



Semestrální práce z předmětu
Programování v Javě

STUDENT INFORMATION SYSTEM

Vypracovala: Smirnova Elina

2020



Příručka uživatele

Studijní informační systém zajišťuje podporu základní studijní agendy. Systém je určen pro studenty a učitele vysokých škol.

Také pro správu je tam navíc **role administrátora**, který má vyšší práva a spravuje záznamy učitelů a studentů (může je přidávat či odebírat), také spravuje seznam předmětů.

ID	Username	First Name	Last Name	Date of Birth	Year	Study Program
8	petr	Petr	Novak	1999-01-16	1	Applied Informatics
9	anna	Anna	Svobodova	1998-06-03	1	Applied Informatics
10	testt	Student	Testovací	1998-06-12	4	Cybernetics

Obrazovka administrátora pro správu studentů

Po kliknutí na studenta v tabulce se načtou data v levé části, které admin může upravovat. Také admin tady má možnost smazat účet studenta po kliknutí pravou stranou myši na studenta v tabulce.

Z menu baru po kliknutí na Navigation je možné přejít na jinou obrazovku.

Obrazovky pro předměty a učitele fungují obdobným způsobem.

Informační systém umožňuje studentům:

1. *Prohlížení zapsaných předmětů (Enrolled subjects);*

Z dashboardu student může přejít na záložku "Subjects". V tabulce se zobrazí seznam aktuálních předmětů, které má zapsané.

2. *Přihlašování na zkoušky;*

Z dashboardu je možnost přepnout se na záložku "Exams". Nejdříve student musí vybrat předmět, pro který chce vypsat zkoušky.

Exams

Navigation Exit Help

Select subject

B36BQMAT1

Submit Clear

Date	Time	Classroom	Capacity	Occupied	
2020-05-25	14:00	Blahblah	60	2	Enroll

Pokud učitel vypsál termín zkoušky a označil ji "available to enroll", studentovi bude dovolen přístup přihlásit se ke zkoušce. Zobrazí se kód předmětu, název předmětu, datum, čas konání zkoušky, kapacita a místnost.

3. Prohlížení studijních výsledků;

Z dashboardu je možnost přepnout se na záložku "Study results". V tabulce se vypíšou všechny předměty, které student měl zapsané během studia a dostal z nich známku. Vypíše se kód předmětu, název předmětu, semester, počet kreditů a získaná známka.

4. Změna osobních údajů;

Personal Info

Navigation Exit Help

Student ID 9

Username anna

First Name Anna

Last Name Svobodova

Date of Birth 1998-06-03

Year 1

Study program Applied Informatics

Obtained credits 6

Address

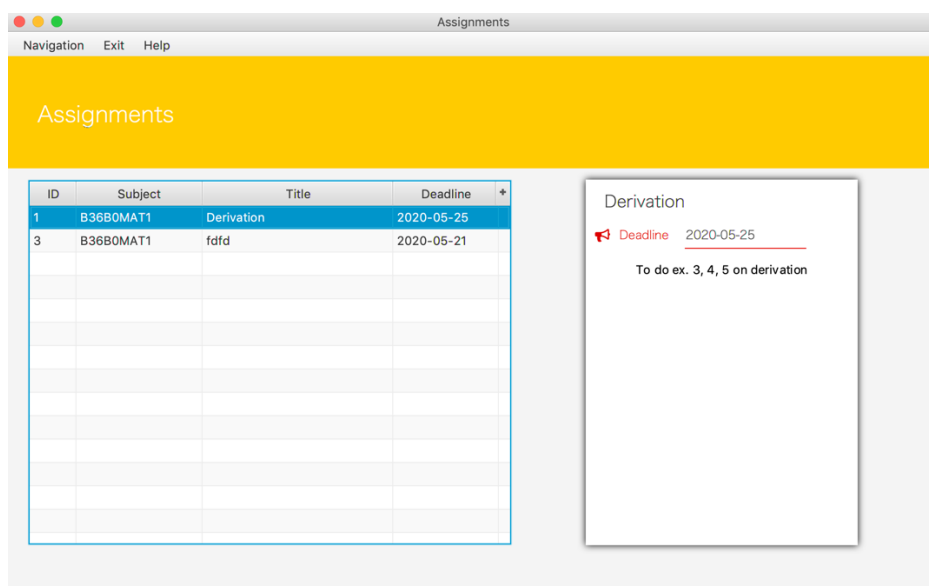
Phone number 776886689

Edit

Z dashboardu je možnost přepnout se na záložku "Personal info". Na stránce se vypíšou základní údaje o studentovi (jméno, příjmení, identifikační číslo studenta, ročník, adresa, telefonní číslo). Student má možnost změnit telefonní číslo a adresu.

5. Prohlížení domácích úkolů/zadání

Z dashboardu je možnost přepnout se na záložku "Assignments". Na stránce se zobrazí přehled aktuálních domácích úkolů, název předmětu a datum odevzdání.



Z dashboardu je možnost přepnout se na záložku "Assignments". Na stránce se zobrazí přehled aktuálních domácích úkolů, název předmětu a datum odevzdání.

Učitelé mají přístup k:

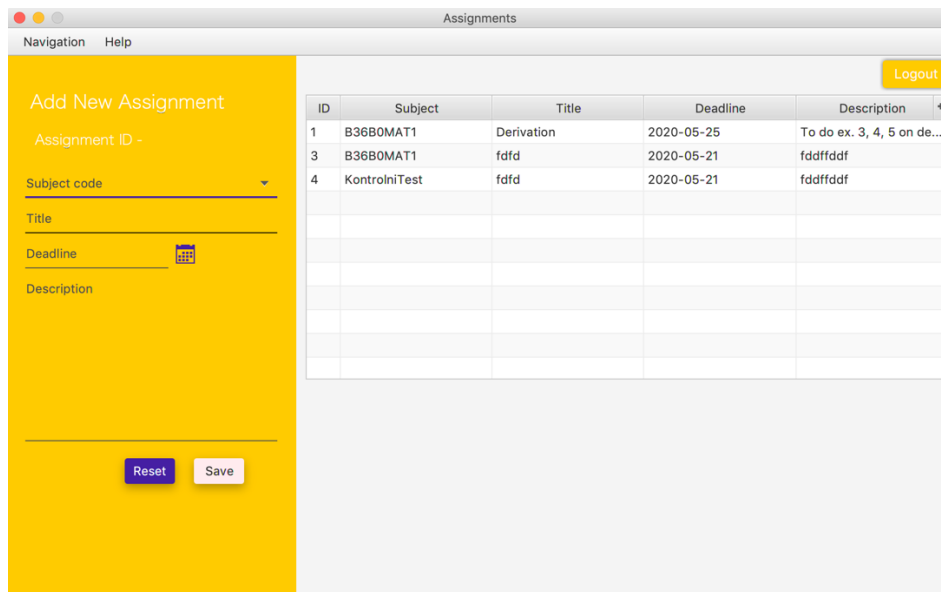
1. Prohlížení vyučovaných předmětů;

Z dashboardu je možnost přejít na záložku "Subjects". Na této stránce se zobrazí kód, název předmětu, počet kreditů za předmět, semestr. Po kliknutí na předmět se zobrazí seznam zapsaných na tento předmět studentů.

2. Správě vyučovaných předmětů;

Z předchozího bodu plyne možnost spravovat seznam studentů. Je možnost přidat/odebrat studenta, přidat/odebrat/změnit známku.

6. Vypisování domácích úkolů/zadání



ID	Subject	Title	Deadline	Description
1	B36B0MAT1	Derivation	2020-05-25	To do ex. 3, 4, 5 on de...
3	B36B0MAT1	fdff	2020-05-21	fdfffdff
4	KontrolniTest	fdff	2020-05-21	fdfffdff

Nastartování aplikace:

Aplikaci se dá spustit jako klasický maven projekt.

1. Po stažení otevřete projekt v IDE (IntelliJ IDEA, Netbeans atd).
2. V *properties* projektu nastavte verzi SDK na 1.8.
3. Stiskněte run (popřípadě na začátku tohoto kroku spusťte build a pak run).

Přihlašovací údaje:

- **Admin**
Username : admin1
Heslo: 1234
- **Student**
Username: anna
Heslo: 1111
- **Učitel**
Username: teacher1
Heslo: 0000

Technický popis projektu

Použité technologie a knihovny:

- Java 1.8
- JavaFX
- Spring Boot
- PostgreSQL
- JUnit
- FontAwesomeFx
- ControlsFx
- JFoenix (Material design)

Struktura aplikace

Nastavení aplikace a nastartování GUI je v balíčku **config**. V balíčku **controller** se nacházejí kontroléry pro jednotlivé obrazovky, odkud třeba můžu přistupovat ke formulářům. Model aplikace je obsažený v balíčku **model**. Tady se pracuje s hlavními entity jako Student, Subject, Teacher atd. Nad modelem je postavená **DAO** vrstva, která reprezentuje aplikační vrstvu a přistupuje k databázi. Hlavní funkcionality aplikace je pak obsažená v balíčku **service**, kde už se provádějí všechny CRUD operace. V balíčku **resources**, v souboru *application.properties* se nastavuje připojení k databázi. Tady se pracuje s databází PostgreSQL, ale dá se to rozběhnout i na localhostu. Také uvnitř balíčku **resources** je balíček **fxml**, kde jsou uloženy všechny obrazovky pro frontend. Poslední balíček je **view**, ve kterém je soubor formátu *enum*, odkud získávám pro vykreslování odkazy na jednotlivé obrazovky.

Backend:

Pro komunikaci s databází jsem se rozhodla používat framework Spring Boot, který zjednodušuje práci s JDBC tím, že zajišťuje správu transakcí, persistenci objektů a celkově korektně zachází s databází.

Frontend:

Frontendová část aplikace je napsána v JavaFX, se kterým se snadno pracuje pomocí SceneBuilderu.

