***#30 Marzenie laboranta***

*Żaneta Mielczarek*

*Tomasz Kukuczka*

*Szymon Bartyzel*

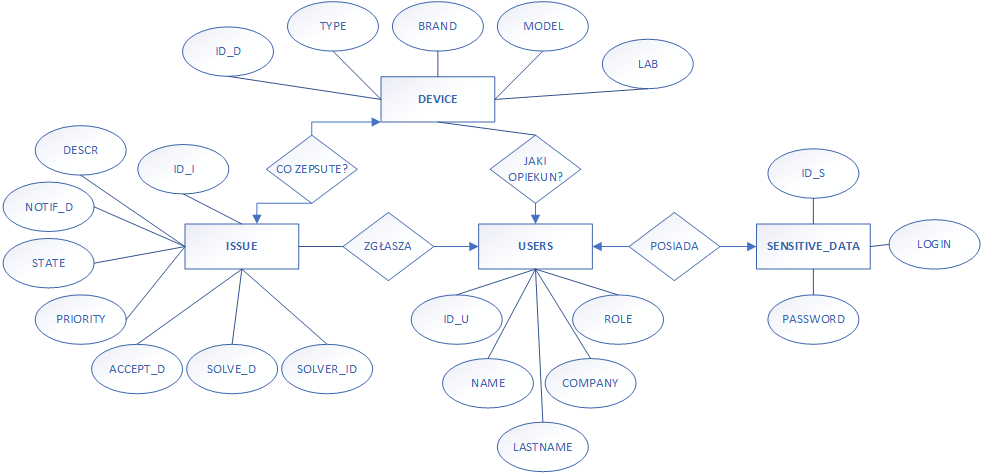
*Łukasz Michalik*

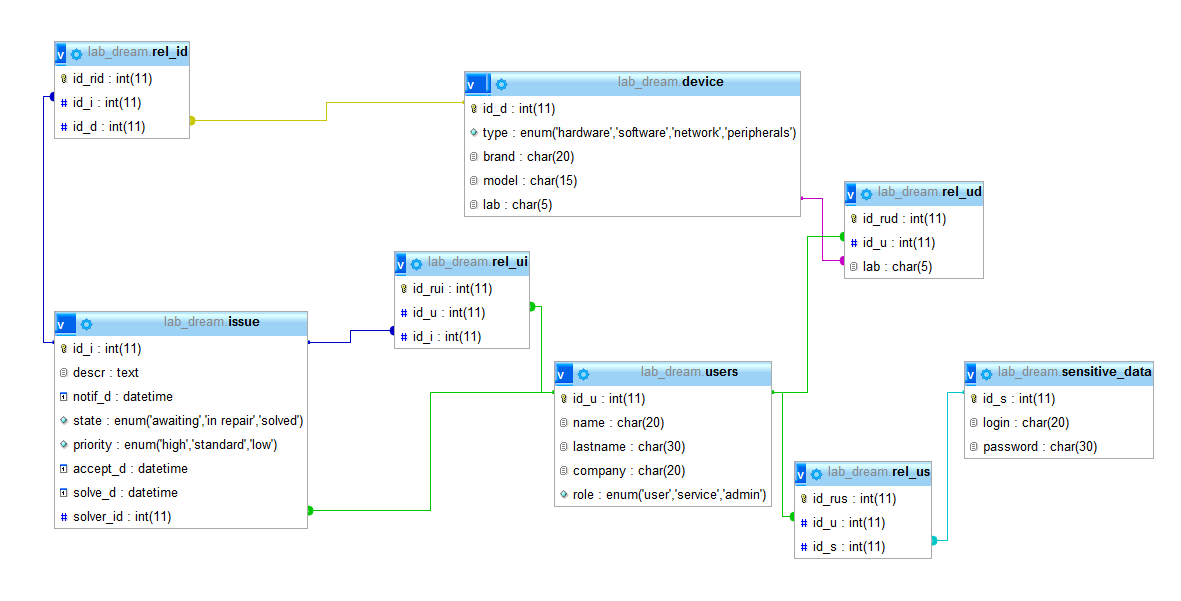
1. Treść ogólna zlecenia:

system do zarządzania zgłoszeniami serwisowymi:

* baza danych o awarii,
* czas zgłoszenia,
* priorytet,
* użytkownicy,
* laboratoria i urządzenia,
* data rozwiązania problemu,
* statystyki
* wow: powiadomienia typu push w aplikacji mobilnej

1. Środowisko:
   1. Baza danych: PostgreSQL
   2. Środowisko backend’owe: Node.js + pg
   3. Środowisko frontend’owe: Ionic + Angular + TypeScript
2. Schemat graficzny logiki





1. Baza:
   1. Tabele
      1. **users**
         * id\_u
         * name
         * lastname
         * company
         * role (rola określająca prawa, np. użytkownik, administrator, serwisant)
      2. **issue**
         * id\_i
         * descr (opis usterki)
         * notif\_d (data, uzupełnia się gdy zostanie dodane zgłoszenie)
         * state (status, np. oczekuje, przyjęte, wykonano…)
         * priority (Zależny od zleceniodawcy, priorytet naprawy)
         * accept\_d (data kiedy serwisant zmienia status na przyjęto)
         * solve\_d (data rozwiązania, pusta dopóki ktoś nie zmieni ‘state’ na ukończono/wykonano)
         * solver\_id (serwisant, który naprawił)
      3. **device**
         * id\_d (numer przypisany urządzeniu przez uczelnie)
         * type (komputer, laptop, myszka, router…)
         * brand
         * model (model, np. Galaxy S5)
         * lab (nr sali)
      4. **sensitive\_data**

* id\_s
* login
* password
  1. Połączenia
     1. **users -> issue 1:n**
        + tabela relacji (id\_u|id\_i)
     2. **users -> sensitive\_data 1:1**
* tabela relacji (id\_u|id\_s)
  + 1. **issue -> device 1:1**
* tabela relacji (id\_i|id\_d)
  + 1. **users -> device 1:n**(przypisanie serwisanta do laboratorium)
* tabela relacji (id\_u|lab)

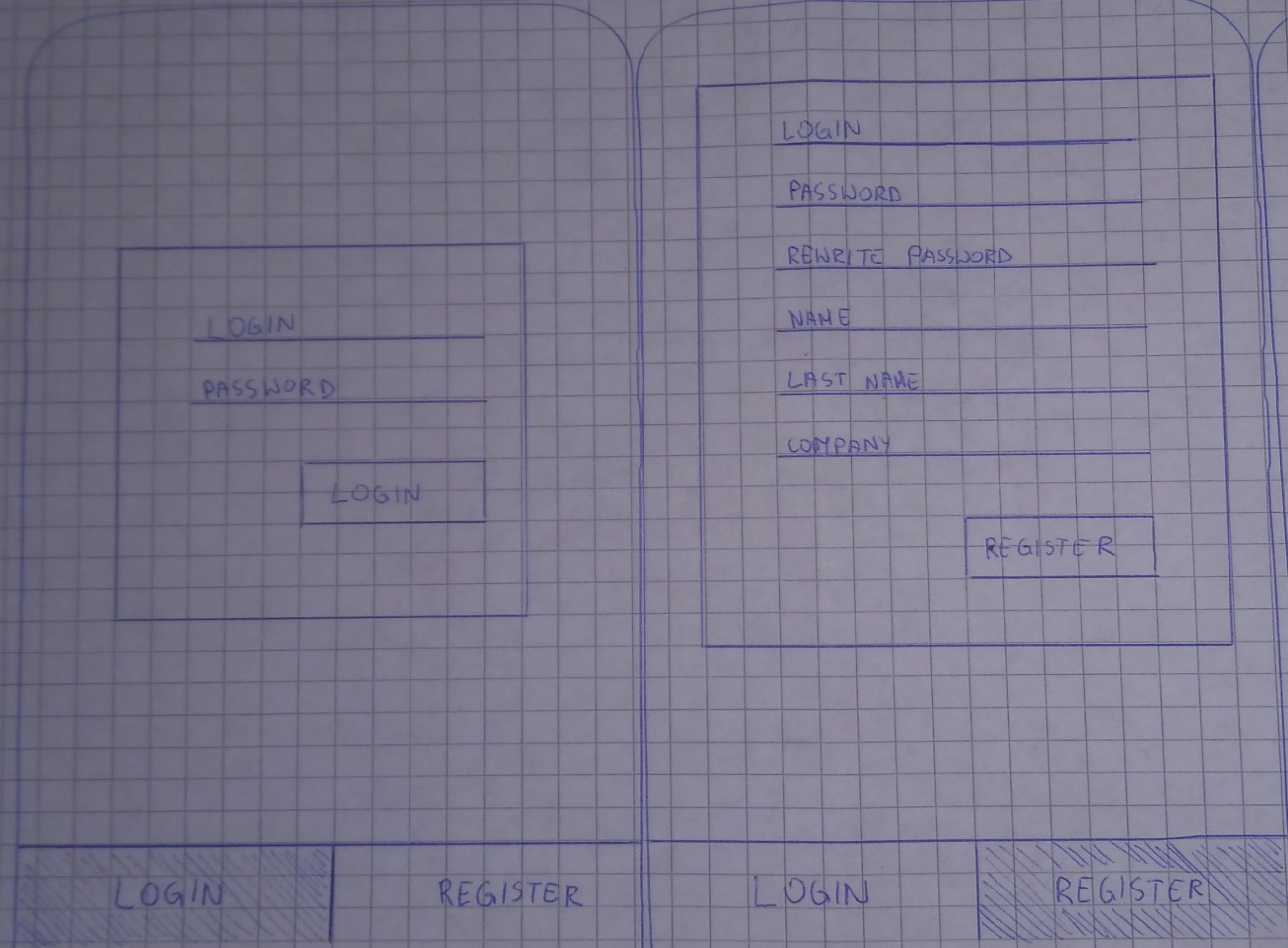
1. Funkcjonalność:
   1. Ogólne
      1. Rejestracja
      2. Logowanie
      3. Zabezpieczenia (jak zostanie czas)
      4. powiadomienia push włączone w tle( jak będzie czas to przy wyłączonej też)
   2. Użytkownik
      1. Panel dodanych zgłoszeń (tylko swoje, edycja, flaga rozwiązania problemu)
      2. Możliwość dodania zgłoszenia
   3. Administrator
      1. Panel zarządzania wszystkimi użytkownikami
      2. Panel dodanych zgłoszeń (wszystkie, usunięcie/edycja, przypisanie do serwisanta)
      3. Statystyki
         * Tydzień/miesiąc
         * Dla danego serwisanta
         * Ile zgłoszeń, ile rozwiązanych (powiadomiono, przyjęto, rozwiązano)
         * 7 dni wstecz, 30 wstecz
   4. Serwisant
      1. Edycja tylko statusu zgłoszeń
      2. Statystyki tygodniowe/miesięczne „swoich” zgłoszeń
2. Planowane funkcje, określone na podstawie powyższych wymagań:

* UŻYTKOWANIE
  + Logowanie
    - Podaje tablice z loginem I haslem
    - Bool
  + Rejestracja
    - Funkcja będzie istnieć jako dodawanie użytkownika (opisane ponizej)
* ZGŁOSZENIA
  + Wszystkie zgłoszenia
    - Nic nie podaję
    - Zwraca tablicę ze wszystkimi zgłoszeniami
  + Usuwanie zgloszenia
    - Podaje id
    - bool, usuwa zgloszenie
  + Pobierz powiazania (urzadzenia)
    - Podaje id zgloszenia
    - Zwraca tablice ze wszystkimi powiazanymi urządzeniami (obiekty)
  + Edycja zgłoszenia
    - Podaje id, nazwe wiersza do zmiany (np. ‘type’ jak w modelu), nowa wartość
    - bool, edytuje zgłoszenie (przy zmianie statusu nadpisuje odpowiednia date)
  + Dodawanie zgłoszenia
    - Podaje opis, priorytet, tablice z urządzeniami
    - Bool, przypisuje do serwisanta który ma najmniej zgloszen (jeśli ktores powiazanie jest nowe to je tworzy)
  + Zwraca wszystkie urządzenia
    - Nic nie podaje
    - Zwraca tablicę ze wszystkimi urzadzeniami
* ZARZĄDZANIE UŻYTKONIKAMI
  + Wszyscy użytkownicy
    - Nic nie podaje
    - Zwraca tablice ze wszystkimi uzytkownikami
  + Pobierz login
    - Podaje id uzytkownika
    - Zwraca login tego uzytkownika
  + Usuwanie
    - Podaje id usera
    - Bool, usuwa usera
  + Dodaje uzytkownika
    - Przesylam obiket użytkownika bez id, tablice z loginem I haslem
    - bool, tworzy użytkownika (chyba ze login jest zajety) wtedy zwraca tez komunikat
  + edytuj użytkownika
    - Przesylam obiket użytkownika razem z id, tablice z loginem I haslem
    - Bool, edytuje istniejącego użytkownika
* STATYSTYKI
  + Zwraca wszystkie staty
    - Podaje id użytkownika (może być puste)
    - Zwraca statystyki
      * Puste id- zwraca statystyki dla wszystkich
      * Jakies id- zwraca statystyki dla tego konkretnego id
  + FORMAT STYSTYK JAKIE MA ZWROCIC

[ [1,3, 5],

[20,46,42],

[800,760,876] ] – ma to być tablica 3 tablic, 1 tabela to tygodniowe, 2 miesieczne, 3 roczne, 1 kolumna to powiadomiono, 2 przyjeto, 3 rozwiazano (czyli zwracać ma tylko ilości zgloszen)

Poglądowe schematy wyglądu poszczególnych paneli:

