

PLatforma Testowa

Projekt na inżynierię oprogramowania



18 stycznia 2023

AGH, IT 2. rok, 3. semestr

Grzegorz Łaziński, Oskar Maciaszek

Spis treści

[Specyfikacja projektu 2](#_Toc124987271)

[Koncepcja produktu 2](#_Toc124987272)

[Organizacja projektu 2](#_Toc124987273)

[Zarządzanie danymi 3](#_Toc124987274)

[Metodyka 3](#_Toc124987275)

[Słownik pojęć – glossary 4](#_Toc124987276)

[Diagramy przypadków użycia 5](#_Toc124987277)

[Diagram wymagań funkcjonalnych 7](#_Toc124987279)

[Diagram wymagań niefunkcjonalnych 7](#_Toc124987280)

[Zastosowane framework-i, środowiska, technologie 7](#_Toc124987281)

[Diagram klas 8](#_Toc124987282)

[Diagramy aktywności 8](#_Toc124987283)

[Diagramy sekwencji 10](#_Toc124987284)

[Diagram komponentów 11](#_Toc124987285)

[Diagram wdrożenia 12](#_Toc124987286)

[Przykłady współpracy zespołu z wykorzystaniem wybranych narzędzi 12](#_Toc124987287)

[Przykład refaktoryzacji 13](#_Toc124987288)

# Specyfikacja projektu

## Koncepcja produktu

Projekt platformy edukacyjnej stanowi klasyczne, minimalistyczne podejście do organizacji przebiegu testów składających się z pytań zamkniętych. Serwis internetowy oferuje funkcjonalność zawartą w dwóch formułach użytkownika:

Nauczyciela – umożliwiającego customizację formularza testowego ze względu na jego rozmiar, treść pytań czy materiały multimedialne, na które składają się grafiki pomocnicze. Tworzenie grup testowych – jednostek klasyfikujących użytkowników niżej opisanego typu *Uczeń* w celu zautomatyzowania udzielania dostępu do formularza testowego. Bezpośredniej edycji bazy ocen, powiązanej z każdym profilem typu *Uczeń*.

Ucznia – który jest głównym użytkownikiem, wchodzącym w interakcje z podstawowymi funkcjonalnościami produktu, na które składają się same testy czy chat, który ma umożliwić komunikację pomiędzy użytkownikami różnych typów kont w obrębia grupy testowej, jak i poza nią.

## Organizacja projektu

Pliki strony podzielone są ze względu na zamierzone funkcjonalności. W głównym folderze projektu, *Platforma-testowa-IO*, znajdują się pliki *.php*, kod których odpowiedzialny jest za kolejne strony web odwiedzane przez użytkownika w trakcie interakcji z zawartością witryny. Są tu między innymi *index.php*, stanowiący główny hub każdej strony internetowej, czy *profile.php*, odpowiedzialny za stronę profilu użytkownika. W folderze tym znajdują się też kolejne pod-foldery, klasyfikujące treści o innej funkcji. Folder *includes* zawiera pliki, które w procesie korzystania ze strony odpowiedzialne są za obsługę interakcji użytkownika, w sposób dla niego utajony, tj. ich działanie jest ze strony front-endu niewidoczne, bądź niezauważalne. Np. l*ogin.inc.php* czy *logout.inc.php*, które zajmują się kolejno obsługą wprowadzonych do okna logowania danych i zakańczania sesji i przekierowania na stronę główną serwisu. Kolejna grupa plików zrzeszona będzie w folderze pod nazwą *classes*, gdzie zgodnie z nazwą znajduje się kod klas wykorzystywanych w działaniu całej strony na każdym kroku. Znaleźć tu można m.in. klasy obsługi danych wejścia przekazanych przez użytkownika w procesie logowania i rejestracji (*login.class.php* i *signup.class.php*) jak i klasy z nich dziedziczące (*login-contr.class.php* i *signup-contr.class.php*), które zajmują się wstępnym zarządzaniem danych – np. sprawdzają poprawność czy zgodność wprowadzonych danych. Dla organizacji powstały też foldery *css*, przechowujący kaskadowe arkusze styli, *images*, którego zawartość stanowią grafiki, które na stałe są używane w projekcie strony, *uploads*, gdzie przechowywane są pliki przesyłane na stronę przez użytkowników, które mogą być przez nich także aktualizowane. Takowe uporządkowanie pozwala na łatwą orientację w obrębie projektu.

Całościowy projekt zostawiony został z pomocą oprogramowania *XAMPP (modułów Apache i MySQL)*.

## Zarządzanie danymi

Obraz zawierający tekst

Opis wygenerowany automatycznieW celu przechowywania danych użytkowników i zagwarantowania integralności systemu w dowolnym momencie, wykorzystana została baza danych SQL.

Struktura bazy danych projektu: platforma-io  
widziana z panelu phpmyadmin

Bazę danych podzielono na trzy tabele. Pierwszą z nich jest tabela *users*, która zawiera w sobie rekordy zarejestrowanych użytkowników, na które składają się kolejno pola:

*users\_id* – numer rekordu;  
*users\_uid* – nazwa użytkownika;  
*users\_first* – imię użytkownika;  
*users\_last* – nazwisko użytkownika;  
*users\_email* – adres e-mail użytkownika;  
*users\_pwd* – hasło użytkownika, zapisane w formie haszowanej w celu zachowania środków bezpieczeństwo ochrony danych;  
*users\_type* – informacja typu boolean określająca typ kąta (*uczeń*, *nauczyciel*);  
*users\_img* – informacja typu logicznego, mówiąca czy użytkownik przesłał własne zdjęcie profilowe.

Drugą, jest tabela *questions*, w której na każdy rekord przypadają dwa pola:

*question\_nr* – numer identyfikacyjny pytania, automatycznie inkrementowany w bazie danych;  
*text* – treść samego pytania, która zająć może obszar do 65 535 bajtów.

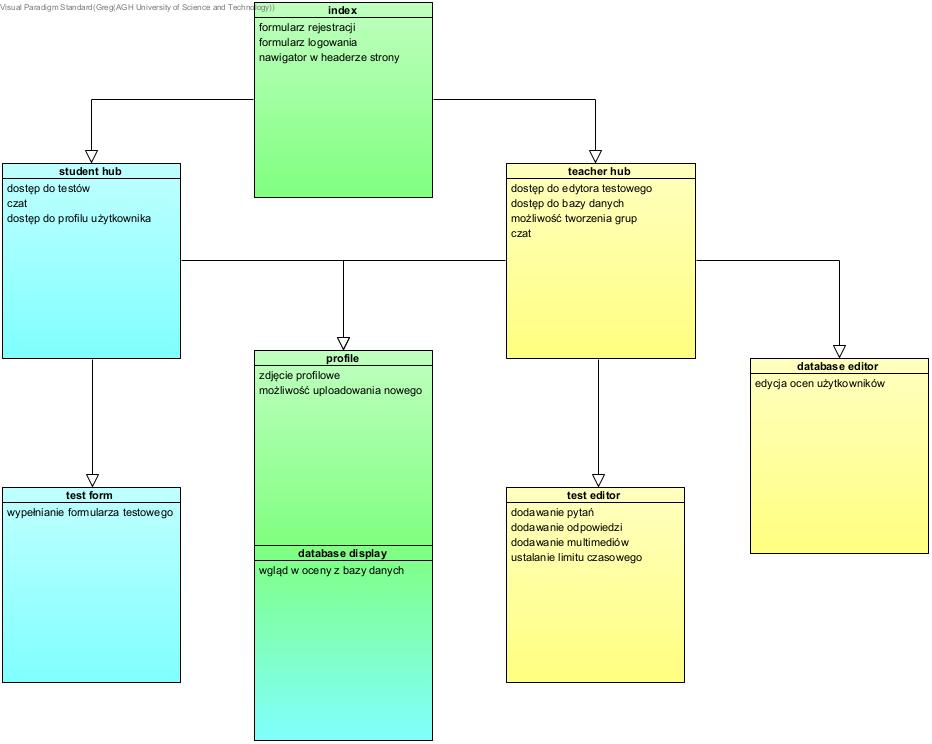
Trzecia tabela, stanowi strukturę komplementarną do tabeli drugiej. W jej skład wchodzą pola:

*id* – identyfikator w bazie;  
*question\_nr* – numer pytania, odpowiadający numerowi przypisanemu w polu o tej samej nazwie w poprzedniej tabeli;  
*is\_correct* – przechowuje wartość logiczną używaną do sprawdzania poprawności odpowiedzi;  
*text* – treść odpowiedzi, która zająć może obszar do 65 535 bajtów.

## Metodyka

Wybrana metodyka jest modelem hybrydowego podejścia do zagadnienia, na które składa się projektowanie prototypowe i wybrane elementy systemów Agile.

Podejście ogólne – projekt rozpoczęto od utworzenia minimalnego szkieletu strony, oferującego prymitywne funkcjonalności. Na wireframe składa się szereg stron, reprezentujących konceptualnie podstrony odpowiedzialne za określone funkcjonalności systemów.



Rysunek 1 - kolor niebieski oznacza dostępność dla użytkownika Uczeń, żółty dla użytkownika Nauczyciel, zielony dla obu

Na bazie powyższego szkieletu powstały projekt był stale rozbudowany w miarę możliwości oraz odzewu z testów. Prace nad kolejnymi elementami zostały podzielone tematyczne pomiędzy twórcami projektu na część dostępową, związaną z dwiema pierwszymi generacjami powyższego drzewa, i testową reprezentowaną przez najniższy poziom. Następnie, rozpoczęto pracę w cyklach, pod koniec których następowało spotkanie, gdzie omawiany i refaktoryzowany był powstały kod.

Przybliżona przed chwilą metoda, ma na celu podejście „od ogółu do szczegółu” do zagadnienia jakim jest nasz system. Skupia się ona na rozwijaniu koniecznych na danym etapie funkcjonalności lub poprawek z zakresu QoS, które w końcowym etapie rozwoju mają stworzyć dopracowaną całość. Główną zaleta takiego podejścia jest elastyczność na zmiany. Powstałe funkcje powstają na bazie bieżących zapotrzebowań wynikających m. in. z testowania.

## Słownik pojęć – glossary

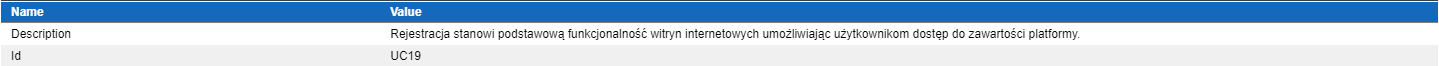
|  |  |
| --- | --- |
| Użytkownik | Dowolny internauta odwiedzający witrynę, który zarejestrowany jest w bazie danych użytkowników. |
| Nauczyciel | Użytkownik, który w procesie rejestracji założył konto typu nauczyciela. Posiada on uprawniania do tworzenie i edycji formularzy testowych, oraz do tworzenia baz i grup testowych oraz ich edycji. |
| Uczeń | Użytkownik, który w procesie rejestracji założył konto typu ucznia. Posiada on możliwość wypełniania formularzy testowych. |
| Test | Jednostka zestawu formularza testowego kreowanego przez profil typu Nauczyciela w obrębie danej grupy, w skład której wchodzą użytkownicy o profilu ucznia. |

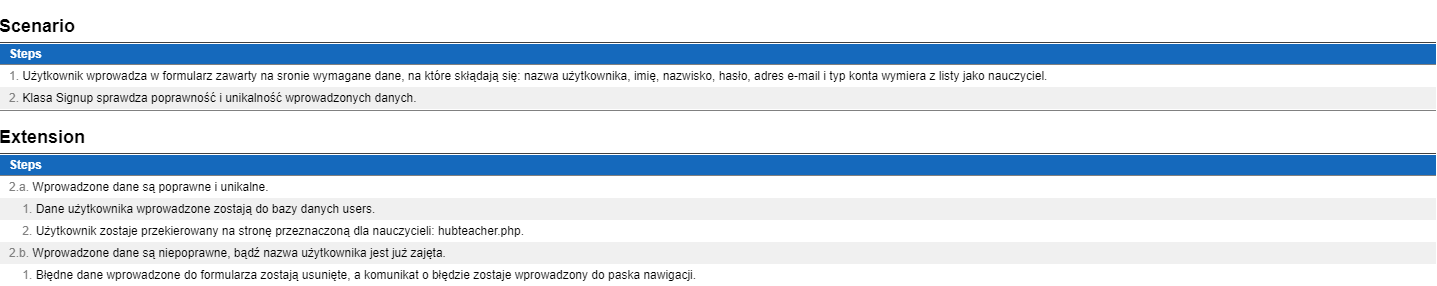
## Diagramy przypadków użycia

Dla aktora będącego Nauczycielem

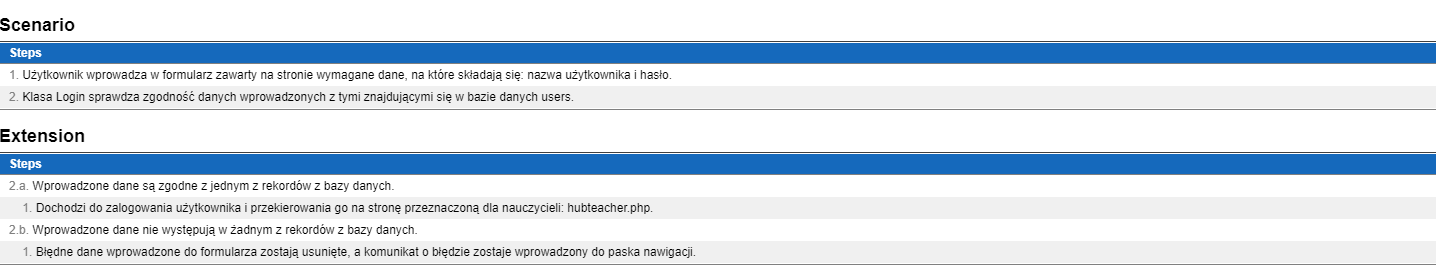
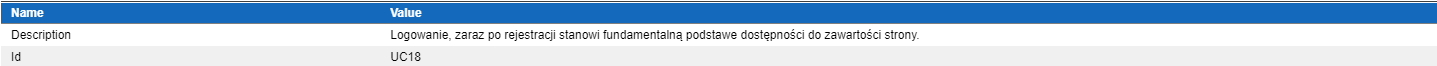
## 

**Use Case – Rejestracja**

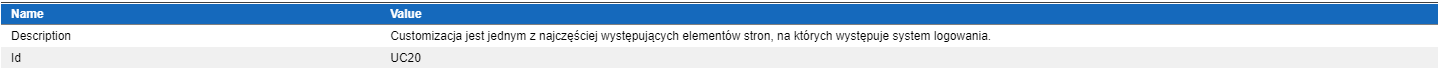


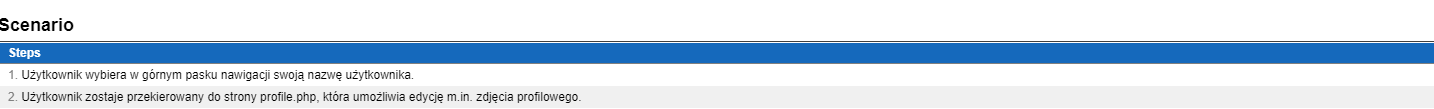


**Use Case – Logowanie**

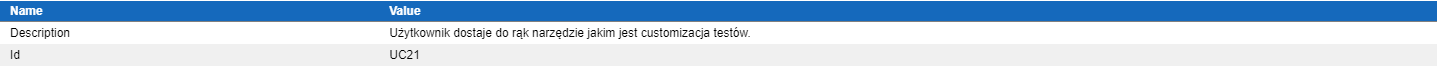


**Use Case – Zarządzanie kontem**

****

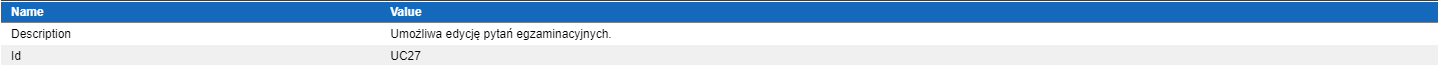
****

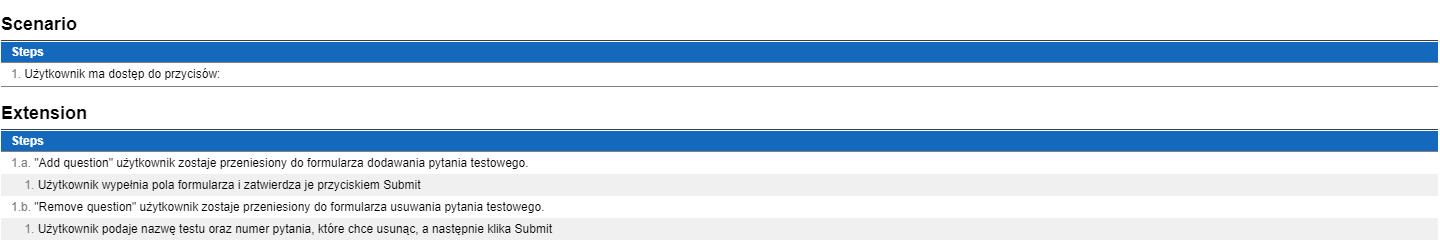
**Use Case – Zarządzanie testem**

****

****

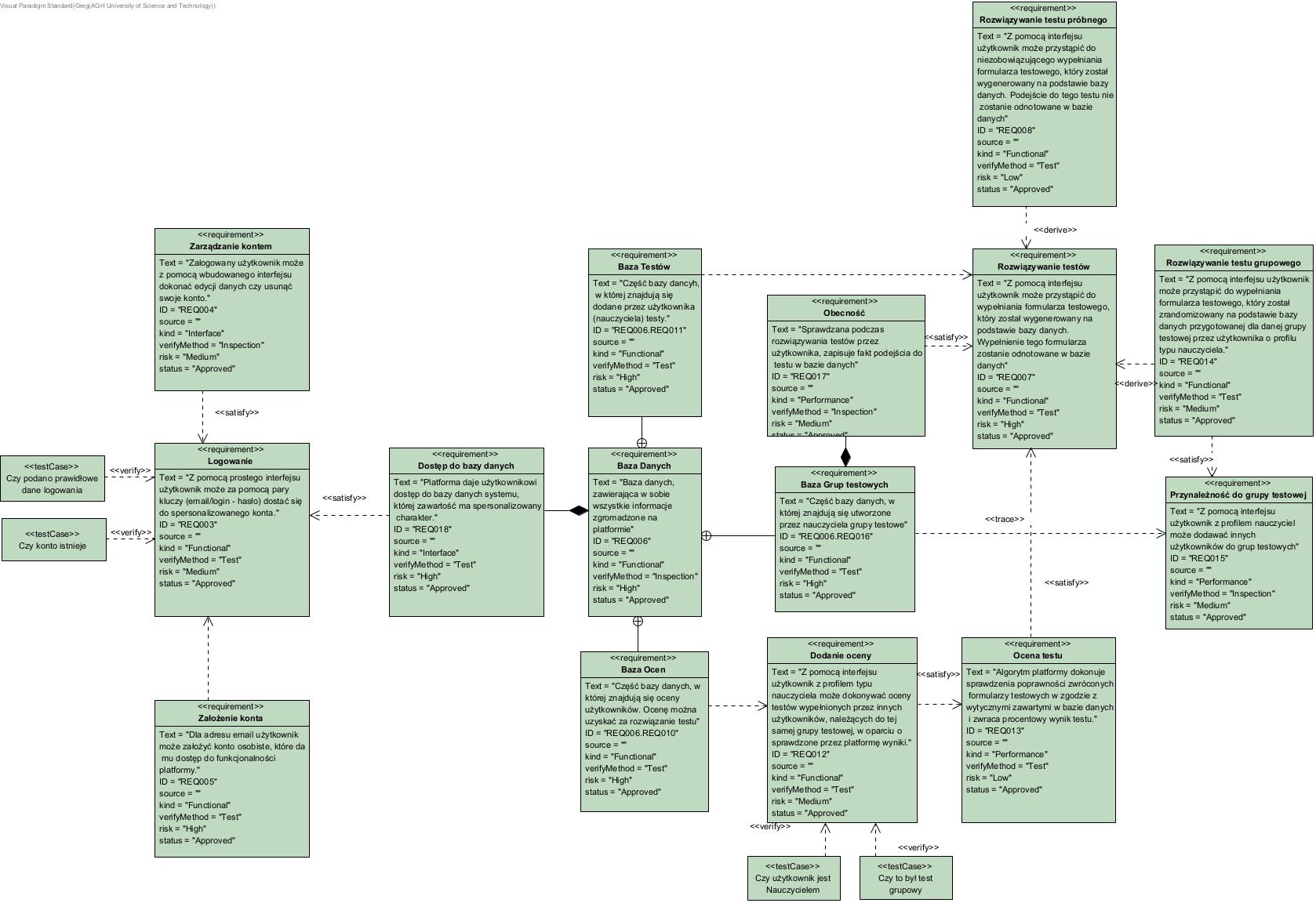
**Use Case – Edycja pytań egzaminacyjnych**

****

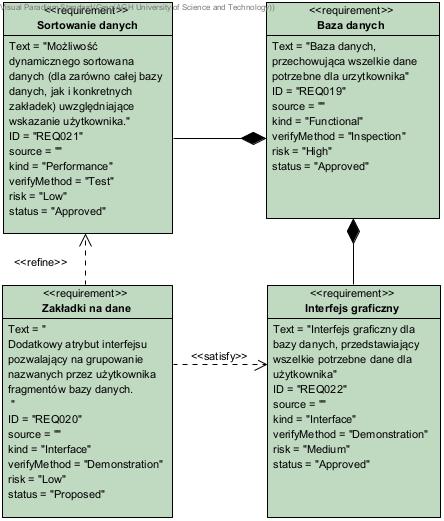
****

6. Diagram przypadku użycia wraz z pełnym opisem

## Diagram wymagań funkcjonalnych



## Diagram wymagań niefunkcjonalnych

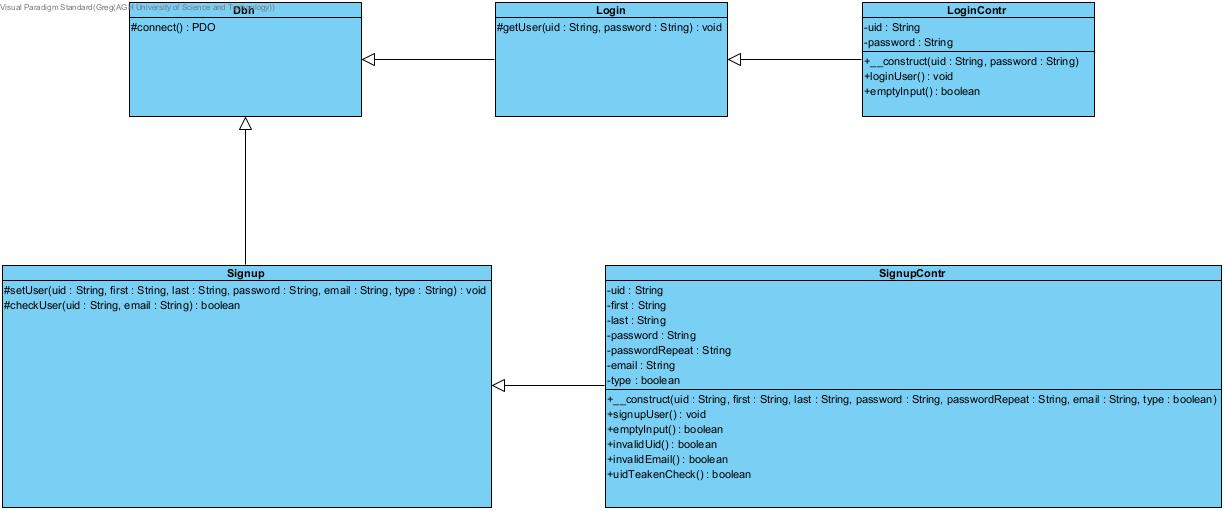


## Zastosowane framework-i, środowiska, technologie

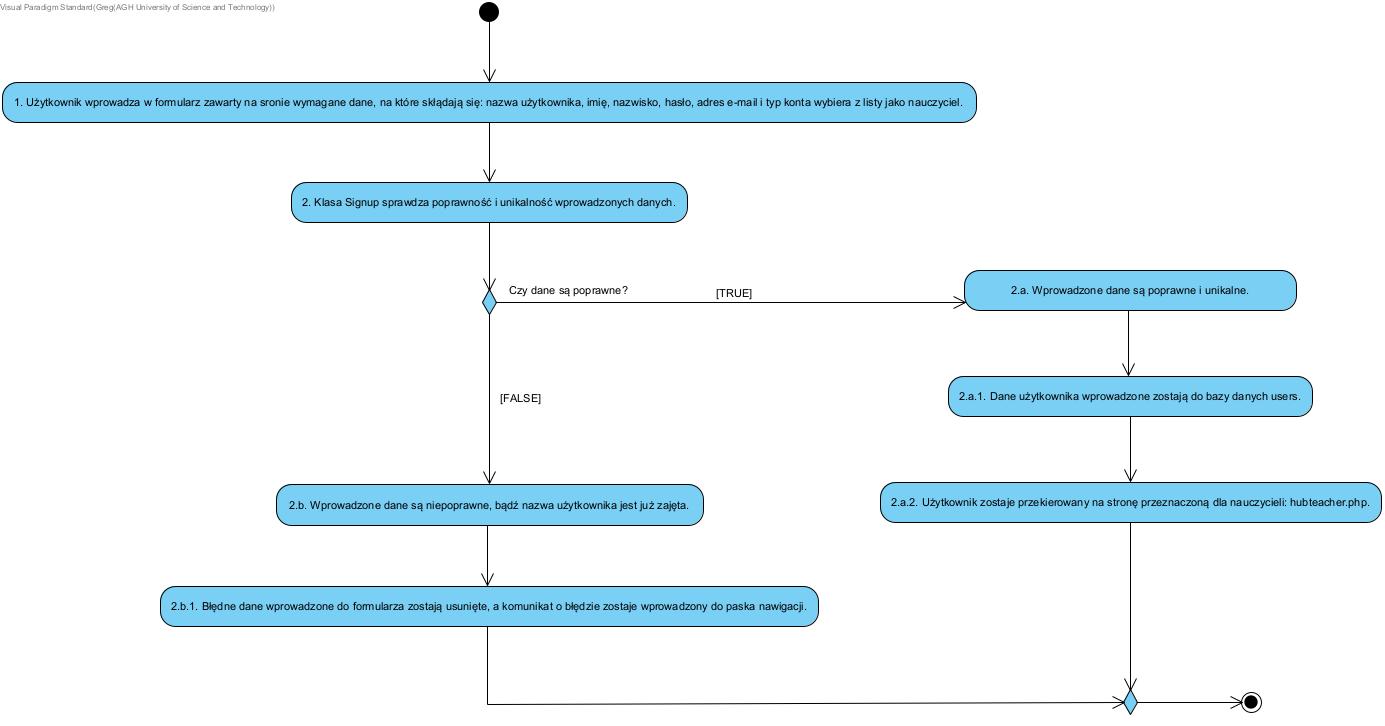
W celu postawienia witryny używamy oprogramowania XAMPP, czyli cross-platformowego serwera opracowanego przez Apache Friends, w skład, którego wchodzą wykorzystywane Apache HTTP Server czy PHP. Sam kod strony pisany był w języku OOP PHP, oczywiście z włączeniem HTML5 i CSS.

Kod pisany był w Visual Studio Code i Notepad++.

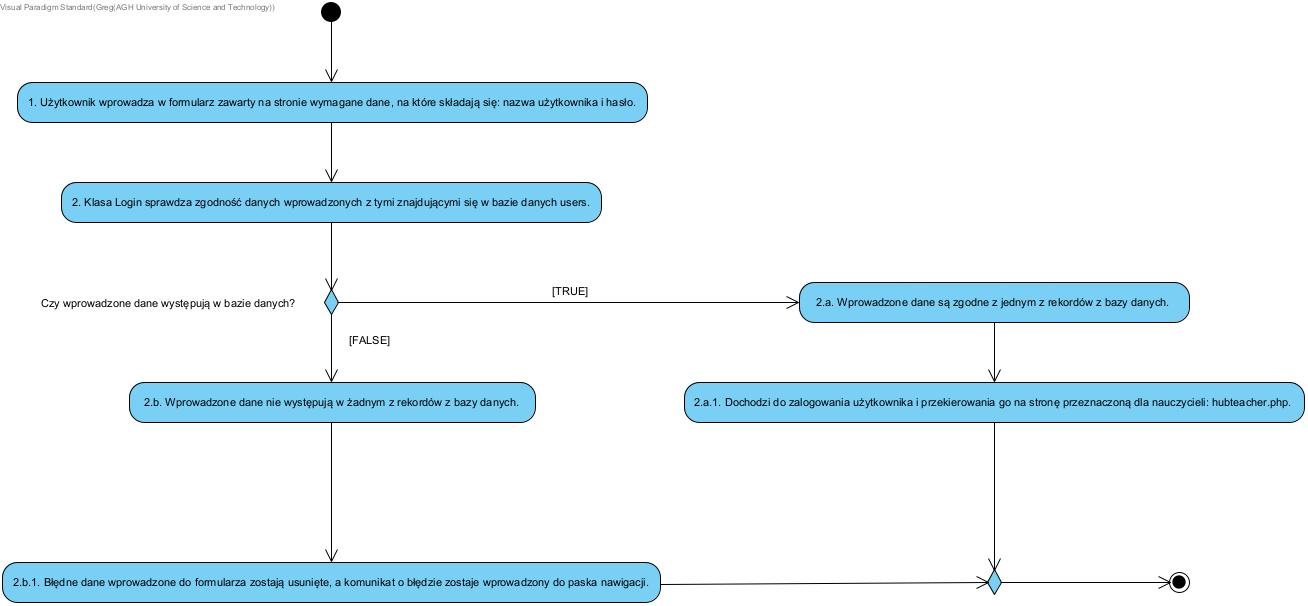
## Diagram klas



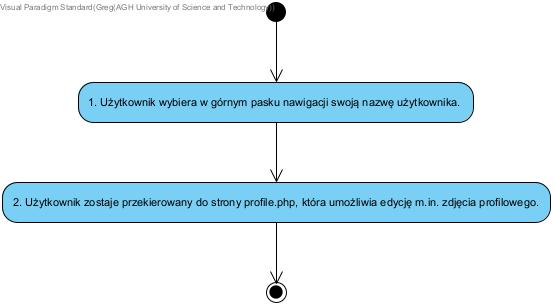
## Diagramy aktywności

Rejestracja

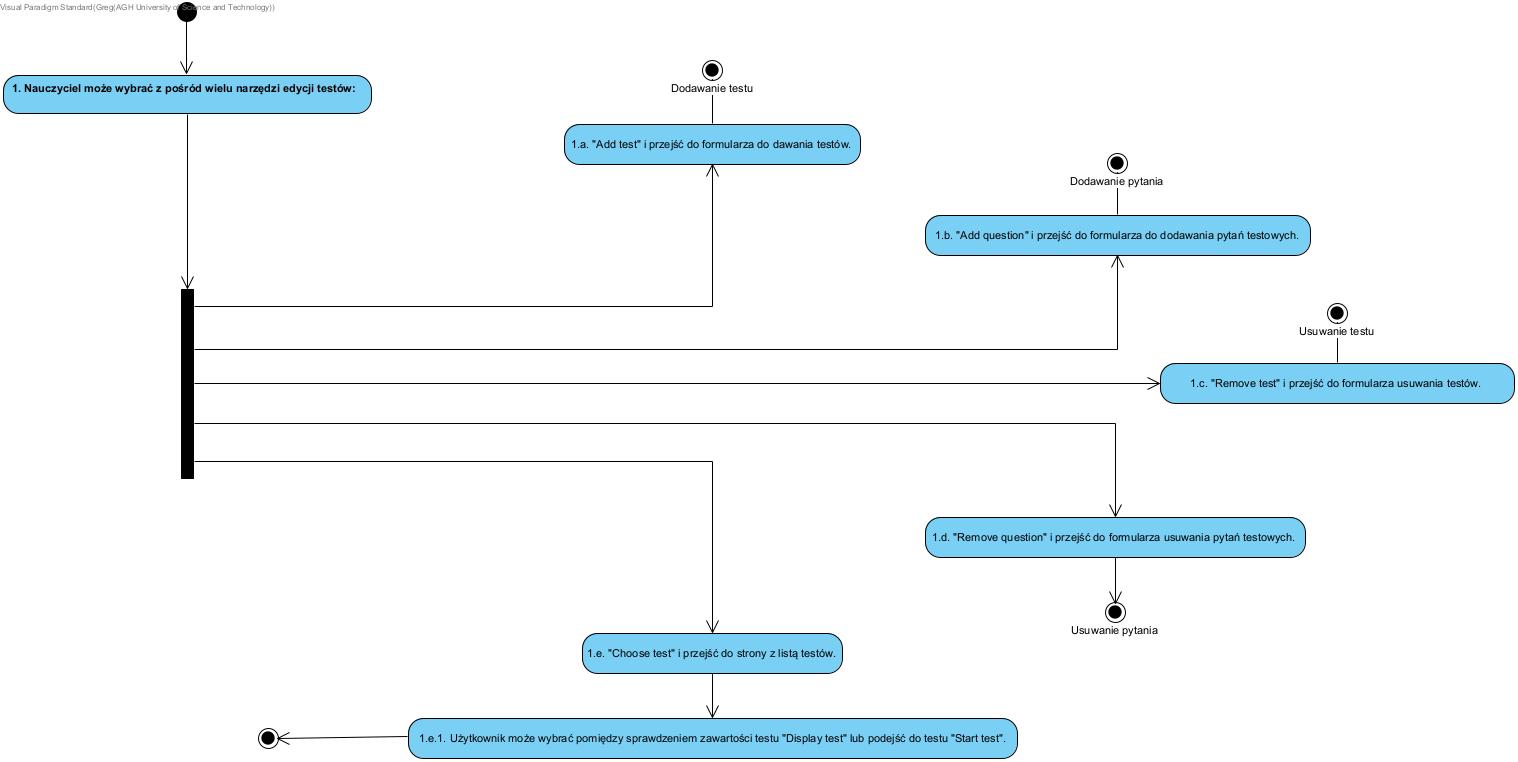
Logowanie



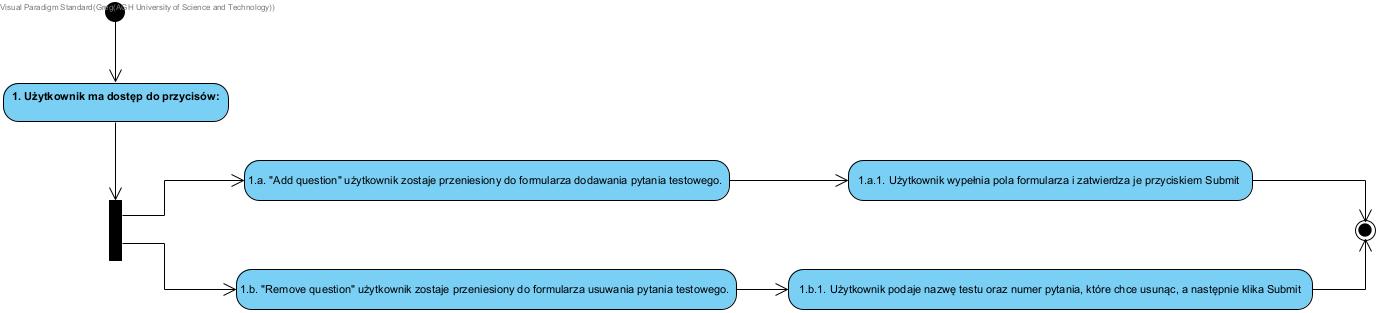
Zarządzanie kontem



Zarządzanie testem

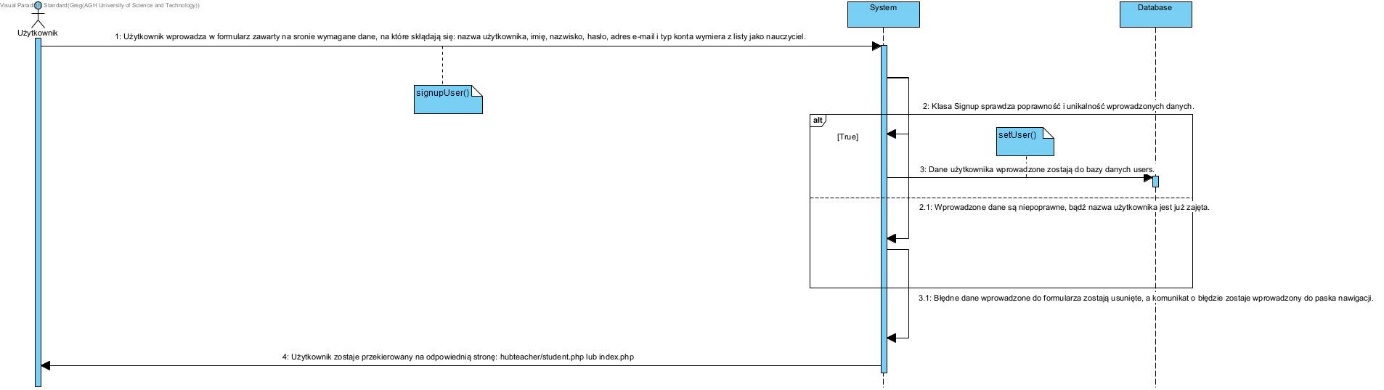


Edycja pytań egzaminacyjnych

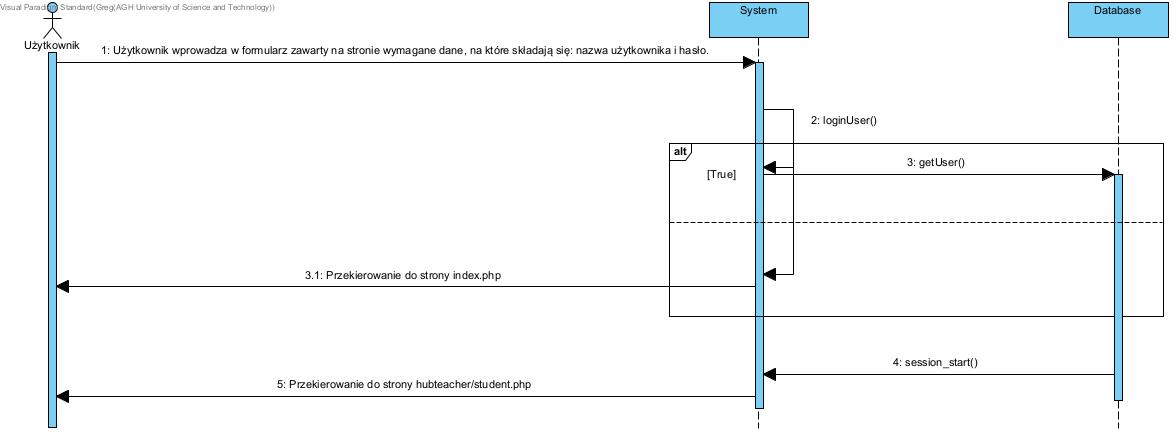


## Diagramy sekwencji

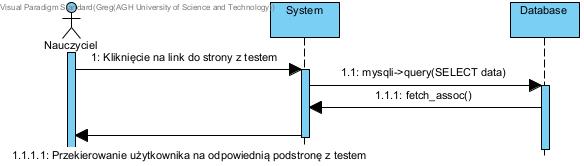
Rejestracja



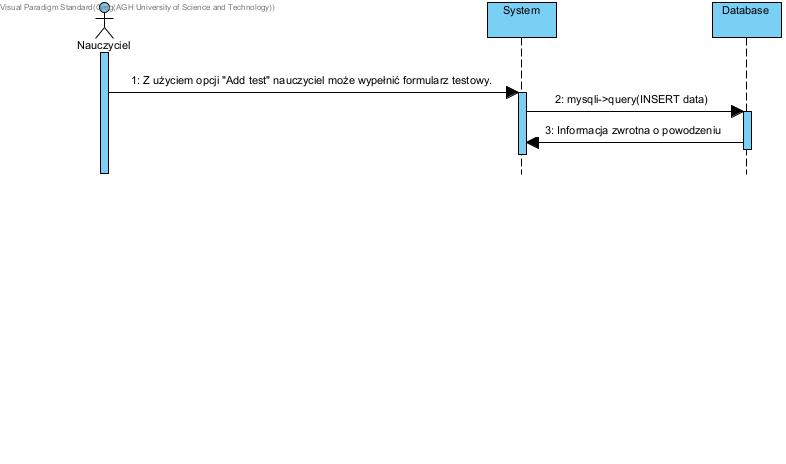
Logowanie

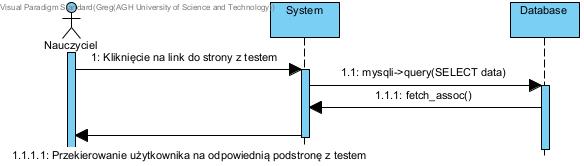


Zarządzanie testem

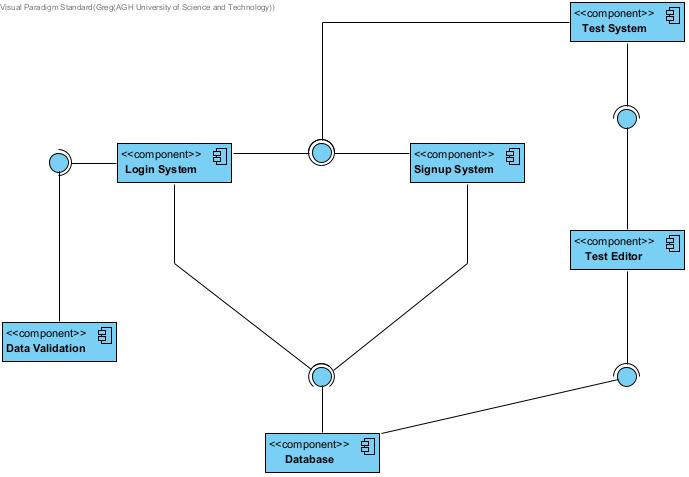
Start test 

Add test

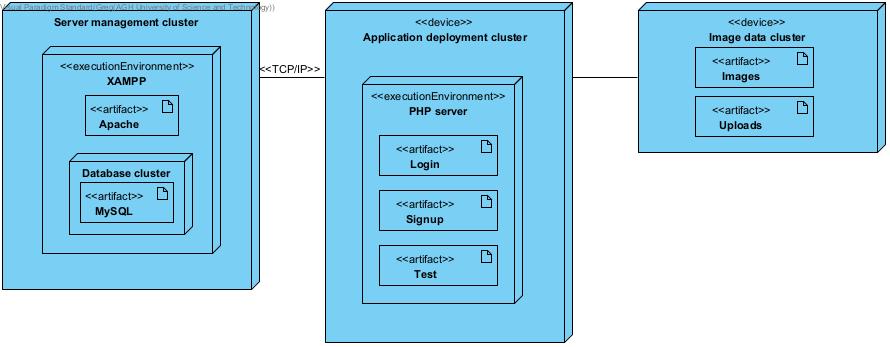


Display test

## Diagram komponentów



## Diagram wdrożenia



## Przykłady współpracy zespołu z wykorzystaniem wybranych narzędzi

Obraz zawierający tekst

Opis wygenerowany automatycznie

Testy jednostkowe – min. 2 przykłady

## Przykład refaktoryzacji

Obraz zawierający tekst, zrzut ekranu, monitor, telefon

Opis wygenerowany automatycznie

Obraz zawierający tekst, monitor, zrzut ekranu, ekran

Opis wygenerowany automatycznie

Obraz zawierający tekst, pomarańczowy, ekran, ciemny

Opis wygenerowany automatycznieObraz zawierający tekst

Opis wygenerowany automatycznie

18. Przykład wzorca projektowego, który mógłby zostać wykorzystany podczas tworzenia projektu. Należy w tym punkcie wyjaśnić przyczyny wyboru danego wzorca/wzorców. 19. Instrukcja użytkowania programu - Pomoc. 20. Podsumowanie (Opis prac, problemy napotkane podczas realizacji projektu, cechy oraz funkcjonalności wyróżniające opracowane rozwiązanie)