlačítko Update Article stránka přesměruje uživatele na `/user/article/update`.

Na této stránce uživatel nalezne editor, uvnitř kterého je text článku, který chce uživatel změnit. Autor již nemůže změnit název a předmět. Po zmáčknutí se text uloží a uživatel je z této stránky přesměrován na homepage. Tuto stránku můžete vidět na obrázku 10.





Obrázek 10: Editace textu Článku StudyBuddy

## 5. Použité knihovny a nastavení aplikace

Kromě frameworku Spring a knihovny React, byly použity knihovny mui/icons-styled, mui/materials, bootstrap, react-dom, react-quill, react-router-dom, react-scripts, prop-types, axios, javovské knihovny, lombok, javax, postgresql, slf4j a web-vitals.

Nastavení aplikace naleznete v `application.properties`. Prvních 11 řádků slouží k nastavení propojení programu s databází a práci s databází. Pro funkčnost programu se musí na místo `username` zadat uživatel, na místo `password` heslo a na místo `url` url databáze postgres. Následující dvě řádky povolují po zapnutí ukazovat vše co proběhne v databázi. Šestá řádka nastavuje, jakou databázi chceme propojit s programem. Sedmá řádka určuje, co chceme dělat s tabulkami vytvořenými v aplikaci (např. `drop-create` znamená, že tabulky po zapnutí budou vždy vytvořeny pokud již neexistují a po vypnutí jsou všechny smazány, `create` znamená, že tabulky jsou jen, vytvořeny pokud již nebyly vytvořeny, nebo `update`, který jen updatuje databázi). Následující čtyři řádky vypisují errorry v databázi (aplikace vypisuje error i když budou nastaveny, tak aby errorry nevypisovali, jelikož aplikace chytá některé situace, které by se neměli stát a vypisuje je do konzole, tyto errorry jsou způsobeny díky metodám ve složce `validators`) a nastavení databáze, aby fungovala tak jak má.

Následující řádky (12-21) jsou určeny pro konfiguraci emailu. Jelikož jsem používal email od seznamu, tak je použitou `smtp.seznam.cz`. Pro jiné druhy emailových adres je potřeba použít jiné `smtp`. Pro funkčnost programu je nutné zadat email, přes který se budou posílat emaily, pod `username` a heslo tohoto emailu. Zbylé nastavení už jen nastavují cestu mezi emailem a zachytávají errorry.

## 6. Závěr

Aplikace obsahuje všechny kritéria zadání. Cílem této ročníkové práce bylo naučit se základy frameworku Spring, práci v teamu na jednom projektu, základy knihovny React a základní práci s databází.

## 7. Citace a zdroje

Material UI SAS, 2024. In: MUI Core [Online]. 2024. Dostupné z:

<https://mui.com/material-ui/>

Bootstrap team, 2024. In: Bootstrap [Online]. 2024. Dostupné z:

<https://getbootstrap.com/docs/5.2/getting-started/introduction/>

Jason Chen a Byon Milligan, 2024. In: Quill [Online]. 2024. Dostupné z:

<https://quilljs.com/docs/modules/toolbar/>

Broadcom Inc., 2024. In: Spring [Online]. 2024. Dostupné z:

<https://spring.io/why-spring>

Meta Open Source, 2024. In: React[Online]. 2024. Dostupné z: <https://react.dev/learn>

The project Lombok authors., 2024. In: Project Lombok [Online]. 2024. Dostupné z:

<https://projectlombok.org/>

Refsnes Data, 2024. HTML <select> Tag. In: W3Schools [Online]. 2024. Dostupné z:

[https://www.w3schools.com/tags/tag\\_select.asp](https://www.w3schools.com/tags/tag_select.asp)

GeeksforGeeks, 2024. Spring Security - Get the Current Logged in User Details with Database. In: Geeks for Geeks [Online]. 2024. Dostupné z:

<https://www.geeksforgeeks.org/spring-security-get-the-current-logged-in-user-details-with-database/>

Spring Boot Tutorial | Full Course, 2021. In: Youtube [online]. 10. 1. 2021 [2023-12-1].

Dostupné z: <https://www.youtube.com/watch?v=9SGDpanrc8U&t=5145s>. Kanál

uživatele Amigoscode

Java Tutorial - Complete User Login and Registration Backend + Email Verification, 2021. In: Youtube [online]. 17. 1. 2021 [2023-01-12]. Dostupné z:

<https://www.youtube.com/watch?v=QwQuro7ekvc>. Kanál uživatele Amigoscode

React tutorial for beginners, 2023. In: Youtube [online]. 9. 10. 2023 [2023-12-19].

Dostupné z:

[https://www.youtube.com/watch?v=hn80mWvP-9g&list=PLZPZq0r\\_RZOMQArzyI32mVndGBZ3D99XQ&index=1](https://www.youtube.com/watch?v=hn80mWvP-9g&list=PLZPZq0r_RZOMQArzyI32mVndGBZ3D99XQ&index=1). Kanál uživatele Bro Code

How to make Sign In & Sign Up From Using React JS. In: Youtube [online]. 14. 8. 2023 [2023-12-19]. Dostupné z: <https://www.youtube.com/watch?v=8QgQKRcAUvM>. Kanál uživatele GreatStack

Packaging a React.JS Application into Java Spring Boot Application. In: Youtube [online]. 21. 5. 2022[2024-1-9]. Dostupné z: [https://www.youtube.com/watch?v=\\_CLLw3QAUOE&t=1374s](https://www.youtube.com/watch?v=_CLLw3QAUOE&t=1374s). Kanál uživatele Wazoo Web Bytes

Create Responsive Navigation Bar With MaterialUI 5 and ReactJS | MaterialUI 5 Navbar | React Navbar. In: Youtube [online]. 8. 1. 2022[2024-3-17]. Dostupné z: <https://www.youtube.com/watch?v=lUkxSnJ7aDw>. Kanál uživatele Indian Coders

How to write a custom login page in Spring Security and Spring Boot - Java Brains. In: Youtube [online]. 22. 12. 2019 [2024-3-29]. Dostupné z: <https://www.youtube.com/watch?v=yoTohM2jYhs&t=587s>. Kanál uživatele Java Brains

Save State to LocalStorage & Persist on Refresh with React.js. In: Youtube [online]. 17. 3. 2022 [2024-3-30]. Dostupné z: <https://www.youtube.com/watch?v=rWfhwW9forg>. Kanál uživatele Colby Fayock

Master Media Queries And Responsive CSS Web Design Like Chameleon!. In: Youtube [online]. 12. 4. 2023 [2024-3-30]. Dostupné z: <https://www.youtube.com/watch?v=K24lUgcTOMs>. Kanál uživatele Slaying The Dragon