



**Politechnika Krakowska**  
**im. Tadeusza Kościuszki**

Projekt z przedmiotu:

***Programowanie w języku JAVA***

Temat Projektu:

***System wspomagający sprzedaż biletów kolejowych***

*Raport II*

*Wykonali:*

*Pancerz Marcin*

*Polak Jacek*

*rok akademicki 2016/2017*

# *Spis treści*

1. Wstęp .....	3
2. Postępy Prac .....	3
2.1 Klient/Serwer .....	3
2.1.1 Aplikacja Serwera .....	3
2.1.2 Aplikacja Klientcka .....	4
2.1.3 Panel Logowania .....	4
2.1.4 Panel Rejestracji .....	4
2.1.5 Panel Użytkownika .....	4
2.1.7 Moduł Daty i Czasu .....	5
2.1.8 Baza danych .....	5
3. Planowane funkcjonalności .....	5
3.1 Wyszukiwarka połączeń .....	5
3.2 Moduł rezerwacji i zakupu biletów .....	5
3.3 Rozkład Dworca .....	5
3.4 Panel Informacyjny .....	5
3.5 Panel Pomocy .....	5
3.6 Panel administracyjny .....	5
3.7 Panel rezerwacji biletów .....	5
3.8 Panel historii zakupionych biletów .....	5
4. Technologia JavaFx i wynikające z niej kwestie .....	6
5. Diagram Klas .....	6
6. Wygląd Aplikacji .....	7
6.1 Menu główne .....	7
6.2 Panel Logowania .....	8
6.3 Panel Rejestracji .....	9
6.4 Panel Użytkownika .....	10
6.5 Panel Profil Użytkownika .....	11

# ***1. Wstęp***

Niniejszy dokument prezentuje postępy prac w tworzeniu aplikacji wspomagającej sprzedaż biletów kolejowych. Zostały w nim opisane funkcjonalności, które na dzień tworzenia dokumentu udało się zaimplementować. Załączony został także diagram klas, wstępnie opisujący strukturę aplikacji. Zwrócona została także uwaga na funkcjonalności, które są w planach do wykonania. W dokumencie zawarte są również screen-shoty prezentujące wygląd aplikacji.

## ***2. Postępy Prac***

Zgodnie z założeniami projektu, aplikacja powstaje w języku JAVA. Została ona oparta o architekturę klient/serwer. Implementacja aplikacji wsparta została przez technologię JavaFX co zdecydowanie usprawnia pracę, jak również polepsza jej efekt końcowy. Aplikacja współpracuje z bazą danych za pośrednictwem serwera.

## ***Wykonano***

### ***2.1 Klient/Serwer***

Aplikacja pisana jest i rozwijana jako dwa człony.

#### ***2.1.1 Aplikacja Serwera***

Aplikacja serwera jest pomostem łączącym aplikację kliencką, która korzysta z funkcjonalności serwera, z bazą danych. Użytkownik zgłasza zapytanie do serwera, a ten w odpowiedni sposób na nie odpowiada. Procesy serwera obsługiwane są wielowątkowo, dzięki czemu możliwym jest aby więcej niż jeden użytkownik korzystał z usług serwera. Na ten moment w aplikacji serwera stworzono kilka klas z nią związanych. Wśród nich można wyodrębnić dwie kategorie. Pierwszą są klasy odpowiedzialne za utworzenie i uruchomienie serwera. Drugą kategorią są klasy odpowiedzialne za współpracę z bazą danych.

### **2.1.2 Aplikacja Klientka**

Aplikacja kliencka jest platformą, z którą bezpośredni kontakt ma klient. Za jej pomocą dokonuje on wszystkich interesujących go procesów, tak więc ważnym jest żeby miała ona przyjemny i wygodny w użyciu interfejs. Dzięki zastosowaniu technologii JavaFx i możliwości stylizacji wyglądu za pomocą plików CSS, kreowanie wyglądu aplikacji staje się bardzo łatwe i przyjemne. Oddzielenie wyglądu aplikacji od jej ścisłej logiki staje się bardzo wydajną i pomocną praktyką.

Ze względu na zastosowanie technologii JavaFx w procesie tworzenia aplikacji można wyróżnić

### **2.1.3 Panel Logowania**

Udało się wykonać w pełni działający Panel Logowania użytkownika. Proces logowania został dokładnie zabezpieczony. W trakcie logowania aplikacja kliencka wysyła zapytanie do serwera czy dany użytkownik jest zarejestrowany, a następnie odbywa się proces walidacji wprowadzonych przez użytkownika danych. Ważną kwestią okazało się zaimplementowanie sprawdzania poprawności struktury adresu e-mail do tego celu została wykorzystana klasa `EmailValidator` – Apache Commons. Cały proces rejestracji opatrzone zostały komunikatami informującymi użytkownika o wynikłym problemie. Gdy proces logowania przebiegnie pomyślnie, użytkownik przenoszony jest do Panelu Użytkownika.

### **2.1.4 Panel Rejestracji**

Udało się zrealizować również Panel Rejestracji użytkownika. W panelu tym użytkownik proszony jest o podanie danych do rejestracji. Dane są poddane procesowi walidacji, podobnie jak w procesie logowania, wykorzystana została klasa `EmailValidator` – Apache Commons. Hasło użytkownika również musi składa się z określonej ilości znaków, jeśli ilość ta okaże się nie poprawna użytkownik otrzyma adekwatny komunikat. Gdy użytkownik pomyślnie przejdzie ten proces, przenoszony jest do Panelu Użytkownika, jeśli nie, proszony jest o poprawienie pól formularza powodujących błąd.

### **2.1.5 Panel Użytkownika**

Panel użytkownika jest na etapie ciągłego rozwoju. Na ten moment udało zrealizować się funkcję wylogowywania użytkownika.

Przejsie do Profilu użytkownika - funkcjonalność opisana w dalszej części dokumentu.

Reszta funkcjonalności w planach.

### ***2.1.6 Profil użytkownika***

Panel ten skupia się na danych personalnych użytkownika. W celach informacyjnych wyświetla je, jak również umożliwia ich edycję. Panel ten współpracuje z bazą danych, w celu zmienienia danych użytkownika. Adres e-mail wyłączony jest z możliwości zmiany, gdyż jest on identyfikatorem użytkownika(Login) , który nie może ulec zmianie.

### ***2.1.7 Moduł Daty i Czasu***

W aplikacji został zaimplementowany moduł odpowiedzialny za bieżącą datę i czas. Jest on uruchamiany jako osobny wątek, z którego każdy z kontrolerów, za pomocą odpowiednich metod może pobierać potrzebne dane. Następnie dane te wyświetlane są na bieżącym panelu użytkownika. We wszystkich panelach załączony jest widok na bieżącą datę i czas.

### ***2.1.8 Baza danych***

Baza danych jest systematycznie rozwijana. Jej ogólna struktura została już utworzona.

## ***3.Planowane funkcjonalności***

### ***3.1 Wyszukiwarka połączeń***

### ***3.2 Moduł rezerwacji i zakupu biletów***

### ***3.3 Rozkład Dworca***

### ***3.4 Panel Informacyjny***

### ***3.5 Panel Pomocy***

### ***3.6 Panel administracyjny***

### ***3.7 Panel rezerwacji biletów***

### ***3.8 Panel historii zakupionych biletów***

## ***4. Technologia JavaFx i wynikające z niej kwestie***

W procesie tworzenia aplikacji zastosowano technologię JavaFx korzystającą z wzorca architektonicznego Model-View-Controller. Wynika z tego, że powstało i powstanie wiele klas odpowiedzialnych za taki podział, są to klasy Modelu, klasy Kontrolerów jak również pliki FXML odpowiedzialne za wygląd aplikacji. Dodatkowym plikiem jest arkusz stylu CSS, który ułatwia modelowanie wyglądu.

Z racji wymogu odświeżania widoków, w ramach reakcji na zmiany wynikające z działań modelu, w aplikacji wykorzystano wzorzec projektowy *Obserwator* - ułatwia to znacznie system odświeżania widoków.

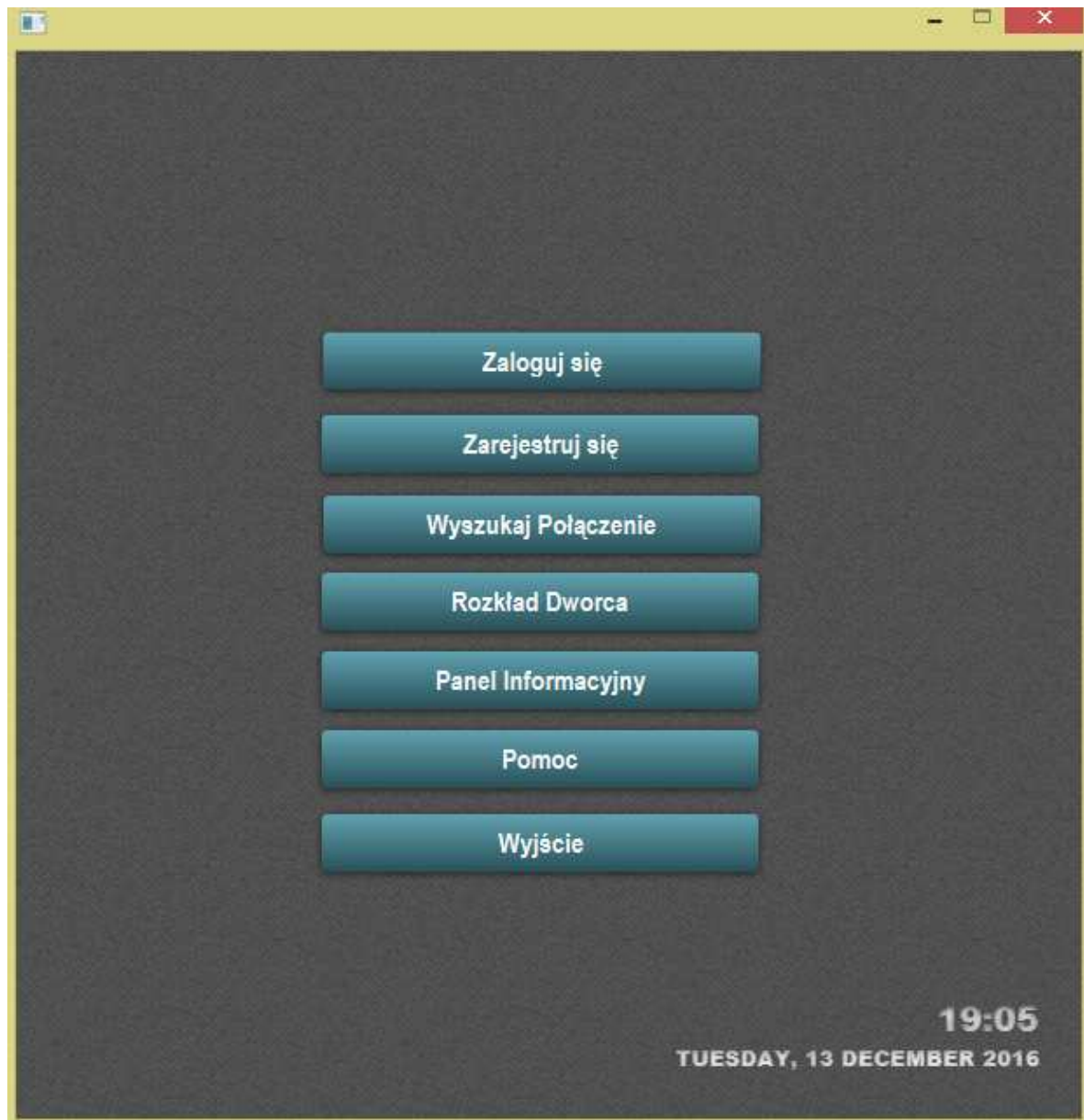
## ***5. Diagram Klas***

Diagram klas prezentuje zbiór klas wewnętrznych aplikacji - poprzez sformułowanie "wewnętrznych" rozumie się klasy związane tylko z ogólnie rozumianym problemem rezerwacji i sprzedaży biletów kolejowych, a więc te związane z aplikacją kliencką, ale w formie logiki a nie zastosowanej technologii JavaFx. Tak więc wszystkie klasy związane ze wzorcem architektonicznym Model-View-Controller nie zostały zamodelowane na diagramie. Niewątpliwie w trakcie dalszego rozwoju aplikacji diagram ulegnie znacznym zmianom.

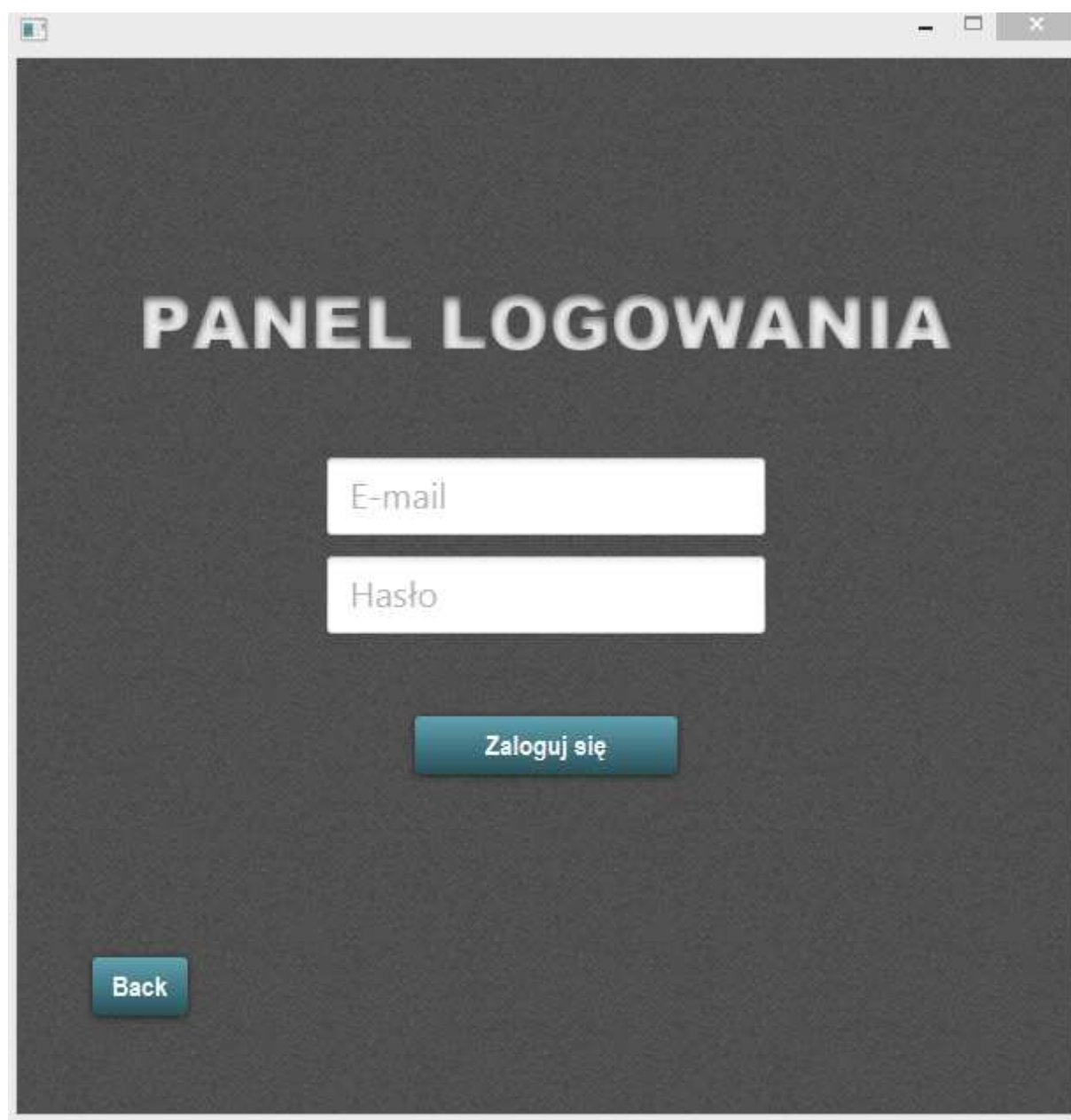
*Z racji słabej czytelności diagramu po załączeniu go do dokumentu, został on dołączony jako załącznik nr 1*

## 6. Wygląd Aplikacji

### 6.1 Menu główne



## 6.2 Panel Logowania



The image shows a screenshot of a web application window titled "PANEL LOGOWANIA". The window has a dark gray background. In the center, there are two white input fields stacked vertically. The top field is labeled "E-mail" and the bottom field is labeled "Hasło". Below these fields is a teal button with the text "Zaloguj się". In the bottom left corner, there is another teal button with the text "Back". The window has a standard OS title bar with minimize, maximize, and close buttons.

**PANEL LOGOWANIA**

E-mail

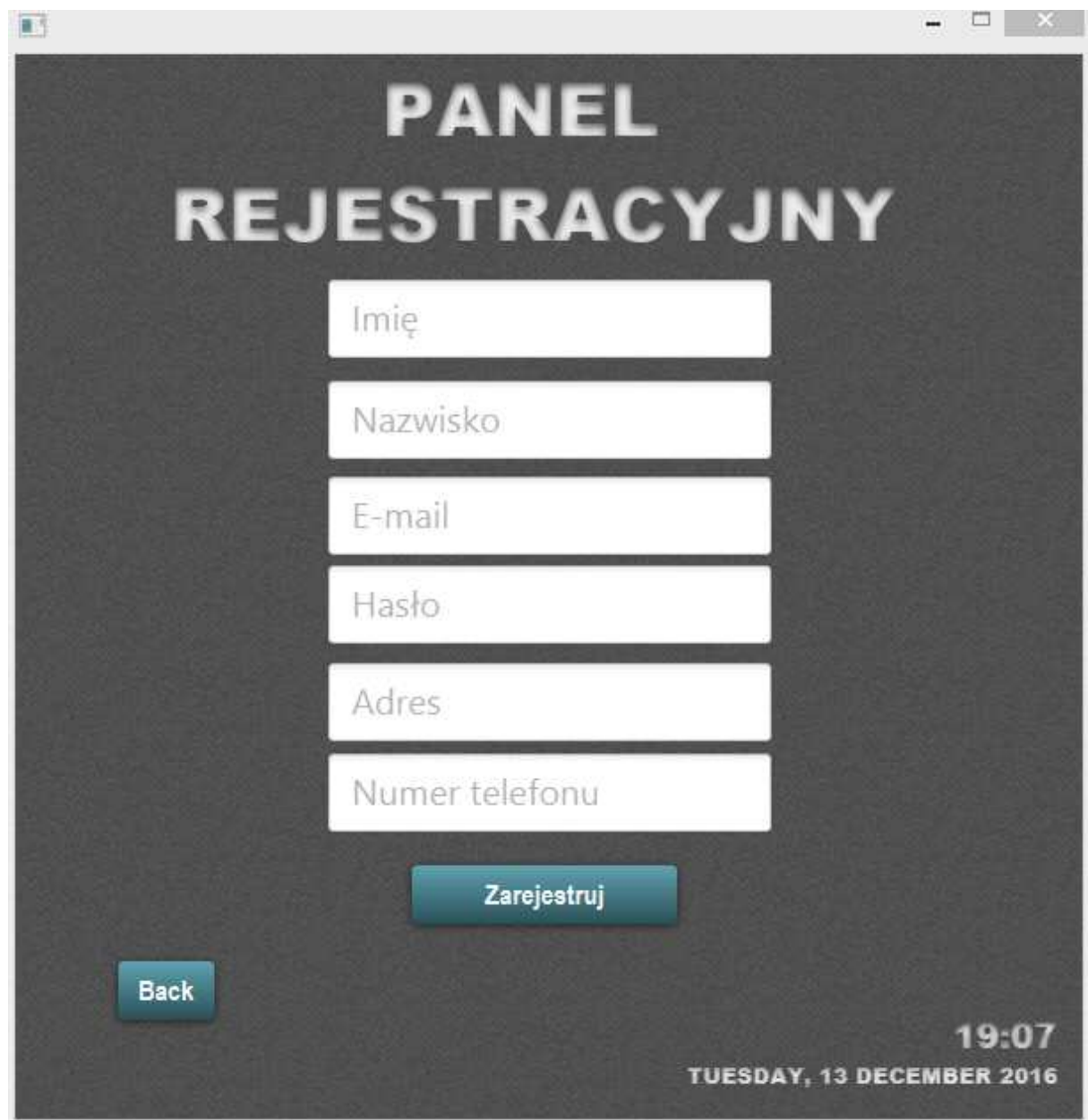
Hasło

Zaloguj się

Back



### 6.3 Panel Rejestracji



A screenshot of a web application window titled "PANEL REJESTRACYJNY". The window has a dark gray background and a light gray title bar with standard Windows window controls. The registration form consists of six white input fields stacked vertically, each with a gray label: "Imię", "Nazwisko", "E-mail", "Hasło", "Adres", and "Numer telefonu". Below the fields is a teal "Zarejestruj" button. In the bottom left corner is a teal "Back" button. In the bottom right corner, the time "19:07" and date "TUESDAY, 13 DECEMBER 2016" are displayed in white.

**PANEL  
REJESTRACYJNY**

Imię

Nazwisko

E-mail

Hasło

Adres

Numer telefonu

**Zarejestruj**

**Back**

**19:07**  
**TUESDAY, 13 DECEMBER 2016**

## 6.4 Panel Użytkownika



## 6.5 Panel Profil Użytkownika



The screenshot shows a web application window with a yellow title bar. The main content area has a dark gray background with the title "TWÓJ PROFIL" in large, white, bold letters. Below the title, there are five input fields for user information, each with a label to its left: "Imię" (Andrzej), "Nazwisko" (Nowak), "Adres" (Kwiatowa 28 Warszawa), "Numer telefonu" (300-500-400), and "Adres e-mail" (anowak@gmail.com). Below these fields are two buttons: "Edytuj" and "Zatwierdz". In the bottom left corner, there is a "Powrót" button. In the bottom right corner, the time "18:49" and the date "TUESDAY, 13 DECEMBER 2016" are displayed.

Imię	Andrzej
Nazwisko	Nowak
Adres	Kwiatowa 28 Warszawa
Numer telefonu	300-500-400
Adres e-mail	anowak@gmail.com

[Edytuj](#) [Zatwierdz](#)

[Powrót](#)

18:49  
TUESDAY, 13 DECEMBER 2016