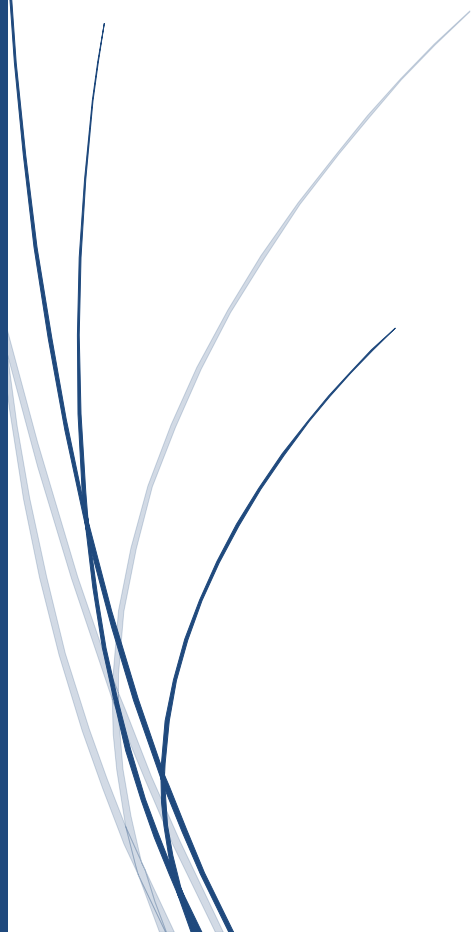




Autor: Krzysztof Małysa

# Kolekcja monet

Aplikacja do zarządzania kolekcją  
monet

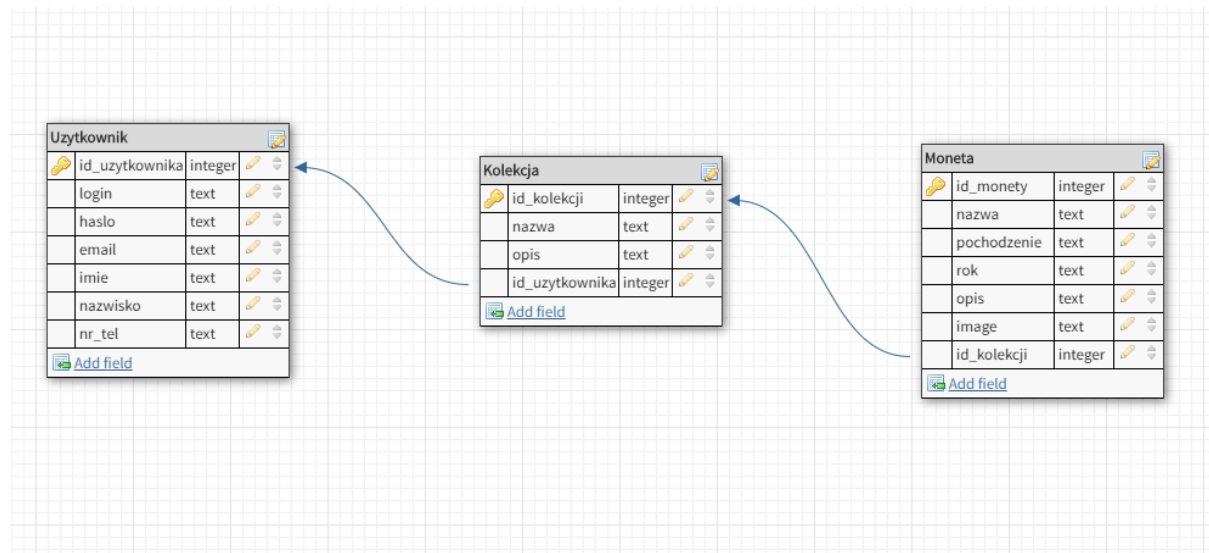


**Użyte programy/pliki:**

- NetBeans IDE 8.2,
- JavaFX Scene Builder Gluon,
- SQLite,
- Baza danych wykonana w sqlite,
- Sterownik do komunikacji z bazą sqlite,
- Biblioteki do obsługi wysyłania E-maila,
- Biblioteka jfoenix i TrayNotification

Mój projekt dotyczy zarządzania kolekcją monet. Aplikacja jest przeznaczona dla użytkowników. Jest możliwa edycja wszystkich danych (usuwanie,dodawanie,edytowanie).

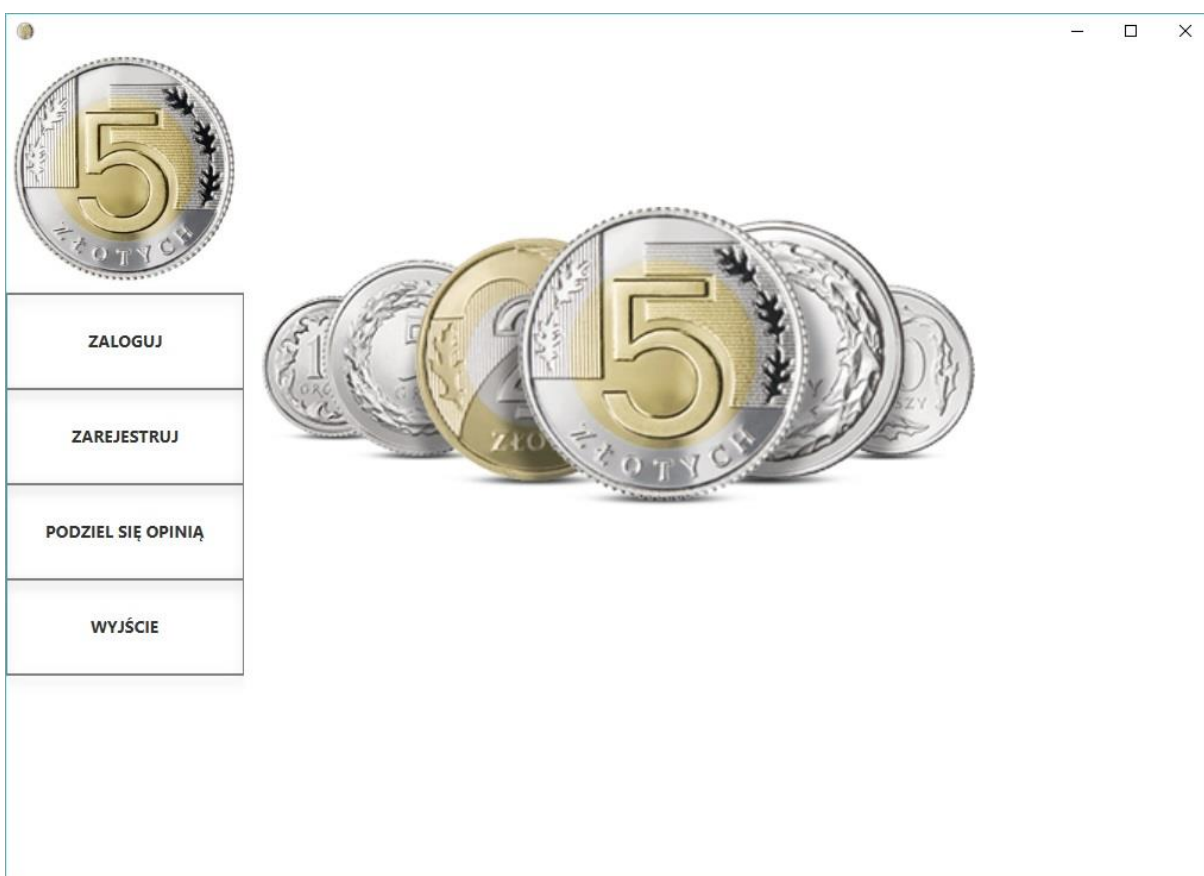
Schemat mojej bazy:



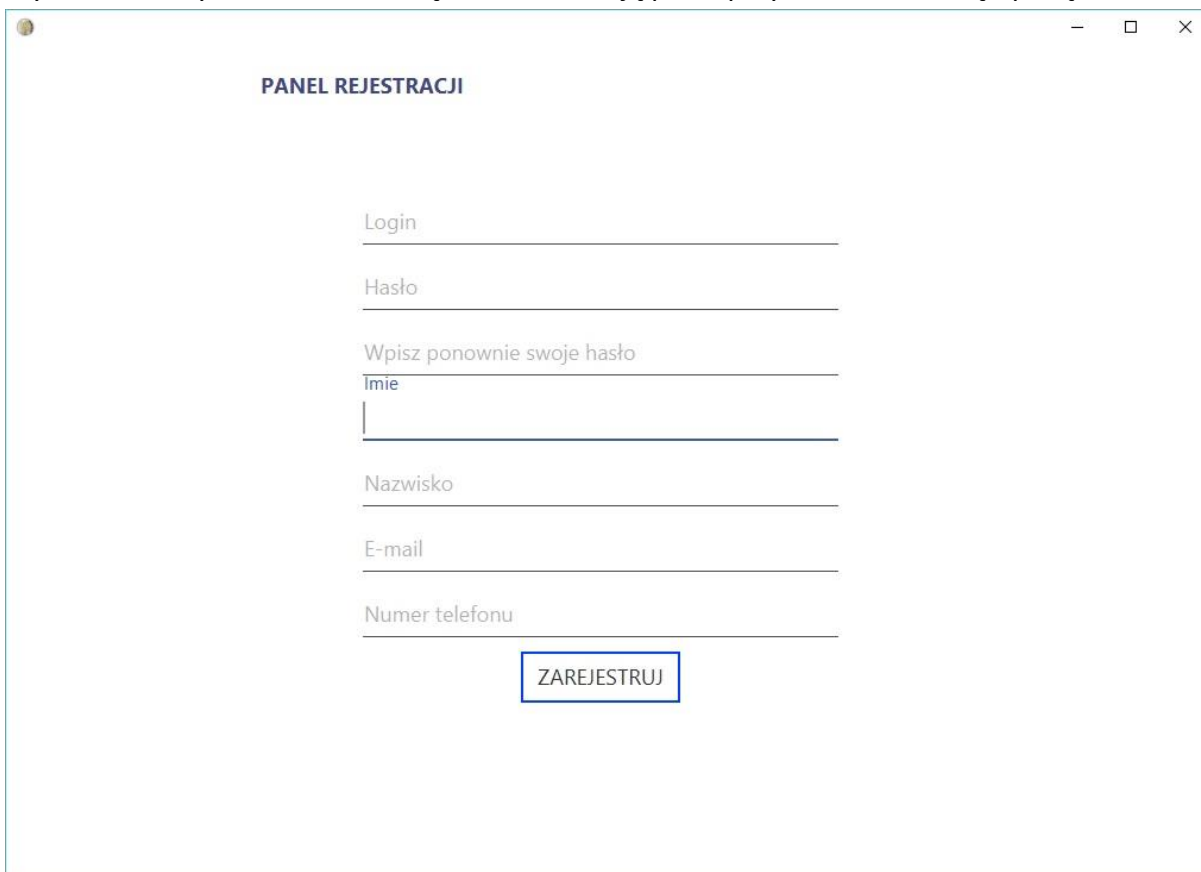
Po uruchomieniu aplikacji naszym oczom ukazuje się następujące okno :



Aby wywołać panel boczny (menu) należy najechać kursorem bliżej lewej krawędzi.

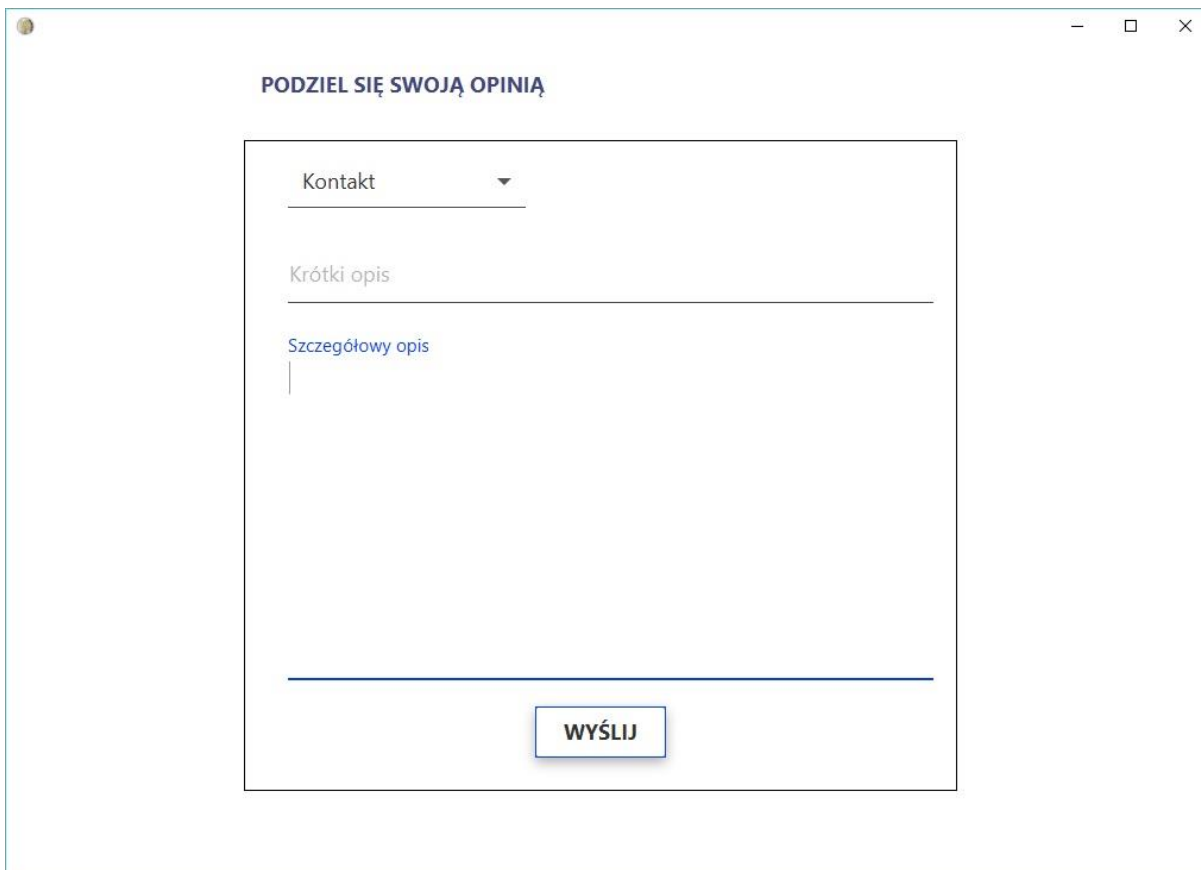


W menu głównym mamy możliwość zalogowania się na swoje konto, dokonania rejestracji czy też możemy skontaktować się z administracją przez przycisk „Podziel się opinią”.



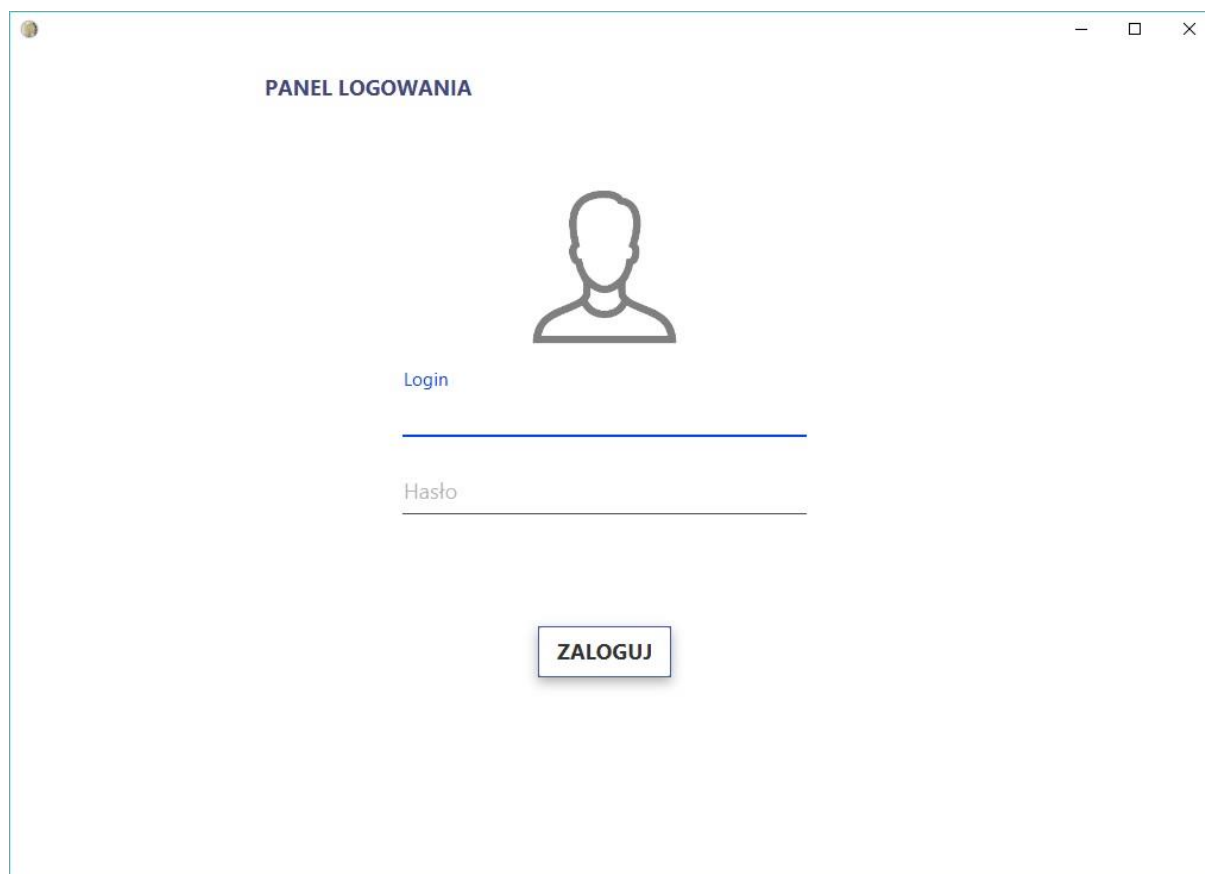
The screenshot shows a web application window titled "PANEL REJESTRACJI". It contains a registration form with the following fields: "Login", "Hasło", "Wpisz ponownie swoje hasło", "Imie", "Nazwisko", "E-mail", and "Numer telefonu". Each field is represented by a text input box with its label above it. Below the "Numer telefonu" field is a blue button labeled "ZAREJESTRUJ".

Powyższy screen ukazuje formularz rejestracji nowego użytkownika.



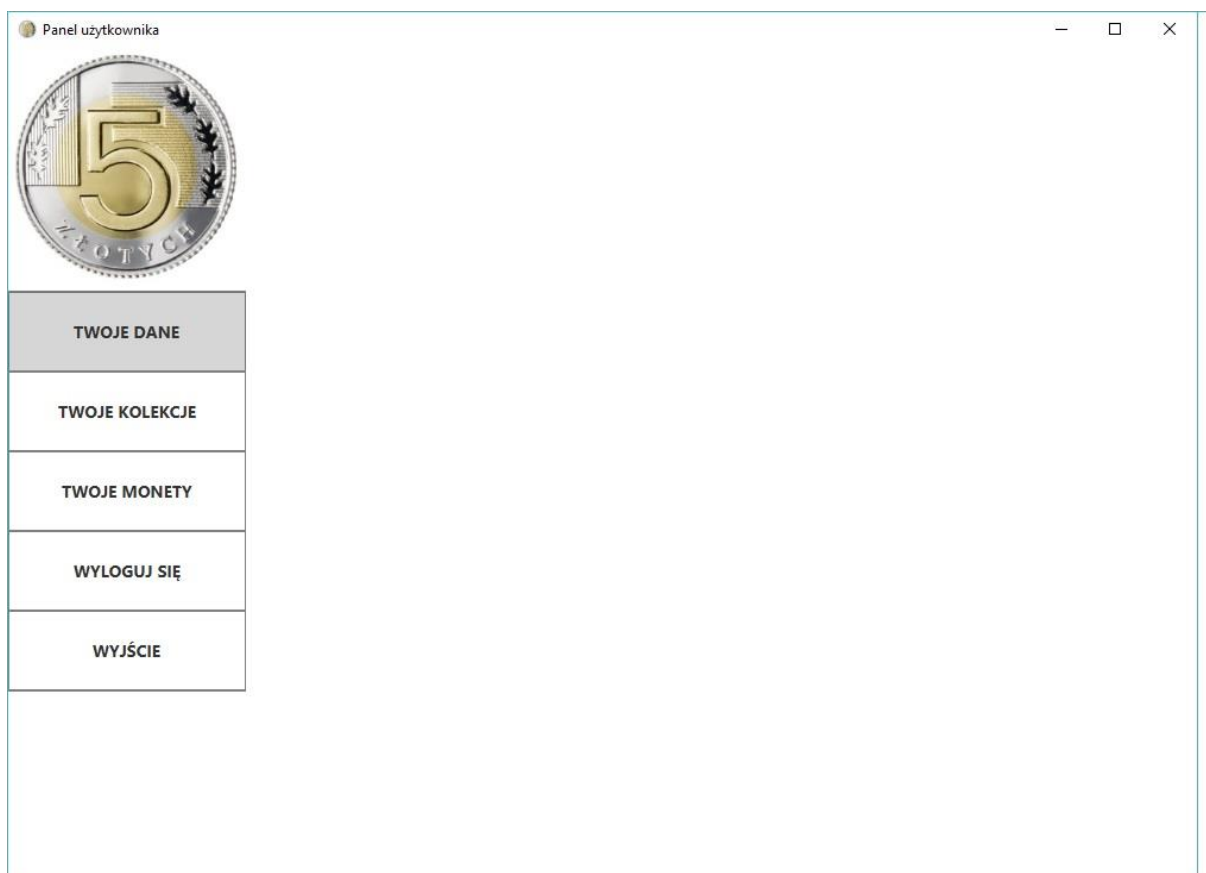
The screenshot shows a web application window titled "PODZIEL SIĘ SWOJĄ OPINIĄ". It contains a feedback form with the following elements: a dropdown menu labeled "Kontakt", a "Krótki opis" (Short description) text input field, and a "Szczegółowy opis" (Detailed description) text area. At the bottom of the form is a blue button labeled "WYŚLIJ".

Tutaj możemy skorzystać z możliwości wystąpienia swojej opinii, zgłoszenia błędu czy też kontaktu z autorem.



The image shows a web application window titled "PANEL LOGOWANIA". At the top center is a line-art icon of a person's head and shoulders. Below the icon are two input fields: the first is labeled "Login" and the second is labeled "Hasło". Both fields have a blue underline. Below the "Hasło" field is a button labeled "ZALOGUJ". The window has a standard title bar with a close button (X) in the top right corner.

Kolejnym oknem jest okno logowania, wystarczy wpisać odpowiedni login oraz hasło i kliknąć „Zaloguj”.



Po zalogowaniu widzimy okno przedstawione powyżej. Z tego poziomu mamy dostęp do edycji swoich danych wpisanych przy rejestracji, naszych kolekcji oraz naszych monet. Ponadto możemy się wylogować bez wyłączania aplikacji.

**TWOJE DANE**

test

...

...

Jaś

Kowalski


jkowalski@gmail.com

123123123

ZAPISZ

„Twoje dane” umożliwia edycję naszych danych. Możemy wprowadzić zmiany i je zapisać.

Panel użytkownika



TWOJE DANE

TWOJE KOLEKCJE

TWOJE MONETY

WYLOGUJ SIĘ

WYJŚCIE

TWOJE KOLEKCJE

Nazwa kolekcji	Opis
Historyczne	Zbiór monet związanych z ważnymi wydarzeniami historycznymi.
Narodowe	Zbiór monet NBP.

DODAJ KOLEKCJĘ


EDYTUJ KOLEKCJĘ

USUŃ KOLEKCJĘ

Po kliknięciu w opcję „Twoje kolekcje” mamy dostęp do naszych kolekcji. Możemy je dodawać, edytować oraz usuwać. Aby przyciski edytuj oraz usuń stały się aktywne wystarczy wybrać dowolną pozycję z tabeli.



Panel użytkownika



TWOJE KOLEKCJE

TWOJE DANE

TWOJE KOLEKCJE

TWOJE MONETY

WYLOGUJ SIĘ

WYJŚCIE

Nazwa kolekcji

Krótki opis kolekcji

DODAJ KOLEKCJĘ

ANULUJ

Powyższy screen przedstawia okno dodawania nowej kolekcji.

Panel użytkownika



TWOJE KOLEKCJE

TWOJE DANE

TWOJE KOLEKCJE

TWOJE MONETY

WYLOGUJ SIĘ

WYJŚCIE

Narodowe


Zbiór monet NBP.

ZAPISZ KOLEKCJĘ

ANULUJ

Tak jak wyżej, jednak ten screen dotyczy edycji wybranej wcześniej kolekcji.

Panel użytkownika



## TWOJE MONETY

Wybierz kolekcję

Historyczne

Narodowe

Nazwa monety	Pochodzenie	Rok	Opis
No content in table			

TWOJE DANE

TWOJE KOLEKCJE

**TWOJE MONETY**

WYLOGUJ SIĘ

WYJŚCIE


DODAJ MONETĘ

EDYTUJ MONETĘ

USUŃ MONETĘ

Po przejściu do opcji „Twoje monety” możemy zarządzać monetami z kolekcji. Na górze z ComboBoxa wybieramy kolekcję, na której będziemy wykonywać działania. Po wybraniu interesującej nas kolekcji, w tabeli wyświetlą się przypisane do niej monety. Możemy wykonać w tym momencie dodawanie monet, ich edycję czy też usuwanie.

Panel użytkownika



TWOJE DANE

TWOJE KOLEKCJE

TWOJE MONETY

WYLOGUJ SIĘ

WYJŚCIE

TWOJE MONETY

Nazwa monety

Piątak

Pochodzenie

Polska

Rok

2010

Krótki opis

Znalezione nie kradzione XD


WYBIERZ ZDJĘCIE

DODAJ MONETĘ

ANULUJ

Aby dodać monetę należy uzupełnić wszystkie pola oraz wybrać zdjęcie.

Panel użytkownika



TWOJE DANE

TWOJE KOLEKCJE

TWOJE MONETY

WYLOGUJ SIĘ


WYJŚCIE

TWOJE MONETY

Wybierz kolekcję

Narodowe

Nazwa monety	Pochodzenie	Rok	Opis
Piątak	Polska	2010	Znalezione nie kradzione XD



DODAJ MONETĘ

EDYTUJ MONETĘ


USUŃ MONETĘ

Tak prezentuje się lista monet, przypisanych do kolekcji „Narodowe”. Po kliknięciu w pozycję, mamy podgląd zdjęcia wcześniej przypisanego do tej monety.

10

Panel użytkownika

## TWOJE MONETY



TWOJE DANE
TWOJE KOLEKCJE
<b>TWOJE MONETY</b>
WYLOGUJ SIĘ
WYJŚCIE

Piątak

Polska

2010

Znalezione nie kradzione XD

WYBIERZ ZDJĘCIE

ZAPISZ MONETĘ

ANULUJ

Powyższe okno jest edycją monety, wygląda analogicznie jak okno dodawania.



Tak prezentują się powiadomienia systemowe w aplikacji.

Teraz opiszę kilka wybranych fragmentów kodu.

Poniżej pokazana jest klasa Polaczenie, która łączy aplikację javafx z bazą danych.

```

public class Polaczenie {

    private static final String DRIVER = "org.sqlite.JDBC";

    public static Connection Connect() {
        try {
            String url = "jdbc:sqlite:Monety";
            String user = "";
            String password = "";

            Class.forName(DRIVER);
            Connection conn = DriverManager.getConnection(url);
            return conn;
        } catch (ClassNotFoundException | SQLException ex) {
            Logger.getLogger(Polaczenie.class.getName()).log(Level.SEVERE, null, ex);
        }
        return null;
    }
}

```

```

public class KolekcjaDetails {

    private final IntegerProperty id_kolekcji;
    private final StringProperty nazwa;
    private final StringProperty opis;

    public KolekcjaDetails(int id_kolekcji, String nazwa, String opis) {
        this.id_kolekcji = new SimpleIntegerProperty(id_kolekcji);
        this.nazwa = new SimpleStringProperty(nazwa);
        this.opis = new SimpleStringProperty(opis);
    }

    public int getId_kolekcji(){
        return id_kolekcji.get();
    }

    public String getNazwa(){
        return nazwa.get();
    }

    public String getOpis(){
        return opis.get();
    }

    public void setNazwa(String value){
        nazwa.set(value);
    }

    public void setOpis(String value){
        opis.set(value);
    }

    public IntegerProperty id_kolekcjiProperty(){
        return id_kolekcji;
    }

    public StringProperty opisProperty(){
        return opis;
    }

    public StringProperty nazwaProperty(){
        return nazwa;
    }

}

```

Powyższy screen zawiera kod klasy, obsługujący tabelę Kolekcja z bazy danych.

```

@FXML
private void wybierzZdjecie(ActionEvent event) {
    filechooser = new FileChooser();
    filechooser.setTitle("Open File");
    filechooser.getExtensionFilters().add(new ExtensionFilter("Image Files", "*.png", "*.jpg", "*.gif"));
    file = filechooser.showOpenDialog(new Stage());
    sciezka = file.getAbsolutePath();
    File f1 = new File(sciezka);
    Image i = new Image(f1.toURI().toString());
    //IV.setImage(i); //imageView
}

```

I ostatni screen, prezentuje wywołanie okienka ładującego obrazek.