

# Uniwersytet Technologiczno-Przyrodniczy Im. J. J. Śniadeckich w Bydgoszczy Wydział Telekomunikacji, Informatyki i Elektrotechniki



Przedmiot	Algorytmy i struktury danych		
Prowadzący	dr inż. Michał Kruczkowski		
Temat	Aplikacja połączona z bazą danych z GUI		
Student	Mateusz Kalksztejn		
Email	Matkal003@utp.edu.pl		
Data wykonania	19.04.2019	Data oddania	15.05.2019

# Wprowadzenie

#### Cel pracy

Projekt ma za zadanie pogłębić wiedzę w zakresie tworzenia aplikacji GUI. Idea projektu jest stworzenie aplikacji, która może służyć jako wsparcie organizacyjne dla firmy i ich pracowników. Interfejs ma być przejrzysty, intuicyjny dla odbiorcy.

## Link do repozytorium

https://github.com/Karatonik/Algorytmy-projekt

#### Założenia

W projekcie ma występować

Dwa typu wprowadzania: Wprowadzenie całego rekordu do bazy danych, edycja rekordu z bazy danych

Aplikacja jest połączona z bazą danych SQLite. Wyszukiwanie po kolumnach.

Wykorzystanie javaFX do wizualizacji aplikacji. Edycja na tabelce bazodanowej w kolumnie.

Sortowanie z wykorzystaniem wyrażenia Implementacja do projektu transakcji.

lambda.

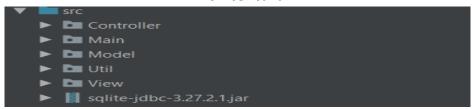
Użyte technologie

Java 8 JavaFX

**SQLite** 

# Projekt i implementacja

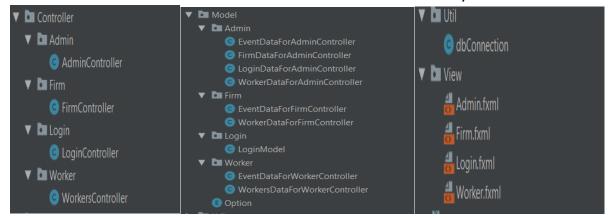
#### Architektura



Projekt jest podzielony na elementy takie jak

Controller -element kontrolujący okno Model-element zawierający model obiektu Util -elementy użytkowe (połączenie z bazą danych) View -plik fxml Main-klasa startowa

## Controller Model Util/View



### Controller

Zawiera metody i zmienne służące obsłudze okna i opcji aplikacji takich jak -Wyświetlanie, dodawanie, filtrowanie, aktualizowanie rekordów -Obsługa transakcji bazy danych

#### Model

Klasy służące do tworzenia z nich obiektów. W tej aplikacji służą do tworzenia list Option jest to klasa wyliczeniowa ( Enum ) służąca do obsługi pola wyboru .

#### Util

Zawiera klasę dbConnection służącą zapewnieniu połączenia z bazą danych

#### View

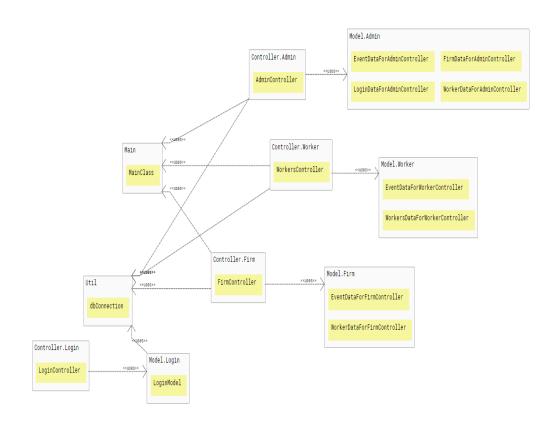
Pliki z rozszerzeniem fxml służące do oprawy graficznej

## Załączone w projekcie

sqlite-jdbc-3.27.2.1.jar jest Biblioteka konieczną do prawidłowego użytkowania w aplikacji z bazy danych.

test.sqlite plik bazy danych

Schemat UML -Paczki



Schemat UML -dziedziczenie



#### AdminController

- combodiv : ComboBox<Option>
- conn : Connection
- data : ObservableList<WorkerDataForAdminController>
- dataev : ObservableList<EventDataForAdminController>
- dataf : ObservableList<FirmDataForAdminController> 11 more

# FirmController

- datap : ObservableList<MorkerDataForFirmController>
- dataproj : ObservableList<EventDataForFirmController
- dataw : ObservableList<EventDataForFirmController>
- dc : dbConnection
- 10 more

#### LoginController

- ~ loginModel : Model.Login.LoginModel
- + mamef : String
- password : PasswordField

#### WorkersController

- + Logout : Button
- conn : Connection
- data : ObservableList<WorkersDataForWorkerController>
- datap : ObservableList<EventDataForWorkerController>
- dataw : ObservableList<EventDataForWorkerController>

#### Application / MainClass

#### EventDataForAdminController

- Date : StringProperty
- ID\_Event : StringProperty
- name\_Event : StringProperty

#### FirmDataForAdminController

- ID\_Firmy ; StringProperty
- Nazwa\_Firmy : StringProperty

#### LoginDataForAdminController

- division : StringProperty
- pass : StringProperty
- username : StringProperty

#### WorkerDataForAdminController

- DOB : StringProperty
- ID : StringProperty
- ID\_Firmy : StringProperty
- email : StringProperty - firstName : StringProperty

# 1 more

- EventDataForFirmController - Date : StringProperty
- ID\_Event : StringProperty
- ID\_Firmy : StringProperty
- name\_Event : StringProperty - pw : StringProperty

# WorkerDataForFirmController

- DOB : StringProperty
- ID\_Firmy : StringProperty
- ID\_Pracownika : StringProperty - Pesel : StringProperty
- Stanowisko : StringProperty

#### 3 more

LoginModel - connection : Connection

#### EventDataForWorkerController

- Date : StringProperty
- ID\_Event : StringProperty
- ID\_Firmy : StringProperty
- Status : StringProperty
- name\_Event : StringProperty
- 1 more

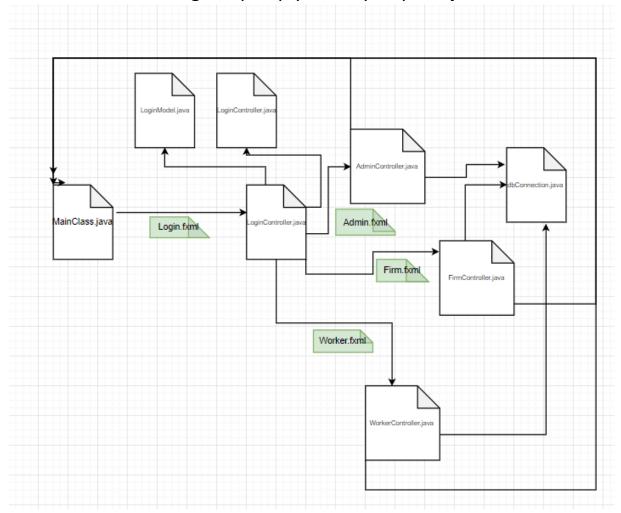
#### WorkersDataForWorkerController

- DOB : StringProperty
- ID\_Firmy : StringProperty
- ID\_Pracownika : StringProperty
- Pesel : StringProperty - Stanowisko : StringProperty

#### 3 тоге

#### ${\tt dbConnection}$

# Diagram przepływów Cykl aplikacji



#### Wnioski

Projekt się znacząco rozrósł, aktualnie jest to już ponad 2000 linijek przy pominięciu plików fxml. Listy wyparły z tablice. Są o wiele bardziej użyteczne. Co do samego kodu , wbrew pozorom najwięcej czasu zajęło mi stworzenie działającego GUI (pierwszy kontakt z JavaFX) i walidacja inputu użytkownika a nie backend i zarządzanie danymi. Do przechowywania danych istnieje wiele gotowych bibliotek zawierających mnóstwo przydatnych funkcji natomiast sprawdzanie danych wprowadzonych przez użytkownika należy dopasować pod swój program.