



# Planeruz

## Część teoretyczna

Zaawansowane technologie usług sieciowych

19.01.2021

---

Prowadzący: dr inż. Piotr Powroźnik

Grupa: 33INF-SSI-SP/B

Erwin Konkel

Mateusz Znojek

Stanisław Mól - Scrum Master

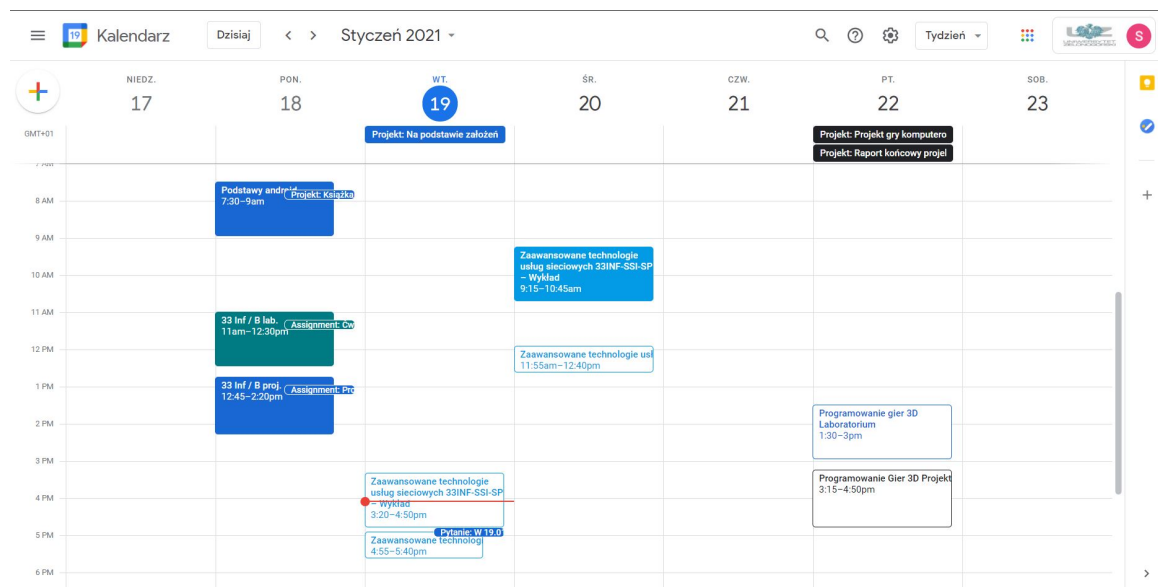
# 1 Opis systemu

System przeznaczony do nakładania planu zajęć Uniwersytetu Zielonogórskiego na czytelną siatkę, na wzór do kalendarza google. Użytkownik ma możliwość wyboru kierunku, grupy oraz podgrupy, względem której ma zostać zmapowany plan. Informacje te zostaną zapamiętane dla zalogowanego użytkownika.

- Grupą docelową systemu są studenci Uniwersytetu Zielonogórskiego.
- Grupą opcjonalną, która mogła by w przyszłości korzystać z możliwości planu, są osoby prowadzące zajęcia.

## 2 Opis istniejących rozwiązań

Najszerzej znanym rozwiązaniem jest "G Suit" działający w chmurze obliczeniowej pakiet zwiększający produktywność oraz oprogramowanie do pracy grupowej i oprogramowanie oferowane przez Google na zasadzie subskrypcji. Na jego podstawie można wywnioskować, że czytelność planu odgrywa główną rolę i właśnie na tym aspekcie zostanie skupiony porojekt. Głównym elementem na którym wzorowany będzie cały projekt jest kalendarz google widocznym na rysunku 1.



Rysunek 1: Kalendarz Google

### 3 Funkcjonalności

| Niezałogowany użytkownik                                    | Zalogowany użytkownik                                      |
|---|--|
| logowanie   | wylogowanie  |
| rejestracja   | możliwość dodania własnego zdarzenia                       |
| wybór grupy dziekańskiej bez możliwości zapamiętania wyboru | wybór grupy dziekańskiej z możliwością zapamiętania wyboru |
| podgląd planu zajęć wybranej grupy dziekańskiej             | podgląd swojego planu                                      |
| wybór motywu  | możliwość dodania informacji o kolokwium do zajęć          |
| -   | wysyłanie powiadomienia e-mail z przypomnieniem            |
| -   | wyszukiwanie wykładowcy                                    |
| -   | wybór motywu   |
| -   | edycja zdarzeń zawartych w kalendarzu                      |
| -   | tablica TODO   |

### 4 Opis funkcjonalności

- **Rejestracja / Logowanie / Wylogowanie** - prosta funkcjonalność pozwalająca na zarejestrowanie się użytkownika, następnie autoryzację na podstawie podanego loginu i hasła, oraz wylogowanie po zakończonej sesji.
- **Wybór grupy dziekańskiej** - użytkownik niezalogowany będzie miał możliwość wyboru grupy i podglądu planu zajęć dla danej grupy, jednakże nie będzie miał możliwości zapamiętania tego wyboru. Dla użytkownika zalogowanego będzie istniała możliwość zapamiętania wybranej grupy, początkowo wybór ten będzie możliwy podczas rejestracji, lecz będzie można zmienić wybór w panelu użytkownika.
- **Podgląd planu** - plan będzie wyświetlany w czytelnej formie, dzięki nałożeniu na siatkę czasu (podobnie jak w kalendarzu google).
- **Dodanie informacji o kolokwium do zajęć** - zalogowany użytkownik będzie miał możliwość dodania informacji czy na danych zajęciach zaplanowane jest kolokwium.
- **Dodanie własnego zdarzenia** - zalogowany użytkownik ma możliwość dodania własnego zdarzenia.

- **Wysłanie powiadomienia e-mail** - po wcześniejszym zaznaczeniu opcji system będzie wysyłał powiadomienie e-mail o zbliżających się kolokwiach (jeżeli są dodane).
- **Wyszukanie wykładowcy** - możliwość wyszukania wykładowcy w systemie i znalezienia potrzebnych informacji o wykładowcy.
- **Wybór motywu** - użytkownik może wybrać motyw aplikacji (ciemny / jasny).
- **Edycja zdarzeń** - użytkownik zalogowany ma możliwość edytowania zdarzeń na własnym planie.
- **Tablica TODO** - tablica z zadaniami do wykonania na następny dzień.

## 5 Opis problemów

Główne problemy przewidziane podczas projektowania, oraz proponowane rozwiązania:

- Stworzenie bazy danych - ważne zagadnienie które może być problematyczne - rozwiązaniem problemu może być użycie 'Hibernate'.
- Przechowywanie danych użytkowników w bazie - jednym z problemów będzie przechowywanie danych wrażliwych takich jak hasła - rozwiązaniem tego problemu będzie przechowywanie zahashowanych danych.

## 6 Diagram bazy danych i relacji

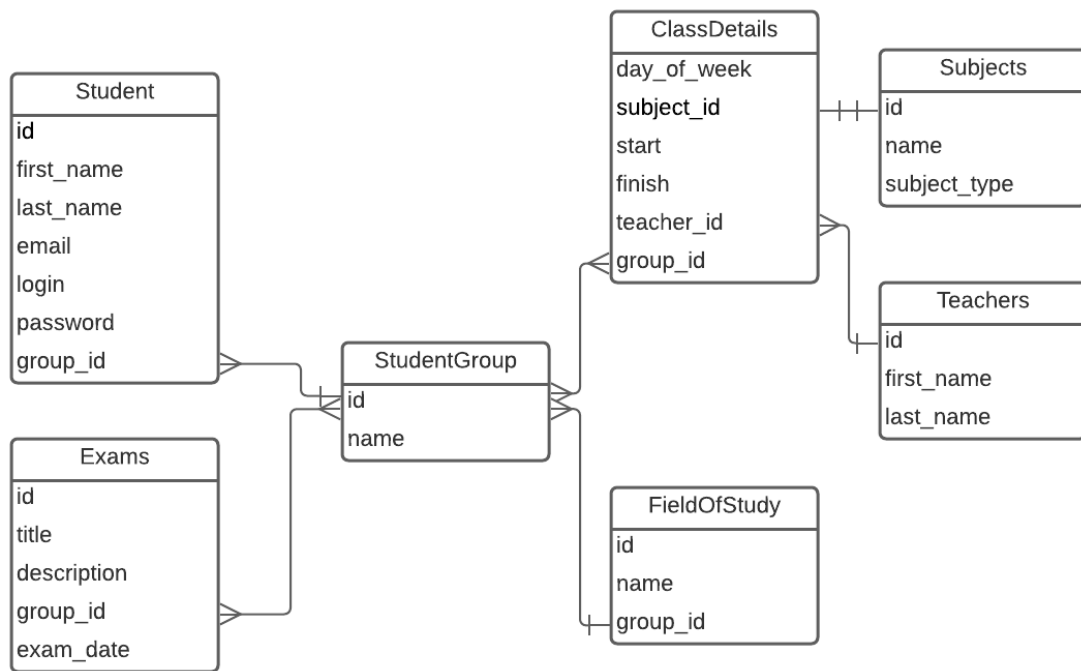
Diagram bazy danych prezentuje się jak na rysunku 2. Trzonem łączącym grupy jest grupa, która posiada najważniejsze relacje łączące studenta z danymi o konkretnych zajęciach.

## 7 Diagram użycia

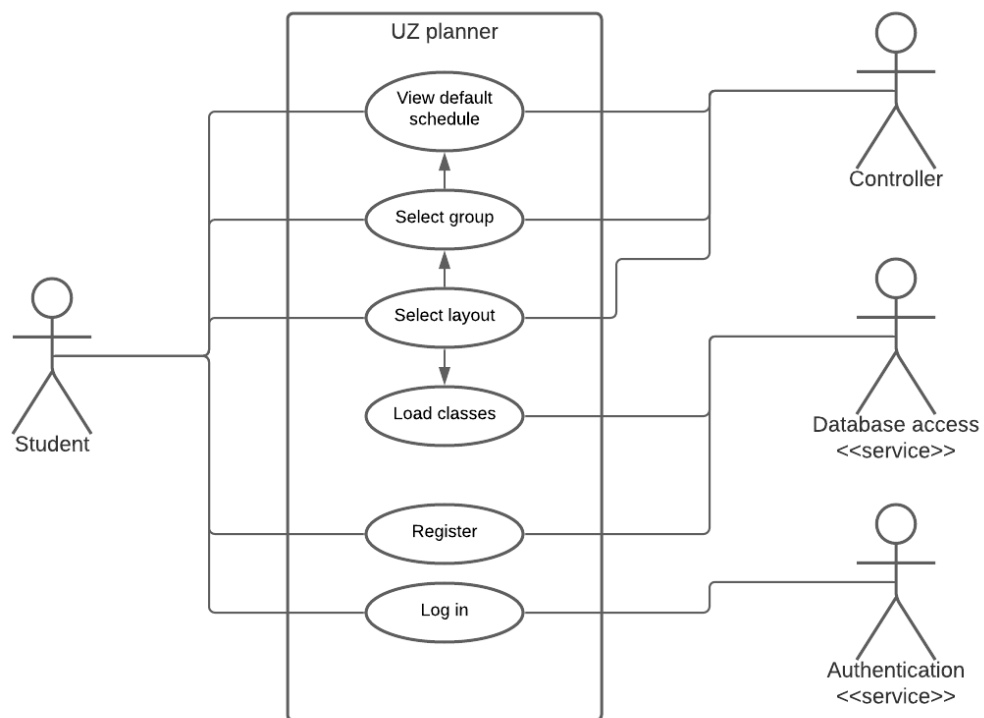
Diagram użycia prezentuje się jak na rysunku 3. Prezentuje on podstawowe funkcjonalności na których się skupiliśmy w budowie tej platformy tj. wyświetlanie odpowiednio sformatowanej tabeli z zajęciami konkretnej grupy lub zarejestrowanego użytkownika.

## 8 Tabele przypadków użycia

Tabele przypadków użycia to kolejno dla każdej funkcjonalności: rejestracja rysunek 4, logowanie rysunek 5, zmiana motywu platformy 6, wyświetlanie głównego widoku planu rysunek 7, wybieranie grupy rysunek 8, wybieranie układu widoku planu rysunek 9



Rysunek 2: Diagram bazy danych



Rysunek 3: Diagram użycia (*Use Case*)

|                 |   |
|-----------------|---|
| Nazwa           | Rejestracja   |
| Opis            | Użytkownik wprowadza swoje dane do bazy serwisu   |
| Aktorzy         | Użytkownik, serwer, baza danych   |
| Scenariusz      | <ol style="list-style-type: none"> <li>1. Użytkownik wchodzi w podstronę rejestracji</li> <li>2. System wyświetla formularz z polami do wypełnienia</li> <li>3. Użytkownik podaje dane i wysyła formularz do bazy</li> <li>4. Baza zapisuje dane o użytkowniku</li> </ol> |
| Warunek wstępny | Przejdzie do formularza   |
| Warunek końcowy | Poprawne uzupełnienie i wysłanie formularza   |
| Rezultat        | Student posiada konto na platformie   |

Rysunek 4: Tabela rejestracji

|                 |   |
|-----------------|---|
| Nazwa           | Logowanie   |
| Opis            | Użytkownik ma dostęp do funkcji dla zalogowanych użytkowników   |
| Aktorzy         | Użytkownik, serwer, serwis autoryacyjny, baza danych  |
| Scenariusz      | <ol style="list-style-type: none"> <li>1. Użytkownik wchodzi w formularz logowania</li> <li>2. Użytkownik wprowadza poprawny login i hasło</li> <li>3. Serwer rozpoczyna sesję dla uwierzytelnionego użytkownika po potwierdzeniu danych</li> </ol> |
| Warunek wstępny | Użytkownik istnieje w bazie   |
| Warunek końcowy | Użytkownik wyśle poprawny formularz   |
| Rezultat        | Student ma dostęp do funkcji dla zalogowanych   |

Rysunek 5: Tabela logowania

|                 |   |
|-----------------|---|
| Nazwa           | Zmiana motywu platformy   |
| Opis            | Użytkownik zmienia kolorystykę platformy  |
| Aktorzy         | Użytkownik, widok platformy, serwer   |
| Scenariusz      | <ol style="list-style-type: none"> <li>1. Użytkownik przechodzi na widok planu</li> <li>2. Wciska przycisk odpowiedzialny za zmianę motywu</li> <li>3. Otrzymuje przełączoną kolorystykę między jasną i ciemną</li> </ol> |
| Warunek wstępny | Użytkownik jest na widoku planu   |
| Warunek końcowy | Użytkownikowi wciska przycisk zmiany  |
| Rezultat        | Kolorystyka ulega zmianie   |

Rysunek 6: Tabela zmiany motywu

|                 |  |
|-----------------|--|
| Nazwa           | Wyświetlanie planu   |
| Opis            | Użytkownikowi wyświetla się widok planu zajęć na stronie   |
| Aktorzy         | Użytkownik, serwer, widok planu, baza danych   |
| Scenariusz      | <ol style="list-style-type: none"> <li>1. Użytkownik wchodzi na stronę główną</li> <li>2. Wyświetla mu się plan pierwszej grupy</li> <li>3. Użytkownik wybiera interesującą go grupę</li> <li>4. Jeżeli użytkownik jest zalogowany dostaje domyślnie widok swojej grupy</li> </ol> |
| Warunek wstępny | Przejdzie na stronę główną   |
| Warunek końcowy | Wybór grupy  |
| Rezultat        | Wyświetlany jest czytelny plan zajęć w odpowiednio sformatowanej tabeli  |

Rysunek 7: Tabela wyświetlania planu

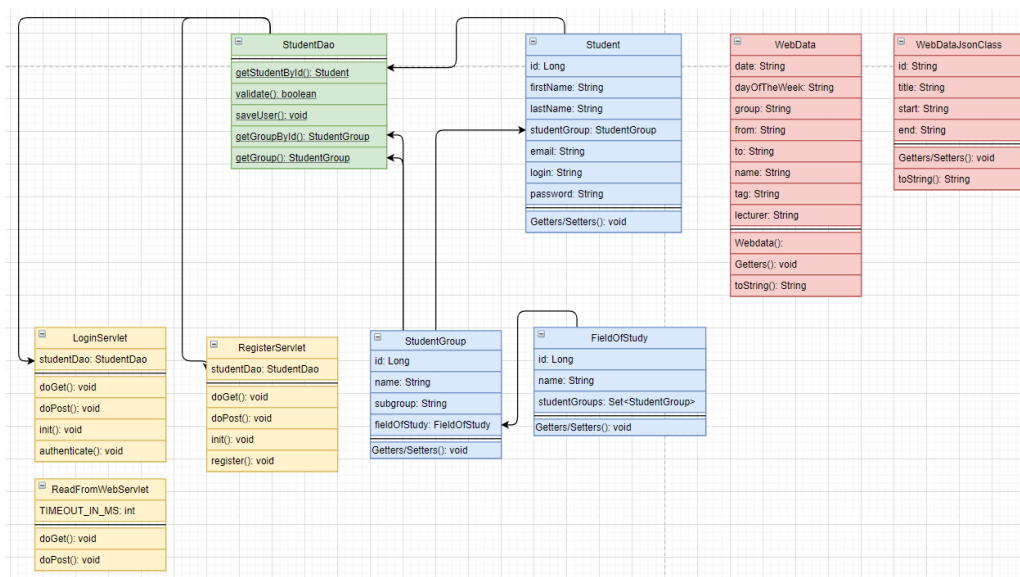
|                 |   |
|-----------------|---|
| Nazwa           | Wybór grupy   |
| Opis            | Użytkownik wybiera grupę do wyświetlania planu  |
| Aktorzy         | Użytkownik, serwer, serwis planu, baza danych   |
| Scenariusz      | <ol style="list-style-type: none"> <li>1. Użytkownik przechodzi na widok planu</li> <li>2. Wybiera interesującą go grupę</li> <li>3. Otrzymuje widok tej grupy</li> </ol> |
| Warunek wstępny | Wejście na adres z widokiem planu   |
| Warunek końcowy | Wybranie dostępnej grupy studenckiej  |
| Rezultat        | Otrzymanie czytelnego widoku planu dla zdefiniowanej w serwisie grupy   |

Rysunek 8: Tabela wyboru grupy

|                 |  |
|-----------------|--|
| Nazwa           | Wybór układu   |
| Opis            | Użytkownik wybiera dostępny układ wyświetlanego planu  |
| Aktorzy         | Użytkownik, widok planu  |
| Scenariusz      | <ol style="list-style-type: none"> <li>1. Użytkownik przechodzi na widok planu</li> <li>2. Wybiera interesujący go dostępny widok</li> <li>3. Otrzymuje sformatowany widok w zależności od wyboru</li> </ol> |
| Warunek wstępny | Wejście na adres z widokiem planu  |
| Warunek końcowy | Wybranie widoku z dostępnych widoków   |
| Rezultat        | Otrzymanie układu o interesującym użytkownika formacie   |

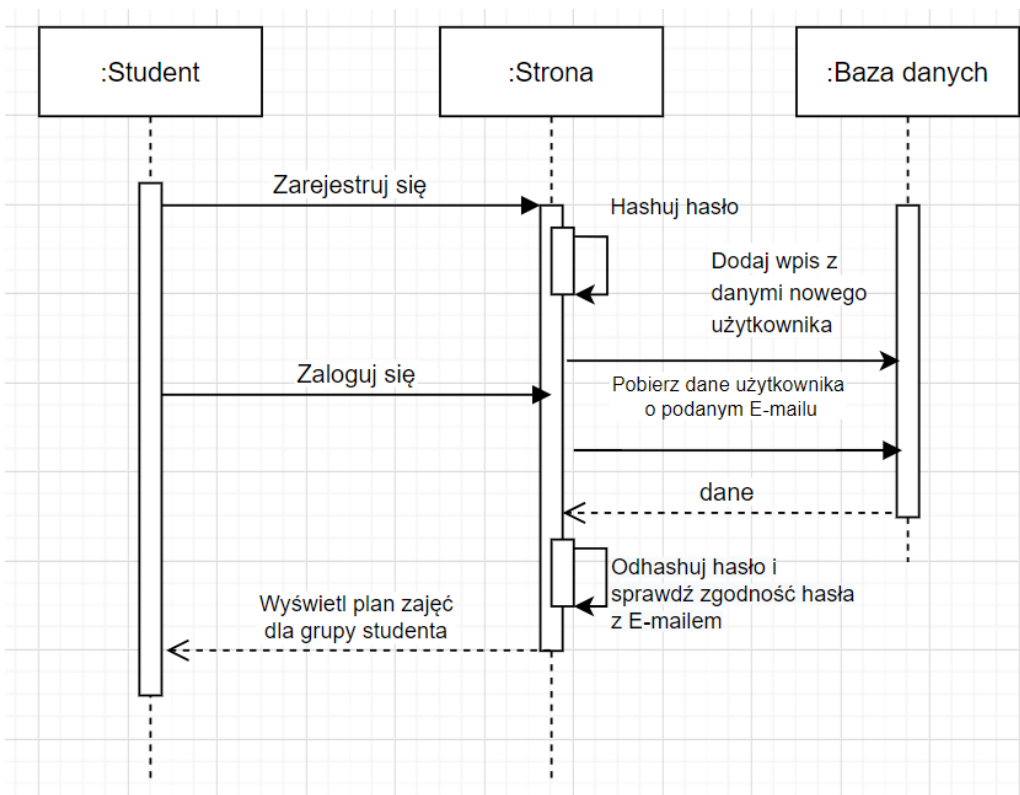
Rysunek 9: Tabela wyboru układu

## 9 Diagram komponentów/obiektów



Rysunek 10: Diagram klas

## 10 Diagram przepływu sterowania

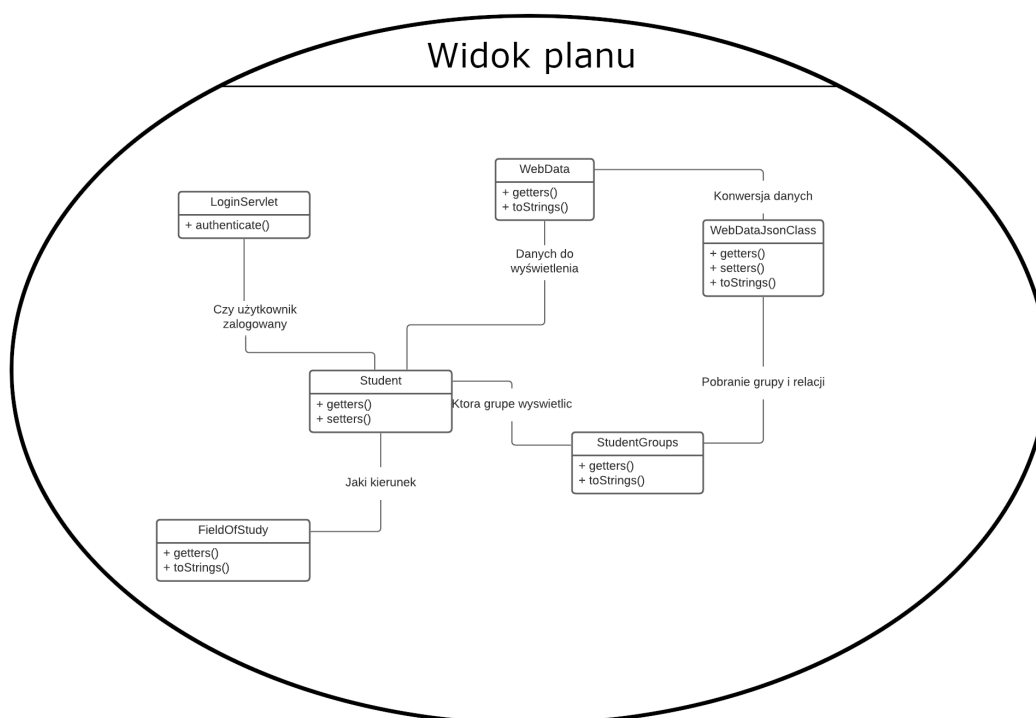


Rysunek 11: Diagram przepływu sterowania



## 11 Diagram funkcjonalny/strukturalny

Diagram strukturalny prezentuje się jako na rysunku 12. Platforma po otwarciu ładuje podstawowe struktury danych czyli klasę odpowiedzialną za parsowanie wszystkich zajęć i wysyłanie ich do reszty komponentów oraz jednocześnie klasa pomocnicza która konwertuje dane na format json. Następnie ładowany jest główny widok planu wraz z jego komponentami w formacie jsp. Dalej ładowane są grupy oraz kierunki które można wybrać z bazy danych. Wybrana grupa wysyłana jest do serwisu odpowiedzialnego za dane do wyświetlenia i zapytanie zostaje obsłużone oraz zwracane są odpowiednie zajęcia. W przypadku zalogowanego użytkownika jego grupa ładowana jest automatycznie

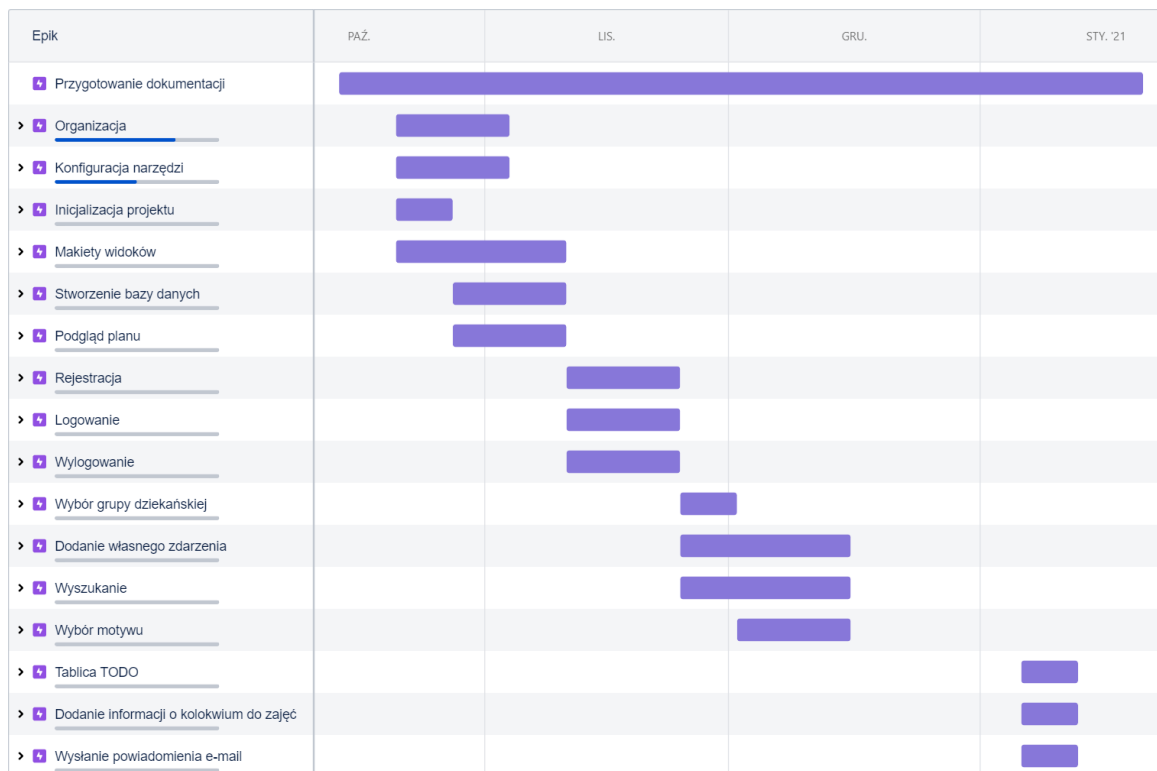


Rysunek 12: Diagram strukturalny

## 12 MOSCOW

| Must  | Should   | Could   | Won't Have              |
|---|--|---|-------------------------|
| logowanie   | możliwość dodania własnego zdarzenia           | możliwość dodania informacji o kolokwium do zajęć | wybór motywu            |
| wybór grupy dziekańskiej bez możliwości zapamiętania wyboru | wysłanie powiadomienia e-mail z przypomnieniem | -   | wyszukiwanie wykładowcy |
| podgląd planu zajęć wybranej grupy dziekańskiej             | edycja zdarzeń zawartych w kalendarzu          | -   | tablica TODO            |
| wylogowanie   | -  | -   | -                       |
| rejestracja   | -  | -   | -                       |
| wybór grupy dziekańskiej z możliwością zapamiętania wyboru  | -  | -   | -                       |
| podgląd swojego planu                                       | -  | -   | -                       |

## 13 Harmonogram zadań



Rysunek 13: Diagram Gantta

| Zadanie                                 | Stanisław Mól | Mateusz Znojek | Erwin Konkel |
|---|---------------|----------------|--------------|
| Przygotowanie dokumentacji              | ✓             |                |              |
| Organizacja                             | ✓             | ✓              | ✓            |
| Konfiguracja narzędzi                   | ✓             | ✓              | ✓            |
| Inicjalizacja projektu                  | ✓             |                |              |
| Makiety widoków                         |               |                | ✓            |
| Stworzenie bazy danych                  |               | ✓              |              |
| Podgląd planu                           | ✓             |                |              |
| Rejestracja                             |               | ✓              |              |
| Logowanie                               |               |                | ✓            |
| Wylogowanie                             |               | ✓              |              |
| Wybór grupy dziekańskiej                |               |                | ✓            |
| Dodanie własnego zdarzenia              |               | ✓              |              |
| Wyszukiwanie                            | ✓             |                |              |
| Wybór motywu                            |               |                | ✓            |
| Tablica TODO                            |               | ✓              |              |
| Dodanie informacji o kolokwium do zajęć | ✓             |                | ✓            |
| Wysłanie powiadomienia e-mail           | ✓             | ✓              | ✓            |

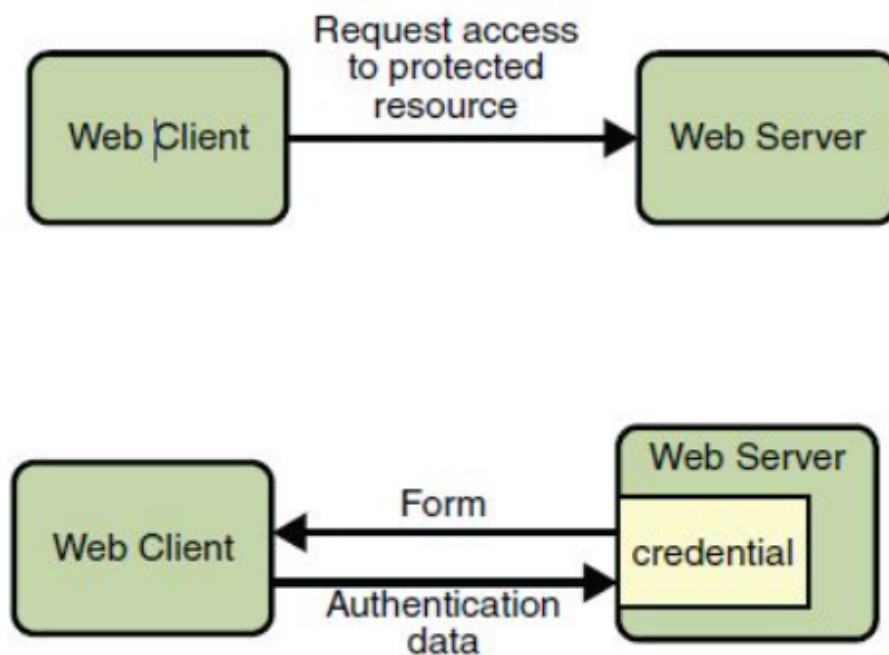
## 14 Plan testów

Plan testów zakłada testowanie każdej historyjki przed oddaniem jej. Oznacza to, że każda funkcjonalność powinna zostać przetestowana przed dodaniem jej do głównej gałęzi projektu. Testy powinny być przeprowadzone na podstawie wyznaczonych kryteriów akceptacji.

## 15 Plan bezpieczeństwa

Plan bezpieczeństwa zakłada dostarczenie możliwie uniwersalnych mechanizmów dla:

- uwierzytelniania (ang. authentication) - jestem tym za kogo się podaję bo: znam pin, znam hasło, mam takie odciski palców, posiadam certyfikat SSL, itp.
- autoryzacji (ang. authorization) - mam dostęp do określonych zasobów, a do innych nie



Rysunek 14: Uwierzytelnienie klienta web

## 16 Plan konserwacji

Plan konserwacji obejmuje weryfikację oraz poprawianie błędów w trakcie implementacji każdej z nowych funkcjonalności, co oznacza, że każda funkcjonalność powinna zostać szczegółowo sprawdzona, po czym każdy z wykrytych błędów powinien zostać naprawiony. Po zakończeniu projektu, nie są planowane kolejne aktualizacje.

## 17 System kontroli wersji

W projekcie został wykorzystany system kontroli wersji GIT ze względu na:

1. Dobre wsparcie dla rozgałęzionego procesu tworzenia oprogramowania
2. Praca off-line: każdy programista posiada własną kopię repozytorium, do której może zapisywać zmiany bez połączenia z siecią
3. Efektywna praca z dużymi projektami.

Repozytorium z projektem jest dostępne pod linkiem "Planneruz".

## 18 Zadania zrealizowane

| Zadanie                                 | Zrealizowane |
|---|--------------|
| Przygotowanie dokumentacji              | ✓            |
| Organizacja                             | ✓            |
| Konfiguracja narzędzi                   | ✓            |
| Inicjalizacja projektu                  | ✓            |
| Makiety widoków                         | ✓            |
| Stworzenie bazy danych                  | ✓            |
| Podgląd planu                           | ✓            |
| Rejestracja                             | ✓            |
| Logowanie                               | ✓            |
| Wylogowanie                             | ✓            |
| Wybór grupy dziekańskiej                | ✓            |
| Dodanie własnego zdarzenia              |              |
| Wyszukiwanie                            |              |
| Wybór motywu                            | ✓            |
| Tablica TODO                            |              |
| Dodanie informacji o kolokwium do zajęć |              |
| Wysłanie powiadomienia e-mail           |              |

## 19 Wkład poszczególnych członków w projekt

Grupa od samego początku pracowała równo i to też tempo zostało utrzymane przez wszystkie sprinty z wyjątkiem Sprintu 4, w którym to jeden z członków - Stanisław Mól - zachorował i nie był w stanie kontynuować swoich prac, przez co praca spadła na pozostałych członków. Ten też fakt spowodował znaczne opóźnienie w projekcie co skutkowało nie wykonaniem wszystkich założeń. Pomimo to można stwierdzić, że grupa współpracowała bardzo profesjonalnie i wsparcie w poszczególnych problemach, poszczególnych członków grupy było znaczące i pozwalało posuwać pracę do przodu. Z każdym kolejnym sprintem grupa była coraz lepiej zgrana i coraz bardziej doświadczona, co też pozwoliło rozwiązać początkowe problemy związane z pierwszymi krokami w nowej technologii, oraz problemy z systemem kontroli wersji.

## 20 Warunki licencji

Planneruz jest wolnym oprogramowaniem: możesz go rozprowadzać dalej i/lub modyfikować na warunkach Powszechnej Licencji Publicznej GNU, wydanej przez Fundację Wolnego Oprogramowania - według wersji 3 tej Licencji.

Planneruz rozpowszechniany jest z nadzieją, iż będzie on użyteczny - jednak BEZ JAKIEJKOLWIEK GWARANCJI, nawet domyślnej gwarancji PRZYDATNOŚCI HANDLOWEJ albo PRZYDATNOŚCI DO OKREŚLONYCH ZASTOSOWAŃ. W celu uzyskania bliższych informacji sięgnij do Powszechnej Licencji Publicznej GNU.

Z pewnością wraz z Planneruz otrzymałeś też egzemplarz Powszechnej Licencji Publicznej GNU (GNU General Public License). Jeśli nie - zobacz <http://www.gnu.org/licenses>.

## 21 Problemy które wystąpiły podczas implementacji projektu

Podczas implementacji projektu wystąpiły dwa znaczące problemy, które znacznie opóźniły prace nad projektem:

- **Baza danych** - znaczne ilości czasu wszystkich członków grupy pochłoneła baza danych, ze względu na rozpoznanie kilku narzędzi i faktem, że większość z nich nie chciała zadziałać. Dopiero wykorzystanie Hibernate, oraz zdalnej bazy danych na Heroku, pozwoliło na kontynuowanie prac.
- **Choroba członka grupy** - w pewnym momencie jeden z członków grupy rozchorował się, przez co tempo pracy znacząco spadło o 33 procent wydajności. Fakt ten miał znaczący wpływ na prace nad projektem.

## 22 Podsumowanie

Podsumowując cały projekt, na pewno nauczył on całą grupę nowej technologii, pracy w grupie, i korzystania z systemu wersji. Zdobyta wiedza na pewno będzie bezcenna w przyszłości np. w przyszłej pracy. Poimmo faktu, że projekt nie został ukończony w 100 procentach, z powodu problemów wymienionych w "Problemy które wystąpiły podczas implementacji projektu", to jest on w użytecznej formie i wszystkie najważniejsze funkcjonalności zostały zaimplementowane. Cała grupa pracowała bardzo sprawnie, i mając na uwadze opóźnienia każdy przykładał się najlepiej jak potrafił, aby jak najwięcej zostało ukończone. Główne założenie projektu na pewno zostało spełnione, czyli czytelność planu - dzięki włożonemu trudowi plan został przedstawiony w czytelnej formie.