



**Instytut Informatyki
Kolegium Nauk Przyrodniczych
Uniwersytet Rzeszowski**

Przedmiot:

Programowanie zespołowe

Dokumentacja projektu:

***System do zarządzania zadaniami
w firmie programistycznej***

Wykonał:

Zespół projektowy **L1GR1**

Prowadzący: mgr inż. Adam Szczur

Rzeszów 2020

1. Zespół projektowy

Nazwa zespołu: **L1GR1**

Skład zespołu projektowego:

Imię i nazwisko	Pełniona rola	Zakres czynności / odpowiedzialności
Dawid Żurek	Project Manager/Architekt Systemu	zorganizowanie i „spięcie” całego projektu. Wybór technologii i narzędzi dla projektu.
Dawid Łysik	Tester	Testy UAT
Arkadiusz Kornafel	Twórca Dokumentacji Użytkownika	Tworzenie manuali oraz dokumentacji technicznej
Krzysztof Witowski	Tester	Testy jednostkowe
Mariusz Olszyk	Programista	Backend
Piotr Bobusia	UI/UX Designer	Design Frontendu

2. Specyfikacja projektu

2.1. Opis programu / systemu

2.1.1. Cel projektu

System służyć będzie do zarządzania i monitorowania procesu produkcyjnego. Zawiera informacje o wykonywanym produkcie oraz jego statusie, który można generować do odpowiedniego PDF’a.

2.1.2. Zakres projektu

Produkty do systemu wprowadza koordynator, który ustala przez jakie etapy musi przejść dany detal, jego nazwę, datę realizacji, ilość oraz z jakiego jest zlecenia (kilka detali może mieć jedno zlecenie).

Po wprowadzeniu detalu gdy np. zaznaczyć krawalnię, ślusarnię i malarnię, detal początkowo pojawia się na krawalni, gdzie przez użytkownika tego działu może zostać pobrany (status zmienia się na: PRACA W TOKU / AKTUALNY), a następnie można go zakończyć (status: ZAKOŃCZONO) co oznacza, że detal może przejść do następnego działu (na ślusarnię).

System generowania PDF tworzy raport z nazwą wszystkich detali zlecenia, wraz z ich stanem produkcyjnym, ilością i datą realizacji.

2.2. Wymagania stawiane aplikacji / systemowi

- System powinien mieć kilka modułów
 - Moduł administracji użytkownikami (role)
 - Moduł raportów
 - Moduł konfiguracji
- System powinien umożliwiać generowanie raportów PDF
- System powinien współpracować z bazą danych

2.3. Panele / zakładki systemu, które będą oferowały potrzebne funkcjonalności

- Panel administratora

- Głównie narzędzie administratorów systemu umożliwiające wykonanie wszystkich czynności potrzebnych do zarządzania systemem np. dodawanie, edycja, usuwanie użytkowników, tworzenie i modyfikacja grup, zarządzanie innymi administratorami.
- Panel kierownika (koordynator produkcji)
 - zajmuje się wprowadzaniem detali do systemu
- Panele poszczególnych działów
 - różnice w panelach będą tylko w zakresie dostępu co do niektórych informacji.
- Zakładka raportów
 -
- Okno ustawień
 -

2.4. Typy wymaganych dokumentów w projekcie oraz dostęp do nich

2.4.1. Raporty PDF

2.4.1.1. Rodzaje raportów PDF

2.4.2. Inne dokumenty

2.5. Przepływ informacji w środowisku systemu

2.6. Użytkownicy aplikacji i ich uprawnienia

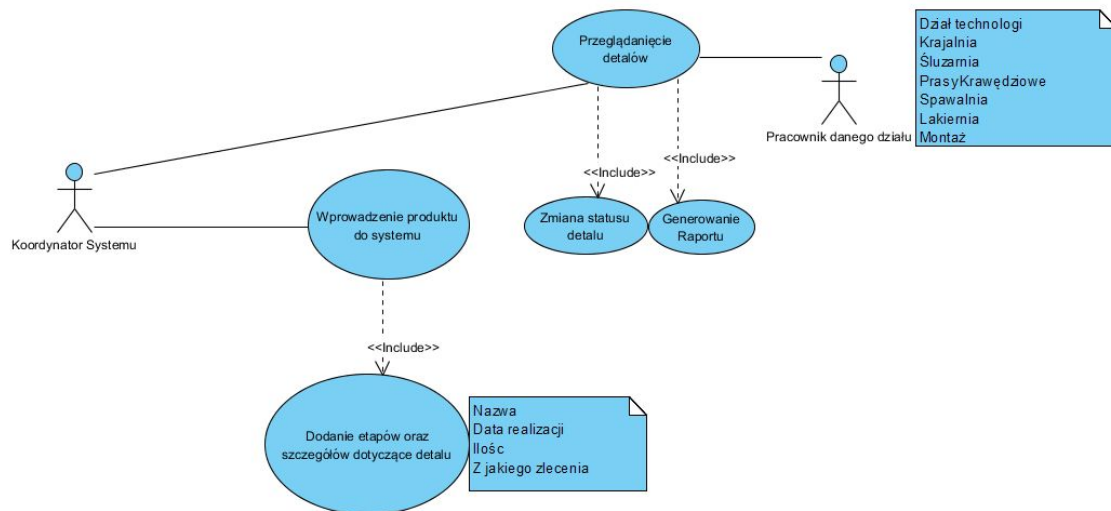
- Administrator
 - Wszystkie uprawnienia
- Kierownik (koordynator produkcji)
 - Uprawnienie umożliwiające wprowadzanie oraz edycje informacje w systemie
- Użytkownik (poszczególne działy produkcji)
 - Uprawnienie umożliwiające przeglądanie informacji w systemie

2.7. Interesariusze

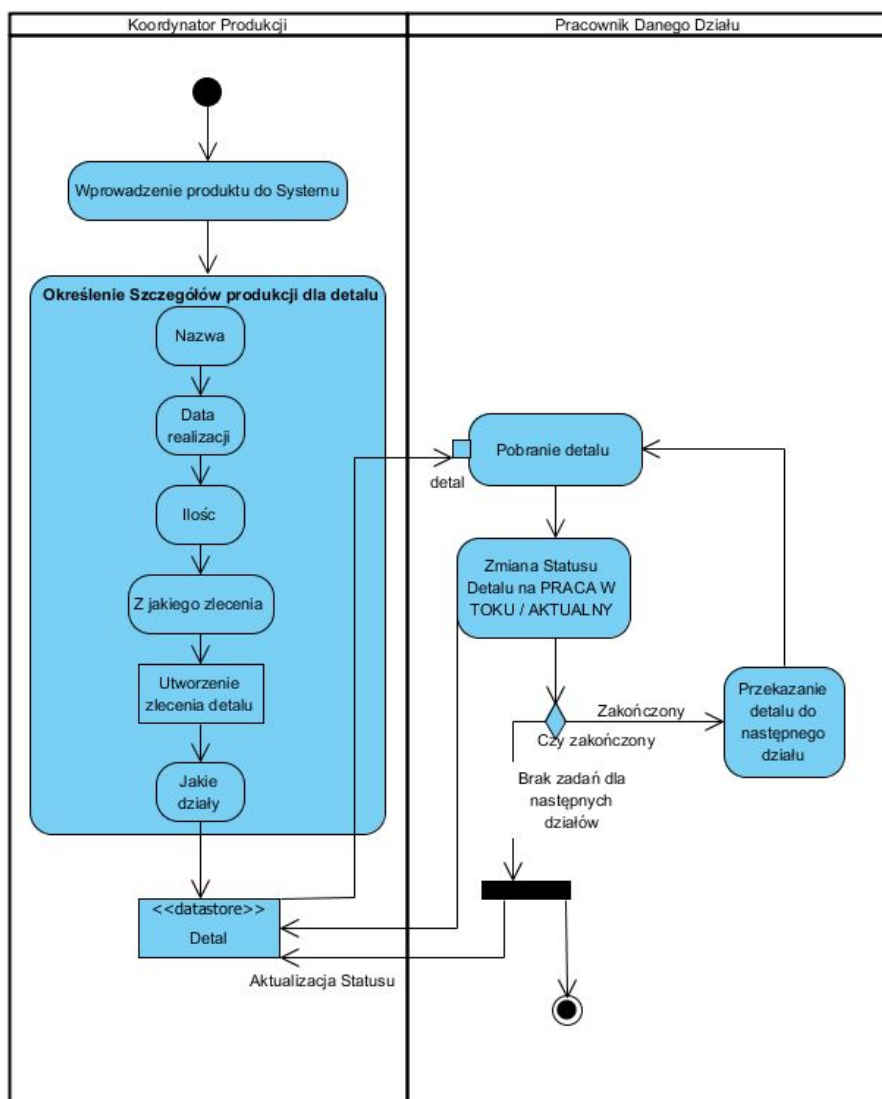
- Interesariusze wewnętrzni
 - Pracownicy (dział technologii, krawalnia, ślusarnia, prasy krawędziowe, spawalnia, lakiernia, malarnia)
 - Koordynator produkcji
- Interesariusze zewnętrzni
 - Brak

3. Diagramy UML

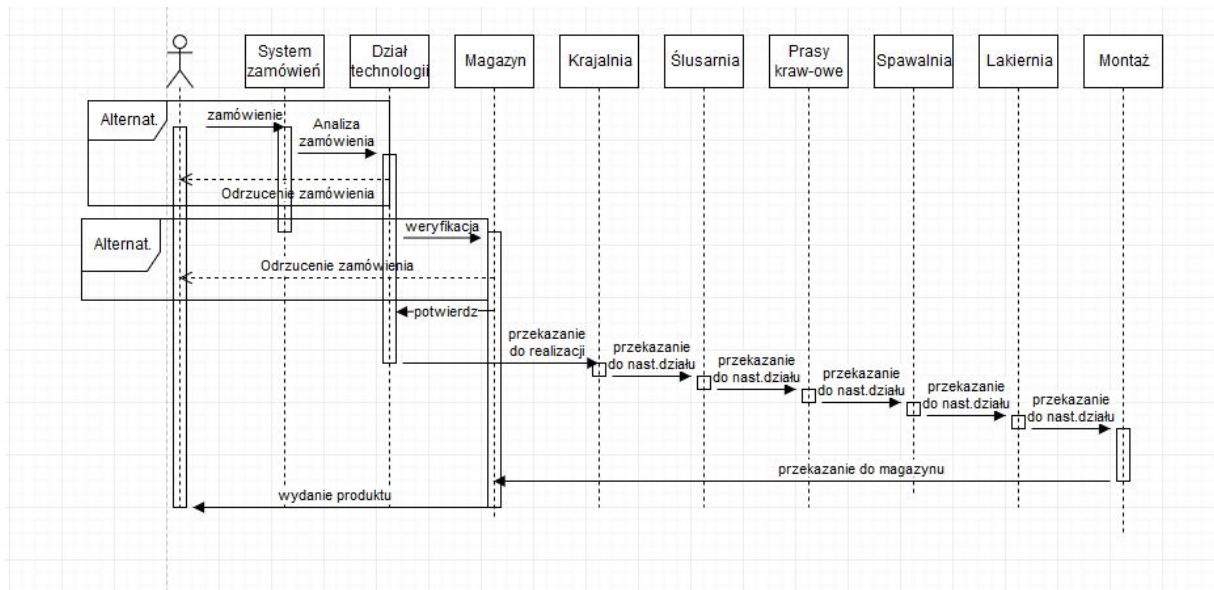
3.1. Diagram przypadków użycia



3.2. Diagram aktywności

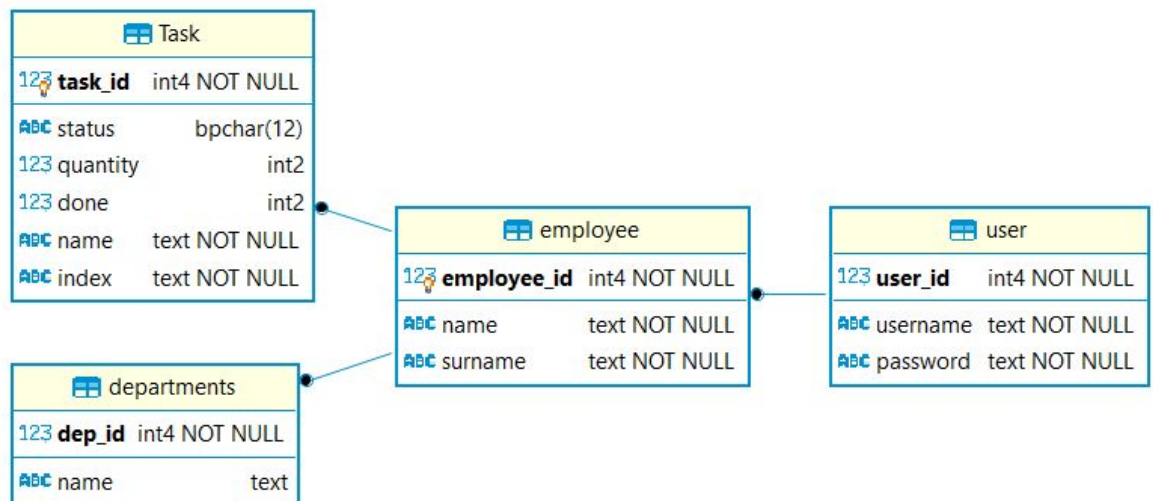


3.3. Diagram sekwencji



4. Baza danych

4.1. Diagram ERD



ERD utworzone w DBeaver

departments -> employee 1 do 1

employee -> user 1 do 1

Task -> employee wielu do wielu

4.2. Skrypt do utworzenia struktury bazy danych

```
SET statement_timeout = 0;
SET lock_timeout = 0;
```

```

SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;

--
-- Name: Firma; Type: DATABASE; Schema: -; Owner: postgres
--

CREATE DATABASE "Firma" WITH TEMPLATE = template0 ENCODING = 'UTF8' LC_COLLATE =
'Polish_Poland.1250' LC_CTYPE = 'Polish_Poland.1250';

ALTER DATABASE "Firma" OWNER TO postgres;

\connect "Firma"

SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;

SET default_tablespace = '';

SET default_table_access_method = heap;

--
-- Name: Task; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."Task" (
    status character(12),
    task_id integer NOT NULL,
    quantity smallint,
    done smallint,
    name text NOT NULL,
    index text NOT NULL
);

ALTER TABLE public."Task" OWNER TO postgres;

--
-- Name: departments; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public.departments (
    dep_id integer NOT NULL,
    name text
);

```



```

ALTER TABLE public.departments OWNER TO postgres;

--
-- Name: employee; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public.employee (
    name text NOT NULL,
    surname text NOT NULL,
    employee_id integer NOT NULL
);

ALTER TABLE public.employee OWNER TO postgres;

--
-- Name: user; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."user" (
    user_id integer NOT NULL,
    username text NOT NULL,
    password text NOT NULL
);

ALTER TABLE public."user" OWNER TO postgres;

--
-- Name: Task Task_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Task"
    ADD CONSTRAINT "Task_pkey" PRIMARY KEY (task_id);

--
-- Name: departments departments_id_key; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public.departments
    ADD CONSTRAINT departments_id_key UNIQUE (dep_id);

--
-- Name: employee employee_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public.employee
    ADD CONSTRAINT employee_pkey PRIMARY KEY (employee_id);

--
-- Name: user user_id_key; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."user"
    ADD CONSTRAINT user_id_key UNIQUE (user_id);

--
-- Name: Task Task -> emp; Type: FK CONSTRAINT; Schema: public; Owner: postgres

```

```
--
ALTER TABLE ONLY public."Task"
    ADD CONSTRAINT "Task -> emp" FOREIGN KEY (task_id) REFERENCES public.employee(employee_id)
NOT VALID;

--
-- Name: departments dep -> emp; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public.departments
    ADD CONSTRAINT "dep -> emp" FOREIGN KEY (dep_id) REFERENCES public.employee(employee_id) NOT
VALID;

--
-- Name: employee emp -> user; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public.employee
    ADD CONSTRAINT "emp -> user" FOREIGN KEY (employee_id) REFERENCES public."user"(user_id) NOT
VALID;
```

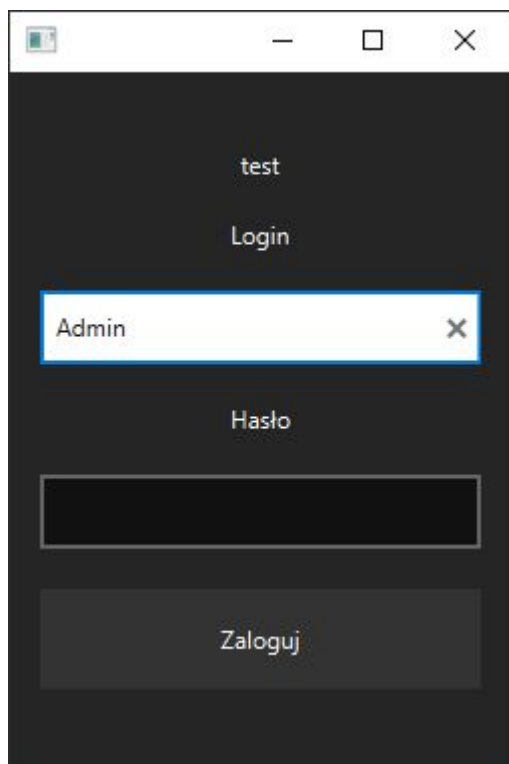
5. Wykorzystane technologie

- Język Java 11
 - Hibernate
 - iText
 - Jmetro (frontend CSS)
- Baza danych Postgres
- Maven
- JavaFX

6. Interfejs aplikacji / systemu

Jeżeli klient zmieni zdanie to dodanie większej ilości modułów nie będzie problemem

Okno Logowania



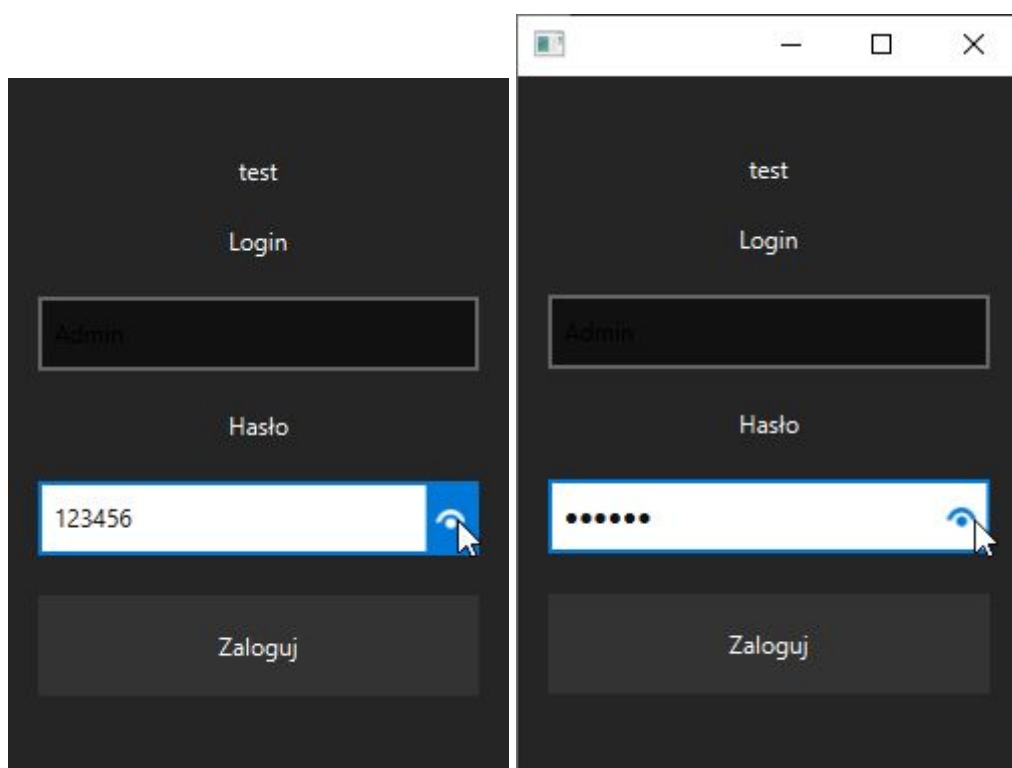
test

Login

Admin

Hasło

Zaloguj



test

Login

Admin

Hasło

123456

Zaloguj

test

Login

Admin

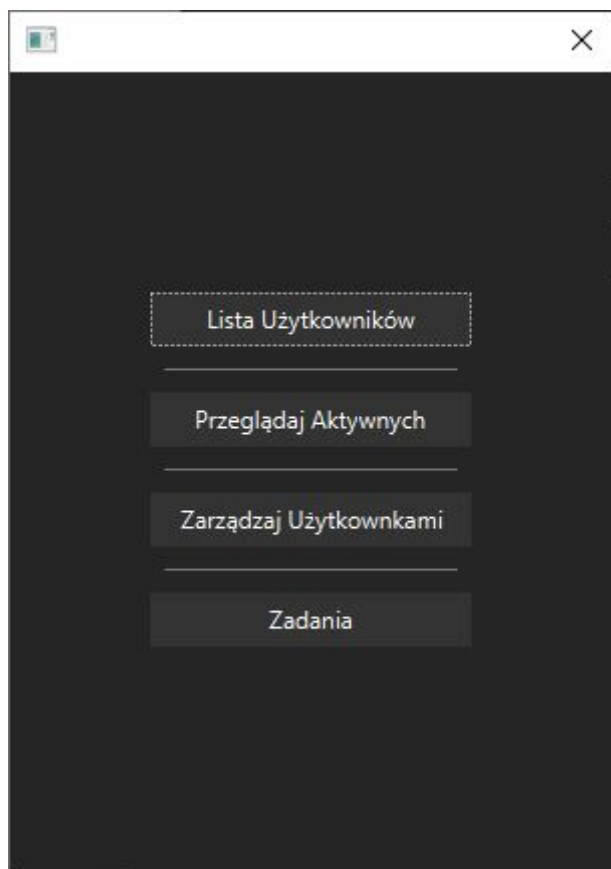
Hasło

.....

Zaloguj

Jesteśmy świadomi buga z polem tekstowym, zmieniającym kolor czcionki na czarny pracujemy nad rozwiązaniem tego

Menu Główne



Lista Użytkowników wyświetla listę użytkowników systemu (potrzebny poziom dostępu 2 lub większy).

Przeglądaj Aktywnych pozwala na wyświetlenie aktualnie zalogowanych użytkowników (ta funkcja pojawi jeśli, zespół programistyczny rozgryzie jak działa sesja zalogowania).

Zarządzaj Użytkownikami pozwala na edycję informacji użytkowników systemu (potrzebny poziom dostępu 3).

Zadania wyświetla okno, w którym jest możliwość przejrzenia, zarządzania oraz generowania raportów o aktualnie przydzielonych zadaniach.

Okno Zarządzaniem Użytkowników

Nazwa Użytkownika	Poziom Dostępu	Hasło	Dodaj
Marcin13	1		Usuń
Kowalski01	1		Edytuj
Adamczyk	1		Więcej Informacji
Admin	3		
Kierownik	2		

Pole hasło jest tylko na chwilę, testowane było kilka rzeczy.

Pole Dodaj dodaje nowego użytkownika do systemu.

Pole Usuń usuwa wybranego użytkownika ze systemu.

Pole Edytuj pozwala na edycje informacji wybranego użytkownika.

Pole Więcej Informacji wyświetla informacje dotyczące szczegółów związanych z pracą danego użytkownika.

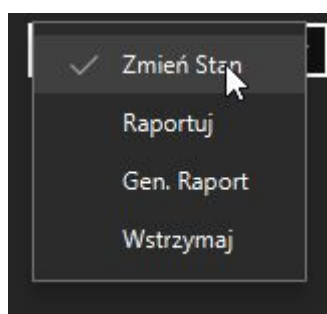
Okno Zadań

ID	Nazwa	Indeks	Ilość	Ile Zr.	Stan	Piorytet
1	Wyrób Metalowy	11.INVO.1120/15	3	1	W realizacji	Wysoki
2	Wyrób Drewniany	12.INVO.1195/16	20	20	Gotowy	Sredni
3	Wyrób Kompozyt...	13.INVO.1132/18	100	45	Przerwany	Niski

Zmień Stan

Powyższy layout przejdzie jeszcze kilka iteracji ponieważ wymagania klienta ciągle się zmieniają

W TableView wyświetlane są wszystkie informacje o danym zleceniu



ActionBox na dole umożliwia wykonywanie operacji na danym zadaniu.