**Interdyscyplinarne Centrum Modelowania Komputerowego  
Wydział Matematyczno – Przyrodniczy  
Uniwersytet Rzeszowski**

**Przedmiot:**

**Programowanie zespołowe**

**Dokumentacja projektu:**

***System do zarządzania zadaniami  
uprawy pomidorów***

**Wykonał:**

**Zespół projektowy L4GR2**

**Prowadzący: mgr inż. Adam Szczur**

**Rzeszów 2019**

# Zespół projektowy

Nazwa zespołu:

Skład zespołu projektowego:

|  |  |  |
| --- | --- | --- |
| Imię i nazwisko | Pełniona rola | Zakres czynności / odpowiedzialności |
| Marcin Rosół | Lider | Rozdzielanie zadań/ programowanie |
| Rafał Pawłowski | Członek zespołu | Projektowanie bazy danych/prowadzenie dokumentacji/tester manualny |
| Marcin Tomczyk | Członek zespołu | Projektowanie bazy danych/tester manualny |
| Paweł Warcaba | Zastępca Lidera/Członek zespołu | Programowanie |
| Grzegorz Żak | Członek zespołu | Programowanie |

# Specyfikacja projektu

## Opis programu / systemu

### Stworzenie programu ułatwiającego do zarządzania zadaniami podczas hodowli pomidorów

### Projekt zakłada aplikację, która będzie służyła jako manager zadań do plantacji pomidorów.

* Aplikacja zakłada trzy grupy użytkowników, Administrator, manager, pracownik
* Rolą administratora będzie dodawanie nowych zadań które zawierają tytuł zadania i opis zadania. Zadania stworzone przez administratora będą dostępne w panelu Managera. Administrator będzie posiadał możliwość generowania raportów. Manager i administrator będzie rozdawał zadania dla pracowników, otrzyma również dostęp do wglądu wykonanych oraz zaplanowanych zadań.
* Pracownik otrzymuje wgląd do zadanych mu zadań oraz wgląd do wykonanych przez siebie zadań.

## Wymagania stawiane aplikacji / systemowi

* System powinien mieć kilka modułów
  + Moduł zarządzania użytkownikami (logowanie, rejestracja, zmiana hasła)
  + Moduł zarządzania zadaniami (definiowanie, edycja, przypisanie, podgląd dostępnych)
  + Moduł generowania raportów (raport z zakresu dat, raport miesięczny użytkownika, raport całkowity użytkownika)
  + Moduł wykonywania zadań (podgląd zadań do wykonania, podgląd wykonanych zadań, oznaczanie zadań jako wykonane)
* System powinien umożliwiać generowanie raportów PDF.
* System powinien współpracować z bazą danych.

## Panele / zakładki systemu, które będą oferowały potrzebne funkcjonalności

* Panel administratora
  + Główne narzędzie administratora systemu umożliwiające wykonanie wszystkich czynności potrzebnych do zarządzania systemem np. tworzenie, przypisanie, modyfikacja zadań, podgląd zadań dostępnych oraz przypisanych, generowanie raportów.
* Panel managera
  + Generowanie raportów, przypisanie zadań dla pracowników, podgląd zadań wykonanych, zmiana hasła.
* Panel pracownika
  + Wgląd na otrzymane zadania, zatwierdzanie wykonania zadania, wgląd na wykonane zadania, zmiana hasła.
  1. Typy wymaganych dokumentów w projekcie oraz dostęp do nich

### Raporty PDF

#### Rodzaje raportów PDF

### Raporty dotyczący wszystkich zadań w podanym zakresie dat.

### Raporty dotyczący wszystkich zadań dla konkretnego pracownika.

### Raporty dotyczący zadań jakie zostały ukończone w danym miesiącu przez podanego pracownika.

## Przepływ informacji w środowisku systemu

Oparty na bazie danych

## Użytkownicy aplikacji i ich uprawnienia

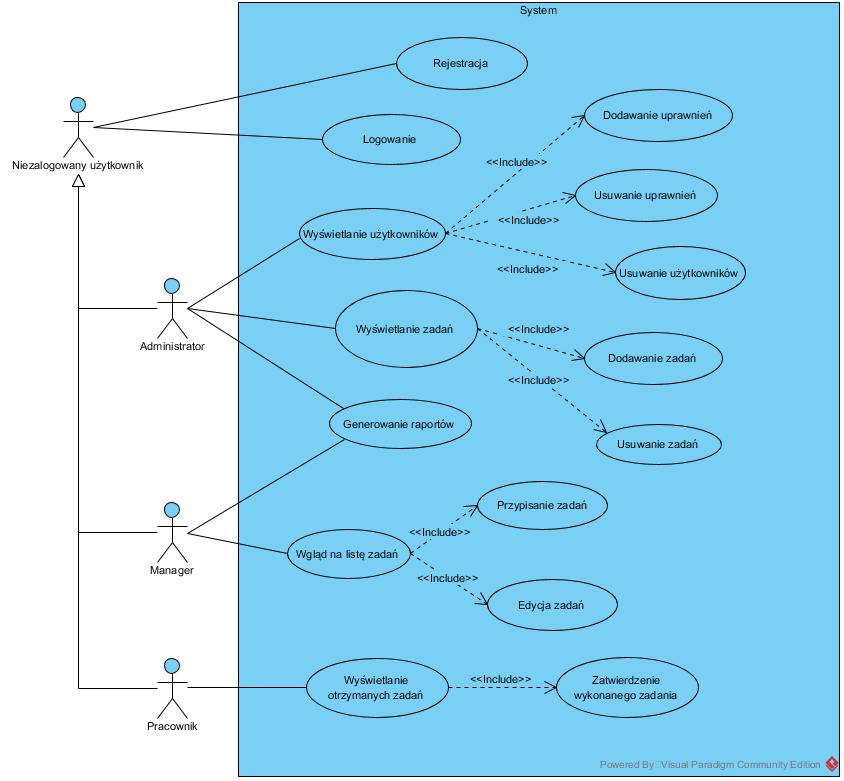
* Administrator
  + Dodawanie zadań
  + Usuwanie zadań
  + Edycja zadań
  + Generowanie raportów
  + Przypisanie zadań dla pracowników
  + Podgląd wykonanych zadań
  + Podgląd dostępnych zadań
* Manager
  + Generowanie raportów
  + Przypisanie zadań dla pracowników
  + Podgląd dostępnych zadań
  + Podgląd wykonanych zadań
  + Edycja zadań
* Pracownik
  + Wgląd na listę otrzymanych zadań
  + Wgląd na listę wykonanych zadań
  + Zatwierdzenie wykonanego zadania

## Interesariusze

* Administrator (zdefiniowany na stałe admin:admin)
* Manager (zdefiniowany na stałe manager:manager)
* Pracownik (każdy zarejestrowany przez interfejs użytkownik automatycznie otrzymuje prawa pracownika)

# Diagramy UML

## Diagram przypadków użycia

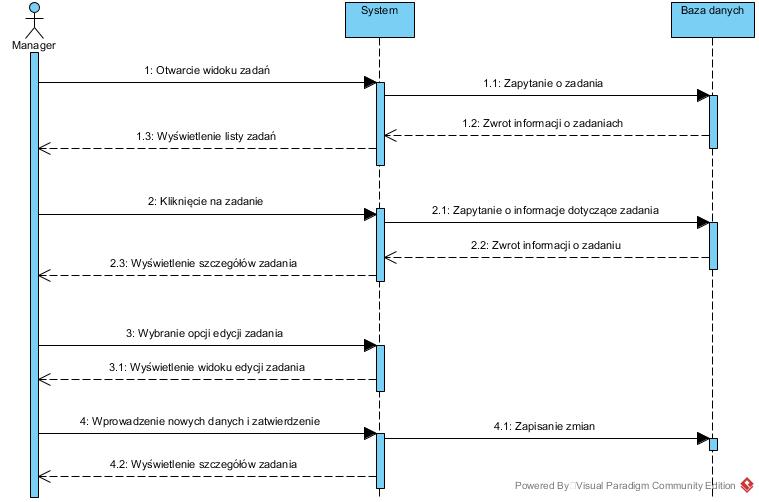


## Diagram aktywności



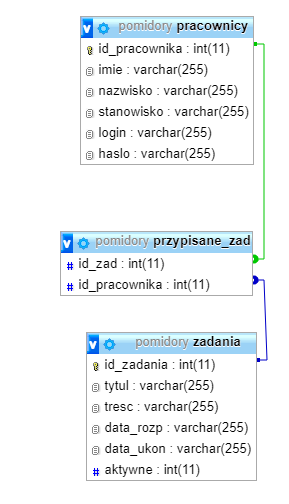
## Diagram sekwencji





# Baza danych

## Diagram ERD



## Skrypt do utworzenia struktury bazy danych

-- phpMyAdmin SQL Dump

-- version 4.6.6deb4

-- https://www.phpmyadmin.net/

--

-- Host: localhost:3306

-- Czas generowania: 01 Kwi 2019, 18:03

-- Wersja serwera: 10.1.37-MariaDB-0+deb9u1

-- Wersja PHP: 7.0.30-0+deb9u1

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

SET time\_zone = "+00:00";

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8mb4 \*/;

--

-- Baza danych: `pomidory`

--

-- --------------------------------------------------------

--

-- Struktura tabeli dla tabeli `pracownicy`

--

CREATE TABLE `pracownicy` (

`id\_pracownika` int(11) NOT NULL,

`imie` varchar(255) NOT NULL,

`nazwisko` varchar(255) NOT NULL,

`stanowisko` varchar(255) NOT NULL,

`login` varchar(255) NOT NULL,

`haslo` varchar(255) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- --------------------------------------------------------

--

-- Struktura tabeli dla tabeli `przypisane\_zad`

--

CREATE TABLE `przypisane\_zad` (

`id\_zad` int(11) NOT NULL,

`id\_pracownika` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- --------------------------------------------------------

--

-- Struktura tabeli dla tabeli `zadania`

--

CREATE TABLE `zadania` (

`id\_zadania` int(11) NOT NULL,

`tytul` varchar(255) NOT NULL,

`tresc` varchar(255) NOT NULL,

`data\_rozp` varchar(255) NOT NULL,

`data\_ukon` varchar(255) NOT NULL,

`aktywne` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--

-- Indeksy dla zrzutów tabel

--

--

-- Indexes for table `pracownicy`

--

ALTER TABLE `pracownicy`

ADD PRIMARY KEY (`id\_pracownika`);

--

-- Indexes for table `przypisane\_zad`

--

ALTER TABLE `przypisane\_zad`

ADD KEY `id\_zad` (`id\_zad`),

ADD KEY `id\_pracownika` (`id\_pracownika`);

--

-- Indexes for table `zadania`

--

ALTER TABLE `zadania`

ADD PRIMARY KEY (`id\_zadania`);

--

-- AUTO\_INCREMENT for dumped tables

--

--

-- AUTO\_INCREMENT dla tabeli `pracownicy`

--

ALTER TABLE `pracownicy`

MODIFY `id\_pracownika` int(11) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT dla tabeli `zadania`

--

ALTER TABLE `zadania`

MODIFY `id\_zadania` int(11) NOT NULL AUTO\_INCREMENT;

--

-- Ograniczenia dla zrzutów tabel

--

--

-- Ograniczenia dla tabeli `przypisane\_zad`

--

ALTER TABLE `przypisane\_zad`

ADD CONSTRAINT `przypisane\_zad\_ibfk\_1` FOREIGN KEY (`id\_zad`) REFERENCES `pracownicy` (`id\_pracownika`) ON DELETE NO ACTION ON UPDATE NO ACTION,

ADD CONSTRAINT `przypisane\_zad\_ibfk\_2` FOREIGN KEY (`id\_pracownika`) REFERENCES `zadania` (`id\_zadania`);

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

# Wykorzystane technologie

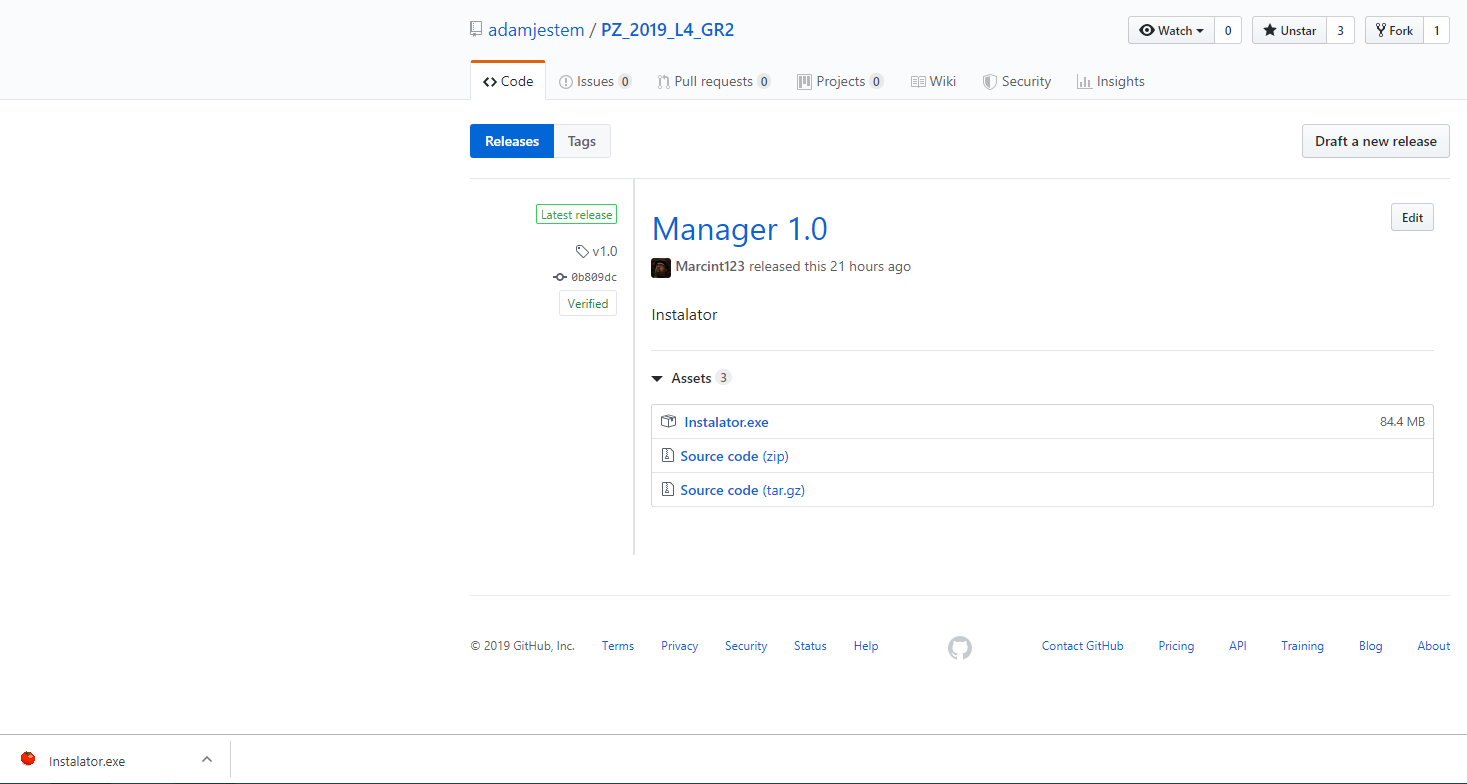
* Język Java 8 - Java jest technologią wykorzystywaną do tworzenia aplikacji, które czynią Internet bardziej atrakcyjnym i użytecznym. Java to nie to samo co JavaScript, będący prostą technologią stosowaną do tworzenia stron internetowych i działającą tylko w przeglądarce.
* Baza danych MySQL - wolnodostępny system zarządzania relacyjnymi bazami danych dostarczany przez firmę Oracle
* JavaFX - pozwala tworzyć aplikacje, które wyglądają bardziej nowocześnie, są równie wydajne, a przy tym można zachować dużo większą czytelność kodu.
* SceneBuilder 2.0 – narzędzie pracujące z javaFX, umożliwia szybkie iteracyjne tworzenie interfejsu użytkownika metodą przeciąg-upuść.
* Maven - narzędzie automatyzujące budowę oprogramowania na platformę Java. Poszczególne funkcje Mavena realizowane są poprzez wtyczki, które są automatycznie pobierane przy ich pierwszym wykorzystaniu. Plik określający sposób budowy aplikacji nosi nazwę POM-u (ang. Project Object Model).

# Instalator aplikacji

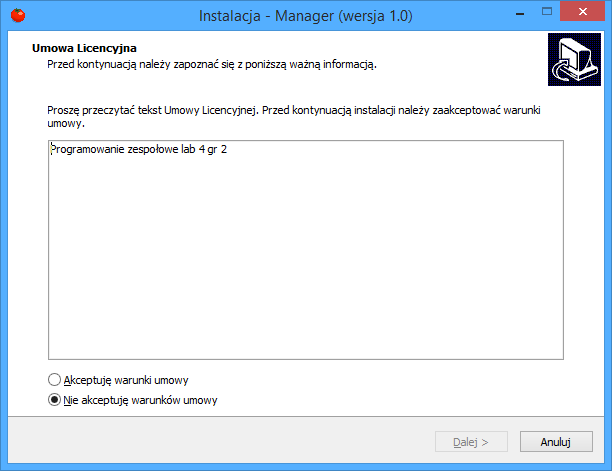
## Kluczowe informacje

* Instalator został wykonany w jrSoftware – InnoSetup.
* Program managera domyślnie zostaje zainstalowany w katalogu C:\Program Files (x86)\Manager
* Instalator instaluje silnik bazy danych MySQL w lokalizacji C:\Users\<NAZWA\_UŻYTKOWNIKA>\AppData\Roaming\Pomidory
* Podczas instalacji można opcjonalnie dodać skrót na pulpicie oraz dodać podmenu w Menu Start.
* Po zainstalowaniu programu, możliwa jest jego deinstalacja po przez załączony w lokalizacji C:\Program Files (x86)\Manager\uins000.exe

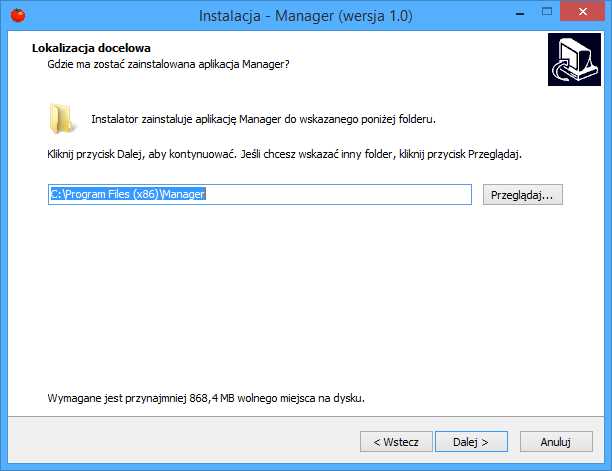
## Interfejs instalatora – przebieg instalacji



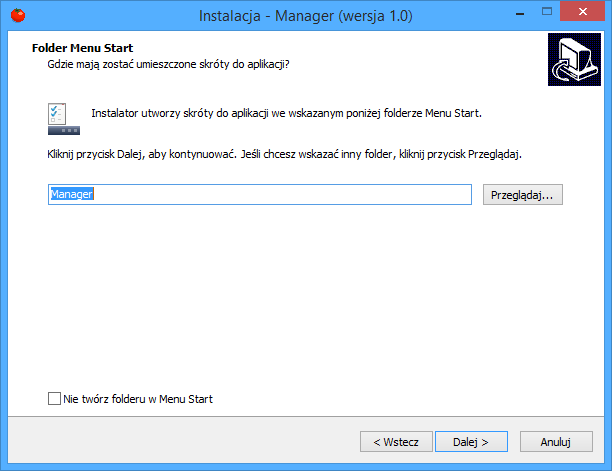
*Instalator można pobrać z Releases projektowego GitHuba, opatrzony jest on ikoną pomidora (Instalator.exe)*



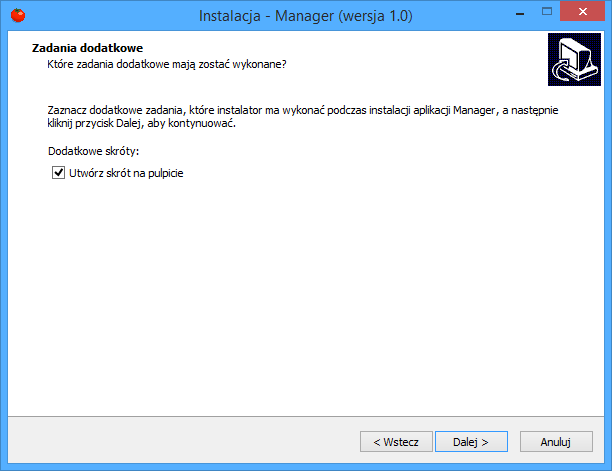
*Pierwsze okno dostępne po uruchomieniu instalatora to okno Umowa Licencyjna. Obecnie umowa licencyjna jest zbędna ponieważ program nie jest nigdzie dystrybuowany.*



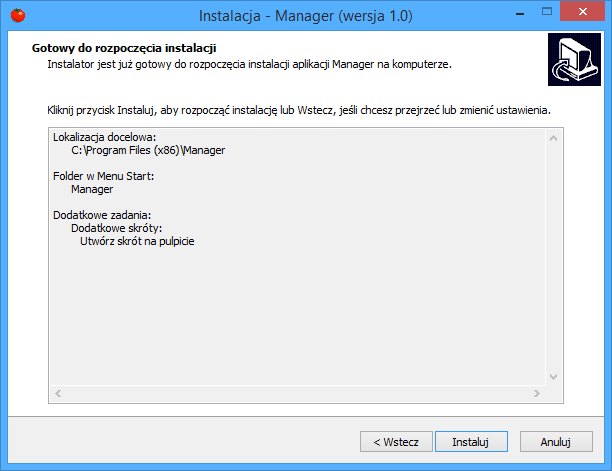
*Okno instalatora pozwalające na wybór lokalizacji instalacji, domyślnie instalator instaluje pliki w lokalizacji C:\Program Files (x86)\Manager*



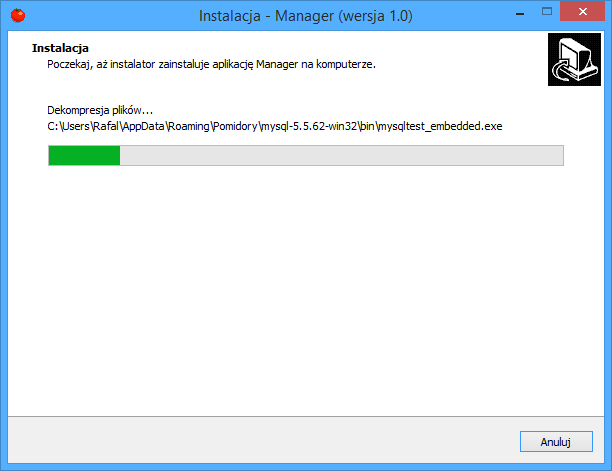
*Okno instalatora definiujące opcjonalne skróty do programu w Menu Start, można zmienić nazwę podkategorii w Menu Start oraz zrezygnować z tworzenia wpisu w Menu Start.*



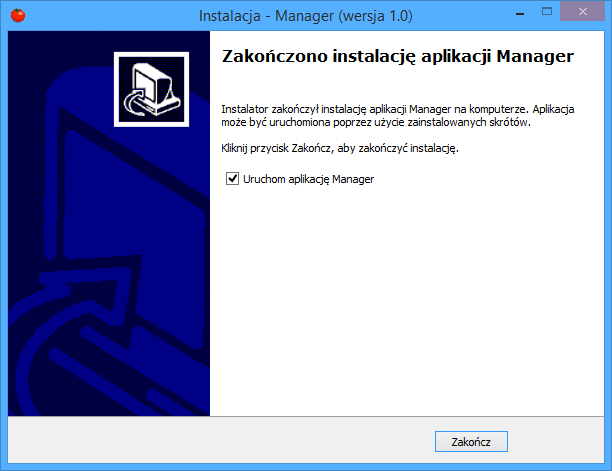
*Okno utworzenia skrótu na pulpicie.*



*Okno podsumowania działania instalatora.*



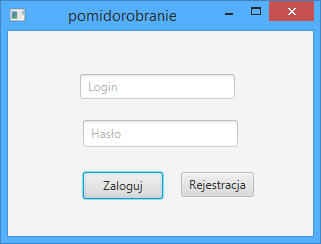
*Proces Instalacji, jak widać Silnik bazy danych MySQL jest instalowany w lokalizacji C:\Users\<NAZWA\_UŻYTKOWNIKA>\AppData\Roaming\Pomidory*



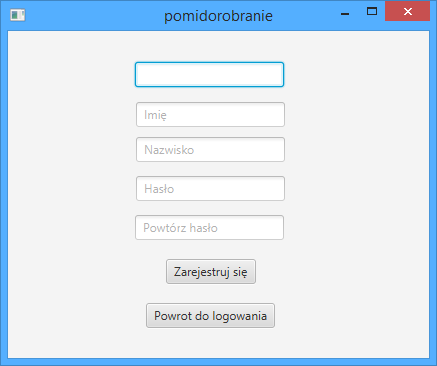
*Zakończenie instalacji programu Manager.*

# Interfejs aplikacji / systemu

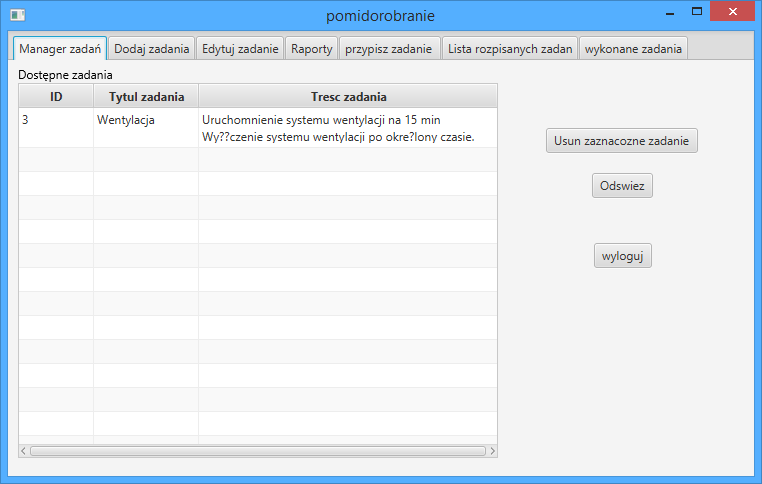
## Okno główne, logowania



## Okno rejestracji

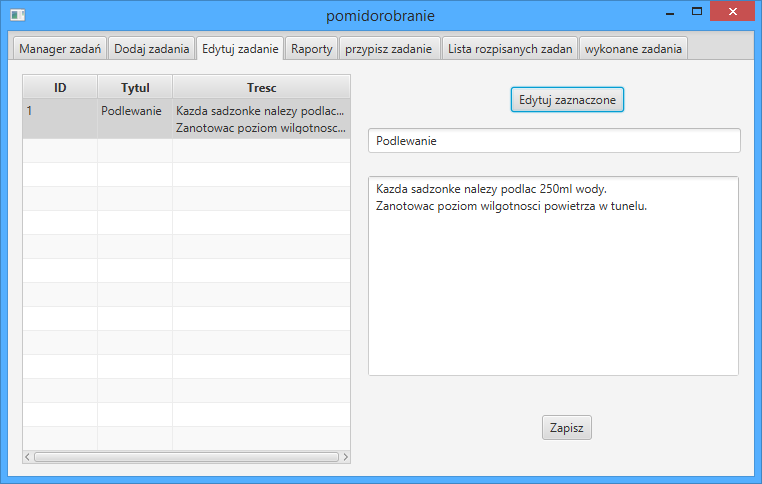


## Panel administratora

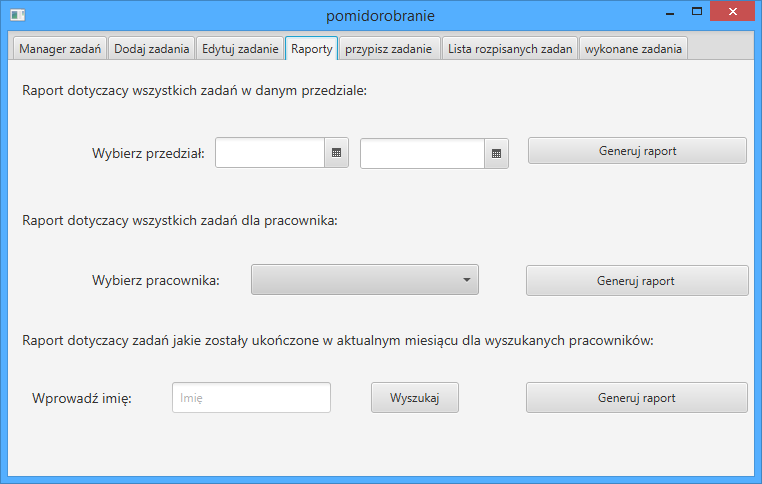


*Zakładka „Manager Zadań” udostępnia podgląd na aktualnie dostępne zadania, które nie są przypisane żadnemu z pracowników. Jeżeli nowe zadanie dostępne zostanie przypisane pracownikowi, znika ono z tego widoku. Z poziomu tej zakładki będąc Administratorem można usunąć zadania. Dostępny jest też przycisk wylogowania.*

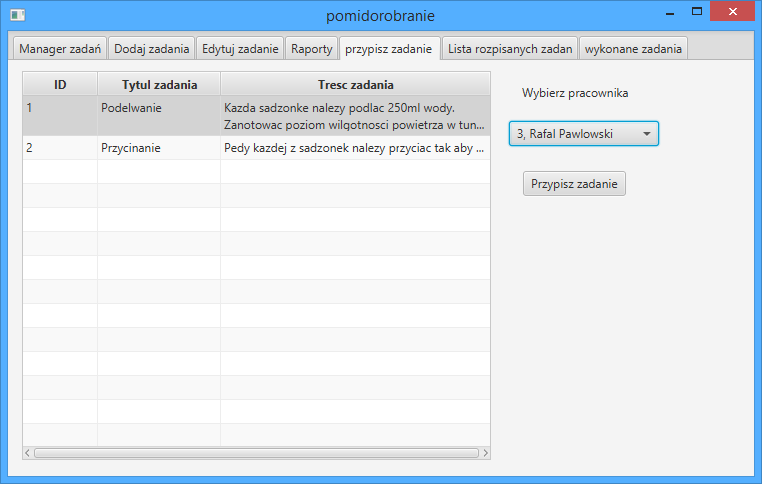
*Zakładka „Dodaj zadania” umożliwia zdefiniowanie nowego zadania. Podajemy tytuł zadania oraz jego treść i przyciskiem „Dodaj zadanie” zapisujemy zadanie w bazie.*



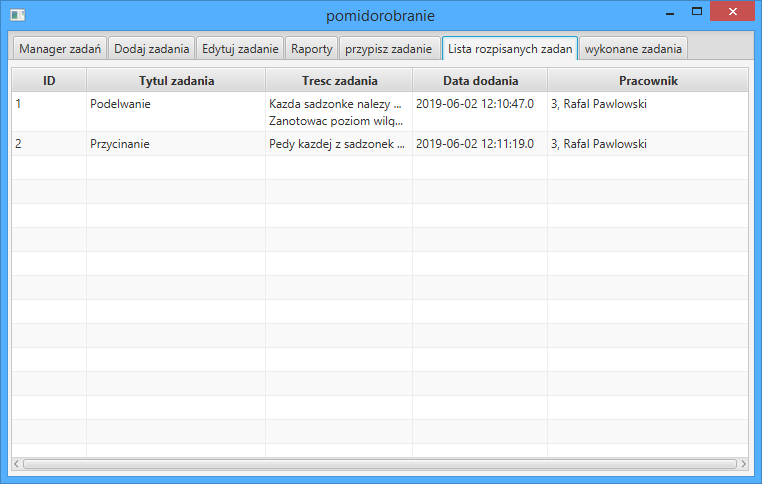
*Zakładka „Edytuj zadanie” pozwala na wybór zadania i jego edycję. Aby to zrobić zaznaczamy zadanie do edycji w tabeli po lewej, klikamy „Edytuj zaznaczone”. Treść zadania oraz tytuł wczyta się do pól tekstowych. Po edycji zapisujemy przyciskiem „Zapisz”*



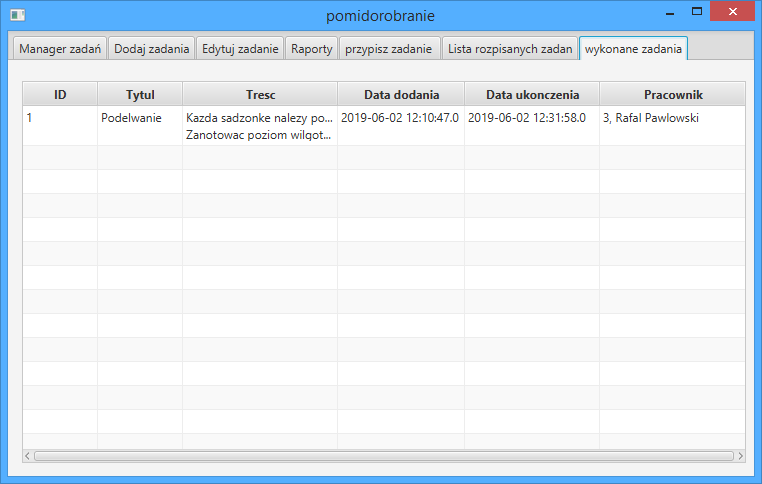
Zakładka „Raporty” umożliwia generowanie raportów. Możemy wygenerować raport całościowy z przedziału wybranych dat, Raport dotyczący wszystkich zadań konkretnego pracownika, oraz Raport zadań danego pracownika w bieżącym miesiącu.



*Zakładka „Przypisz Zadanie” pozwala na przypisane do wybranego z combo-listy zadań dostępnych – nie przypisanych nikomu. Zaznaczamy wybrane zadanie, wybieramy pracownika, i klikamy „Przypisz zadanie”*

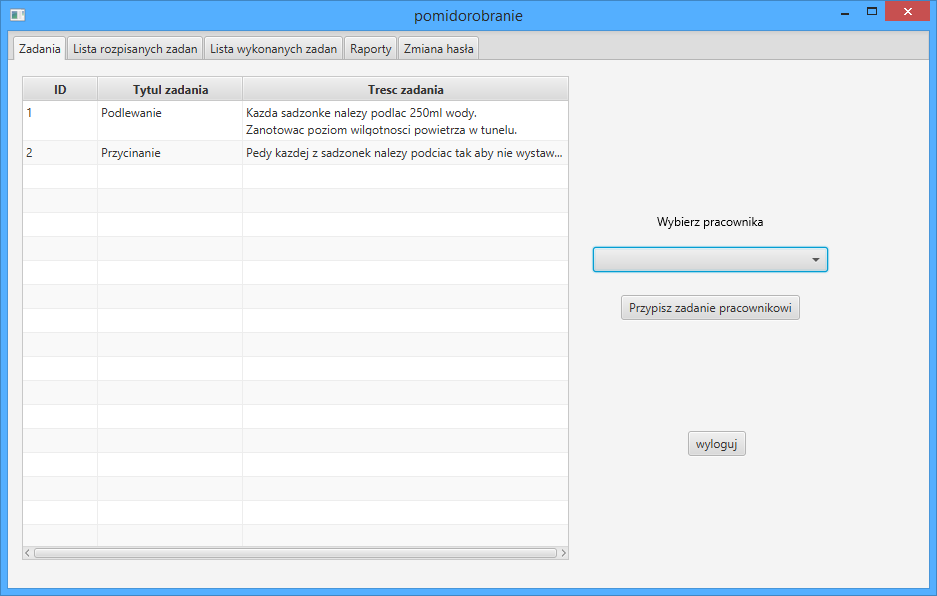


*Zakładka „Lista rozpisanych zadań” pozwala na podgląd wszystkich przydzielonych dla pracowników zadań, które nie zostały jeszcze wykonane.*

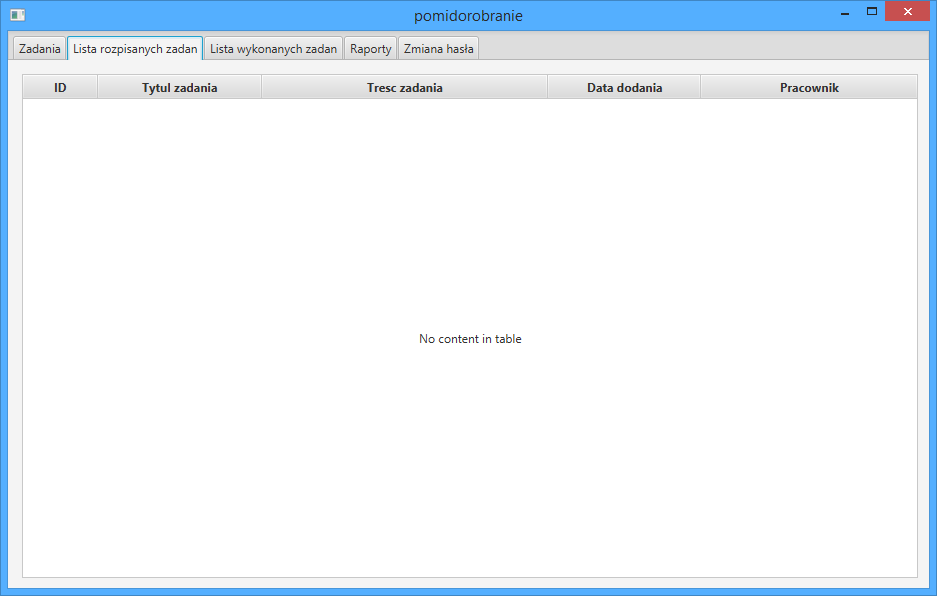


*Zakładka „wykonane zadania” prezentuje listę zadań oznaczonych przez pracowników jako wykonane.*

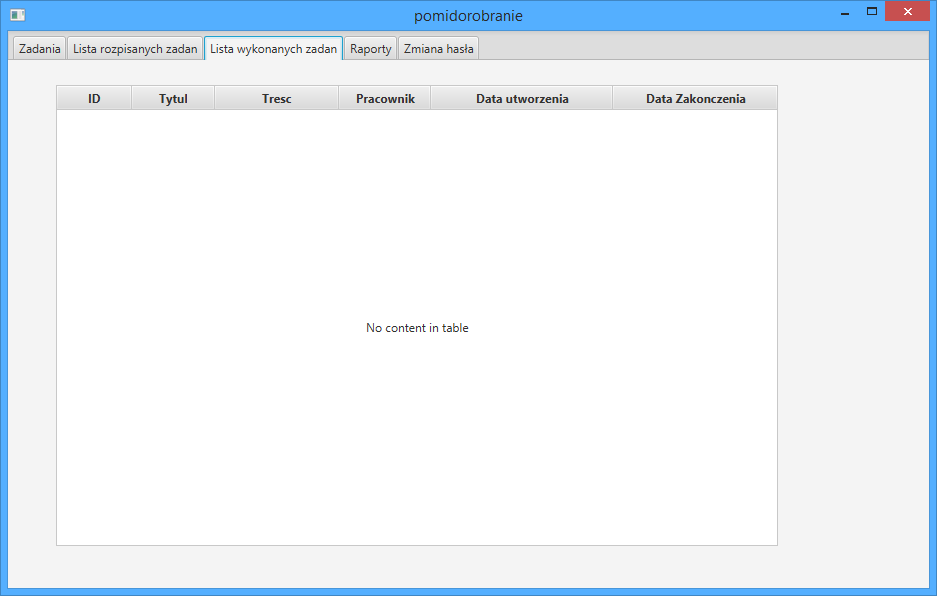
## Panel Managera



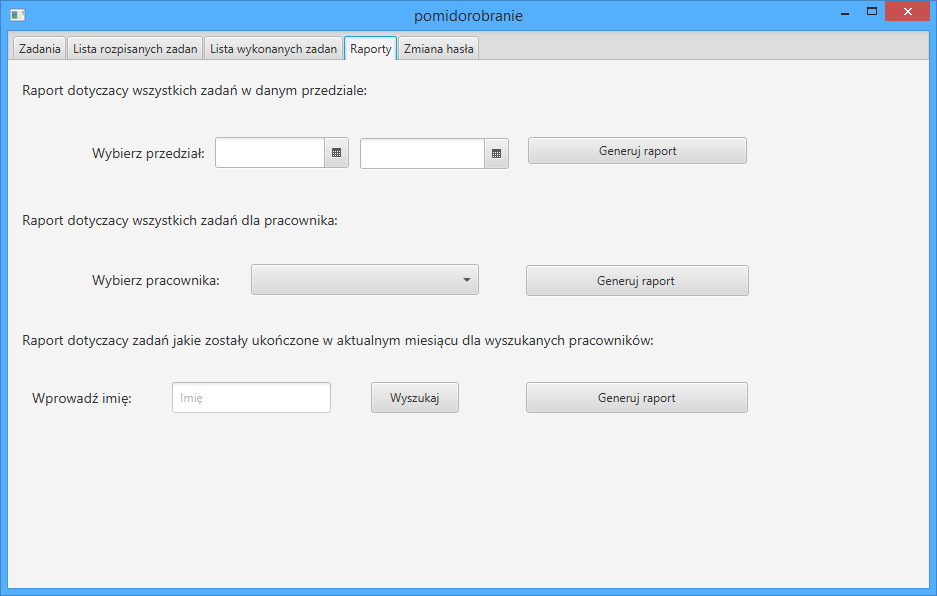
*Zakładka „zadania” dostępna z panelu managera pozwala na podgląd nieprzypisanych zadań oraz na przypisanie zadania do wybranego użytkownika. Aby to zrobić zaznaczamy zadanie w tabeli, wybieramy żądanego użytkownika z combo-listy i klikamy „Przypisz zdanie pracownikowi”.*



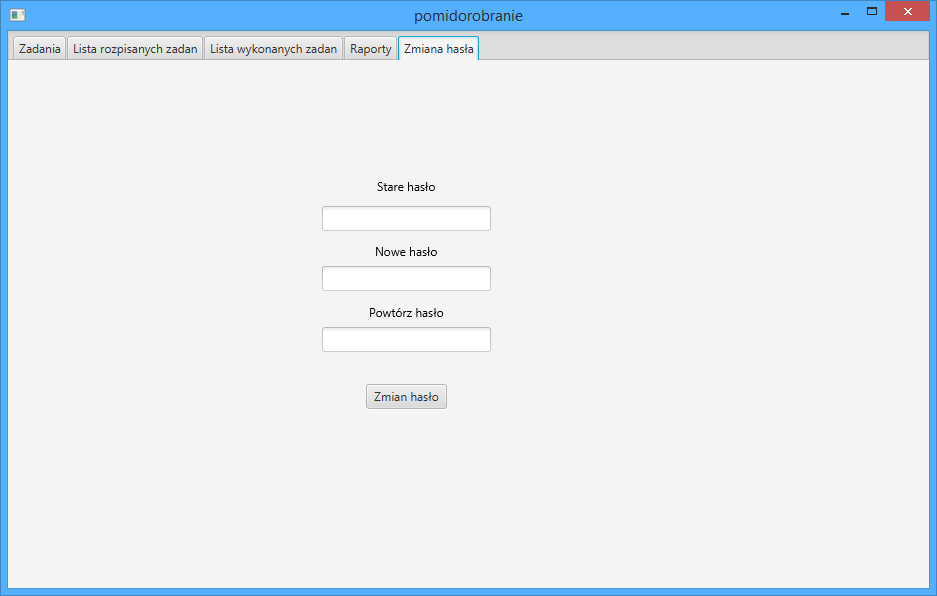
*Zakładka „lista rozpisanych zadań” dostępna dla managera udostępnia podgląd na zadania przypisane do konkretnych użytkowników.*



*Zakładka „lista wykonanych zadań” dostępna dla managera udostępnia podgląd na zadania które zostały oznaczone jako wykonane.*

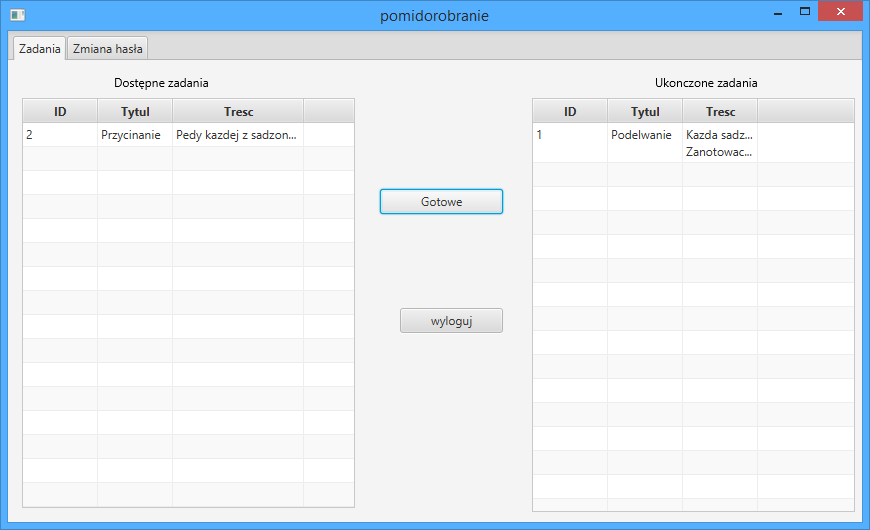


*Zakładka „Raporty” umożliwia generowanie raportów. Możemy wygenerować raport całościowy z przedziału wybranych dat, Raport dotyczący wszystkich zadań konkretnego pracownika, oraz Raport zadań danego pracownika w bieżącym miesiącu. Zakładka działa identycznie dla administratora jak i managera.*

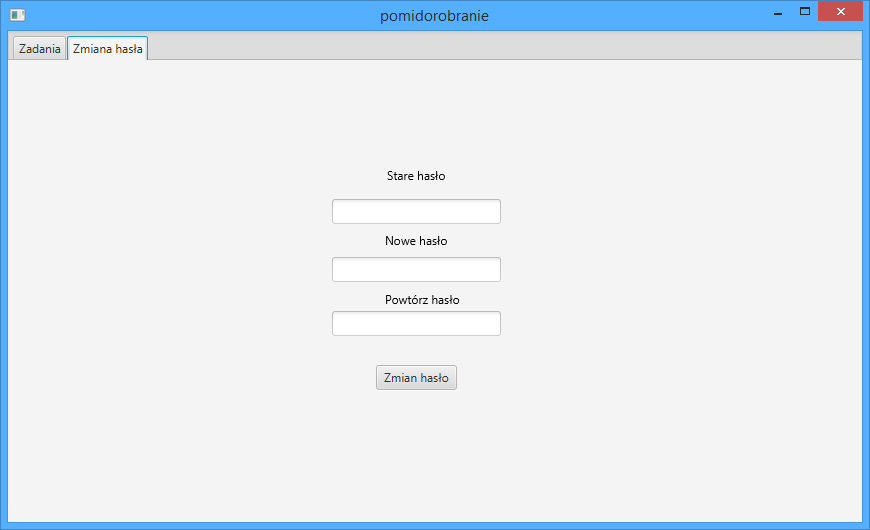


*Zakładka „Zmiana hasła” jest dostępna dla pracownika oraz managera, pozwala na zmianę hasła.*

## Panel użytkownika



*Zakładka „Zadania” jest dostępna tylko dla konta pracownika. Dostępne są tu zadania przypisane do aktualnie zalogowanego pracownika. Użytkownik, który wykona dane zadanie, zaznacza je w tabeli „Dostępne zadania” i klikając „gotowe” oznacza je jako zadanie wykonane. Wykonane zadania zalogowanego użytkownika są dostępne w tabeli „Ukończone zadania”*



Zakładka „Zmiana hasła” jest dostępna dla pracownika, pozwala na zmianę hasła.