

Dokumentacja projektu zaliczeniowego

Przedmiot: Inżynieria oprogramowania

Temat: Pomoc dla studentów w przedmiocie metody numeryczne na 3 semestrze na uniwersytecie ZUT kierunek informatyka.

Autorzy: **Igor Stodolny, Marcin Włodarczyk**

Grupa: I1-221B

Kierunek: informatyka

Rok akademicki: 2020/2021

Poziom i semestr: I/4

Tryb studiów: stacjonarne

1 Spis treści

Dokumentacja projektu zaliczeniowego.....	1
2 Odnośniki do innych źródeł.....	5
3 Słownik pojęć.....	7
4 Wprowadzenie	8
4.1 Cel dokumentacji	8
4.2 Przeznaczenie dokumentacji.....	8
4.3 Opis organizacji lub analiza rynku.....	8
5 Specyfikacja wymagań.....	9
5.1 Charakterystyka ogólna	9
5.2 Wymagania funkcjonalne	11
5.3 Wymagania нефunkcjonalne	41
6 Zarządzanie projektem	42
6.1 Zasoby ludzkie	42
6.2 Harmonogram prac.....	43
6.3 Etapy/kamienie milowe projektu	44
7 Zarządzanie ryzykiem	45
7.1 Lista czynników ryzyka.....	45
7.2 Ocena ryzyka	46
7.3 Plan reakcji na ryzyko	47
8 Zarządzanie jakością	49
8.1 Scenariusze i przypadki testowe	50
9 Projekt techniczny	60
9.1 Opis architektury systemu	60
9.2 Technologie implementacji systemu.....	61
9.3 Diagramy UML.....	63
9.4 Projekt bazy danych	73
9.5 Projekt interfejsu użytkownika	76
9.6 Procedura wdrożenia	79
10 Dokumentacja dla użytkownika	80
11 Podsumowanie.....	82
11.1 Szczegółowe nakłady projektowe członków zespołu	82
12 Inne informacje.....	84

2 Odnośniki do innych źródeł

- Zarządzania projektem – sugerowane JazzHub(google docs)
- Wersjonowanie kodu – <https://github.com/wm46691/IO-project>
- System obsługi defektów – np. Bitbucket, JazzHub
- Źródło informacji do implementacji algorytmów :
<https://www.impan.pl/~szczep/AMM1/Kincaid.pdf>

3 Słownik pojęć

Tabela lub lista z pojęciami, które wymagają wyjaśnienia, wraz z tymi wyjaśnieniami – w szczególności synonimy różnych pojęć używanych w dokumentacji.

- **Student** - grupa użytkowników uczących się
- **Student premium** - grupa użytkowników uczących się, płacących za dostęp do dodatkowych materiałów
- **Nauczyciel** - grupa użytkowników nauczających
- **Wykład** - nagrany materiał wideo wyjaśniający dane zagadnienie analizy numerycznej
- **Materiały** - wszelkiego rodzaju informacje wyjaśniające dane zagadnienia w formie tekstowej i graficznej (ale grafiki statycznej)
- **Lekcja** - połączenie wideogłosowe pomiędzy nauczycielem a jednym lub więcej studentów
- **Grupa** - grupa studentów utworzona przez nauczyciela utworzona w celu przeprowadzania wspólnej lekcji i łatwiejszego udostępniania materiałów
- **Znajomi** - studenci mogą być znajomymi, wówczas mogą prosić się wzajemnie o pomoc

4 Wprowadzenie

4.1 Cel dokumentacji

- Dokumentacja zawiera różnego rodzaju modele, scenariusze ryzyka, diagramy klas, diagramy czynności, diagramy sekwencji, inne diagramy, słownik pojęć, odnośniki, projekty baz danych, projekty interfejsu, harmonogram prac.
- Ułatwienie implementacji aplikacji.
- Informacje na temat działania programu oraz informacje dotyczące samego programu.
- Informacje dotyczące planu reakcji na ryzyko.

4.2 Przeznaczenie dokumentacji

Dokumentacja przeznaczona jest dla zespołu tworzącego aplikację oraz nowych członków zespołu.

4.3 Opis organizacji lub analiza rynku

Nasz produkt jest przeznaczony na masowy rynek.

W Polsce jest wiele uczelni, które prowadzą kierunki, które posiadają przedmiot związany z rozwiązywaniem problemów matematycznych. Taki system będzie przydatny dla każdej grupy studenckiej, która posiada na swoim kierunku przedmiot, który dotyczy rozwiązywania problemów matematycznych(przedmiot metody numeryczne). Daje to szacunkowo kilkunastu-kilkudziesięciu tysięcy studentów rocznie, którzy są potencjalnymi użytkownikami naszej aplikacji. Co roku będą pojawiać się nowi użytkownicy, w mniej więcej podobnej liczbie. Większości studentom nie wystarczają same wykłady do zrozumienia zagadnienia. Potrzebują dodatkowych narzędzi oraz innych źródeł nauki jaka nasza aplikacja dostarcza. Wraz z rozwojem aplikacji nie tylko polscy studenci będą posługiwali się aplikacją ale również możliwość będą mieli studenci za granicą.

5 Specyfikacja wymagań

5.1 Charakterystyka ogólna

5.1.1 Definicja produktu

Program ułatwiający naukę przedmiotu metody numeryczne za pomocą różnego rodzaju narzędzi oraz materiałów dostępnych w aplikacji.

5.1.2 Podstawowe założenia

System ten będzie służył ułatwieniu studentom nauki przedmiotu metody numeryczne za pomocą różnych dostępnych narzędzi oraz materiałów wcześniej przygotowanych. Każda grupa użytkowników będzie posiadała różny zakres uprawnień. Grupa Studenti będzie posiadała dostęp do podstawowych narzędzi. Grupa Studenti premium będą posiadali dostęp do wszystkich narzędzi do nauki. Grupa Nauczycieli będzie tak samo jak grupa Studenti premium posiadać dostęp narzędzi premium oraz możliwość tworzenia prywatnych zajęć lub grupowych zajęć.

5.1.3 Cel biznesowy

Cele długoterminowe:

- Zwiększenie procentu zdawalności z przedmiotu.
- Ułatwienie lepszego zrozumienia treści dotyczącego działu.
- Polepszenie u użytkowników aparatu matematycznego.
- Poznanie nowych metod ułatwiających tworzenie skomplikowanych algorytmów w programowaniu.

Cele krótkoterminowe:

- Rozszerzenie aplikacji o wielojęzyczność.
- Powiększenie aplikacji o dodatkowe działy matematyczne
- Utworzenie kursów pozwalających uzyskać certyfikaty.

5.1.4 Użytkownicy

Nauczyciele

Studenti premium

Studenti

Niezalogowani

5.1.5 Korzyści z systemu

1. Nauczyciele

- Udostępniania materiałów studentom
- Generowanie testów sprawdzających
- Pomoc studentom
- Prowadzenie lekcji

2. Studenti premium

- Dostęp do materiałów i wykładów

- b. Możliwość samodzielnej nauki z wykorzystaniem wizualizacji zadań i ich generowania
- c. Wzajemna pomoc studentów
- d. Uczestniczenie w e-lekcjach
- e. Uzyskiwanie pomocy od nauczycieli

3. Studenci

- a. Dostęp do materiałów
- b. Możliwość samodzielnej nauki z wykorzystaniem wizualizacji zadań i ich generowania
- c. Wzajemna pomoc studentów
- d. Wykupienie dostępu premium

4. Niezalogowani

- a. Możliwość rejestracji
- b. Możliwość logowania

5.1.6 Ograniczenia projektowe i wdrożeniowe

Użytkownicy mają prawo do usunięcia i wglądu w swoje dane. Użytkownicy mogą wyrazić zgodę na przetwarzanie danych. W pierwszych fazach funkcjonowania aplikacja będzie jedynie obejmowała jeden dział matematyczny ze względu na ilość pamięci potrzebnej do przechowywania materiałów dydaktycznych i zadań. Na początku funkcjonowania aplikacji brak możliwości zatrudnienia wystarczającej liczby nauczycieli ze względu na ograniczony budżet. Serwer, na którym będziemy wdrażać aplikację, musi obsługiwać Python w wersji co najmniej 3.7. Optymalne funkcjonowanie aplikacji jest zależne od współpracy z uczelniami, na których znajdują się przedmiot, który dotyczy materiału obsługiwanego przez aplikację. Aplikacja nie będzie dostępna na urządzeniach mobilnych ze względu na duży interfejs aplikacji. Ekran mobilnego urządzenia nie będzie w stanie pokryć całego interfejsu co by powodowało optymalne używanie aplikacji.

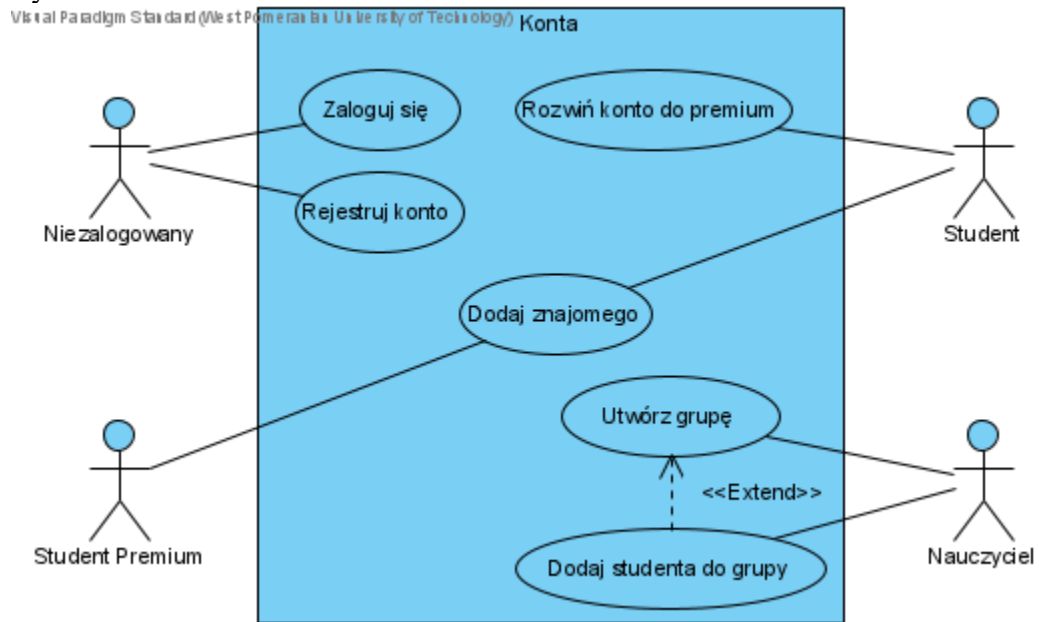
5.2 Wymagania funkcjonalne

5.2.1 Lista wymagań

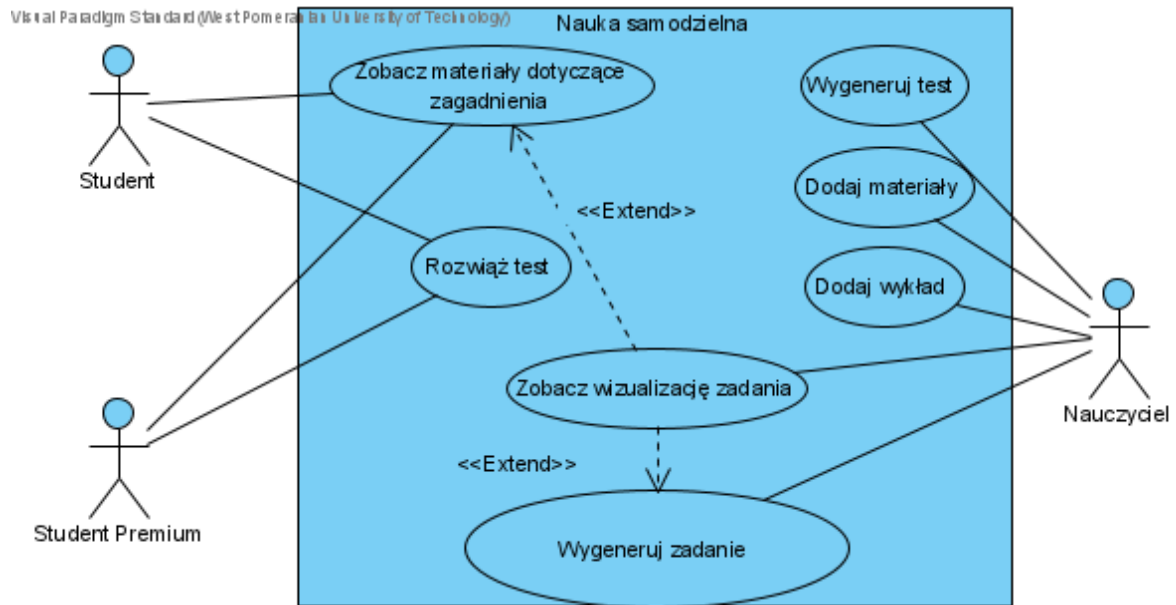
1. Przypadki użycia dla grupy student:
 - 1.1. Rozwiń konto do premium
 - 1.2. Wygeneruj zadanie
 - 1.3. Zobacz wizualizację zadania
 - 1.4. Rozwiąż test
 - 1.5. Zobacz materiały dotyczące zagadnienia.
 - 1.6. Poproś o pomoc innego studenta
 - 1.7. Pomóż innemu studentowi
 - 1.8. Dodaj znajomego
2. Przypadki użycia dla grupy student premium:
 - 2.1. Wygeneruj zadanie
 - 2.2. Zobacz wizualizację zadania
 - 2.3. Rozwiąż test
 - 2.4. Zobacz wykład
 - 2.5. Zobacz materiały dotyczące zagadnienia.
 - 2.6. Poproś o pomoc innego studenta
 - 2.7. Poproś o pomoc nauczyciela
 - 2.8. Pomóż innemu studentowi
 - 2.9. Dołącz do lekcji
 - 2.10. Dodaj znajomego
3. Przypadki użycia dla grupy nauczyciel:
 - 3.1. Wygeneruj zadanie
 - 3.2. Zobacz wizualizację zadania
 - 3.3. Wygeneruj test
 - 3.4. Prowadź lekcję
 - 3.5. Dodaj wykład
 - 3.6. Dodaj materiały
 - 3.7. Pomóż studentowi
 - 3.8. Utwórz grupę
 - 3.9. Dodaj studenta do grupy
 - 3.10. Usuń student z grupy
 - 3.11. Usuń grupę
4. Przypadki użycia dla grupy niezalogowany:
 - 4.1. Rejestruj konto
 - 4.2. Zaloguj się

5.2.2 Diagramy przypadków użycia

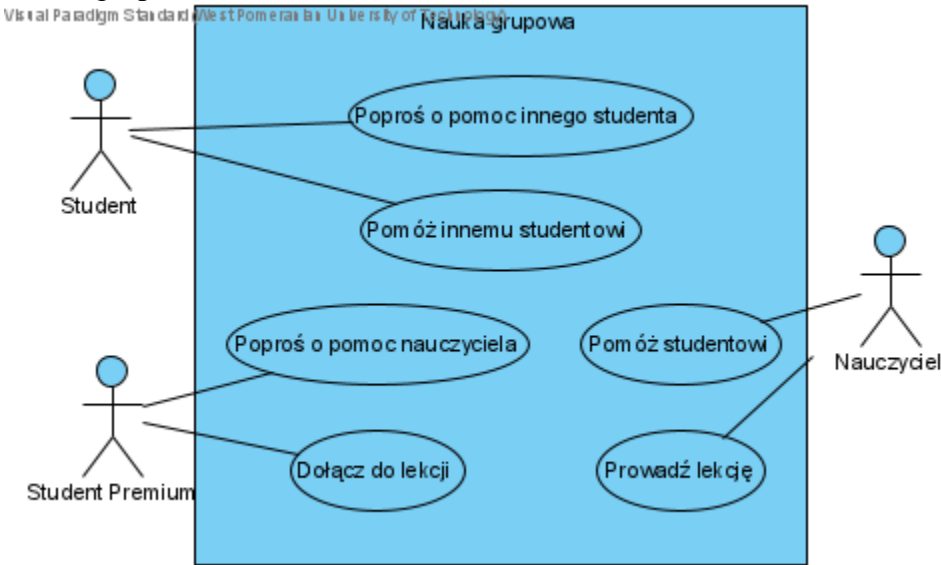
Konta użytkowników:



Nauka samodzielna:



Nauka grupowa:



5.2.3 Szczegółowy opis wymagań

ID przypadku użycia	1.1	
Nazwa	Rozwiń konto do premium	
Uzasadnienie biznesowe	3.d - Wykupienie dostępu premium	
Użytkownicy	Student, Student Premium, Nauczyciel	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik uruchamia okno “Przejdź do premium” 2. System wyświetla formularz 3. Użytkownik wypełnia formularz 4. System przekazuje żądanie płatności do zewnętrznego operatora 5. System przyjmuje potwierdzenie opłaty od zewnętrznego operatora 6. System przenosi użytkownika do grupy Student Premium
	Efekty:	Użytkownik jest w grupie użytkowników Student Premium
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Dane podane przez użytkownika powinny być zaszyfrowane • Dane podane przez użytkownika powinny być usunięte po potwierdzeniu płatności • Po otrzymaniu potwierdzenia płatności użytkownik powinien być przeniesiony do nowej grupy w ciągu 5 sekund.
	Częstotliwość	1
	Istotność	4
	Scenariusz alternatywny 1 - brak potwierdzenia	1.-4. jak w scenariuszu głównym 5. System przyjmuje odmowę płatności od zewnętrznego operatora 6. System wyświetla komunikat o błędzie

	opłaty	
--	--------	--

ID przypadku użycia	1.2, 2.1	
Nazwa	Wygeneruj zadanie	
Uzasadnienie biznesowe	2.b, 3.b - Możliwość samodzielnej nauki z wykorzystaniem wizualizacji zadań i ich generowania	
Użytkownicy	Student, Student Premium	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany
	Przebieg działań:	<p>Użytkownik wybiera opcję Generuj Zadanie</p> <p>Użytkownik wybiera konkretny temat lub szerszy zakres materiału którego ma dotyczyć zadanie</p> <p>System generuje zadanie z zadanego tematu</p> <p>System wyświetla zadanie</p> <p>Użytkownik potwierdza rozwiązanie zadania</p> <p>System wyświetla odpowiedź</p>
	Efekty:	Użytkownik może rozwiązać zadanie, porównać odpowiedź, zobaczyć wizualizację rozwiązania zadania
	Wymagania нефункционалне	<ul style="list-style-type: none"> Wygenerowane zadanie musi być możliwe do rozwiązania daną metodą Zadanie powinno być wygenerowane w czasie maks. 2 sekund.
	Częstotliwość	4
	Istotność	4
	Scenariusz alternatywny I - wizualizacja	1.-6. jak w scenariuszu głównym 7. Użytkownik wybiera opcję wizualizację rozwiązania -> przypadek użycia 1.3,2.2,3.2

	Scenariusz alternatywny II - chęć skorzystania z pomocy nauczyciela	1.-6. - jak w scenariuszu głównym Użytkownik prosi o pomoc nauczyciela (przypadek 2.7, użytkownik musi być w grupie Student Premium)
	Scenariusz alternatywny II - chęć skorzystania z pomocy innego studenta	1.-6. - jak w scenariuszu głównym 7. Użytkownik prosi o pomoc innego studenta (przypadek 1.6, 2.6)

ID przypadku użycia	1.3, 2.2, 3.2	
Nazwa	Zobacz wizualizację	
Uzasadnienie biznesowe	2.b, 3.b - Możliwość samodzielnej nauki z wykorzystaniem wizualizacji zadań i ich generowania	
Użytkownicy	Student, Student premium, Nauczyciel	
Scenariusz główny I	Warunek początkowy:	Użytkownik poprawnie zalogowany, otwarte okno materiałów dotyczące konkretnej metody
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik wybiera opcję “Zobacz wizualizację” 2. Użytkownik podaje parametry 3. System wyświetla wizualizację wykonania metody dla określonych parametrów
	Efekty:	Wyświetla się wizualizacja, użytkownik może zobaczyć jak działa dany algorytm numeryczny krok po kroku
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Wizualizacja powinna być możliwa dla dowolnych wybranych parametrów • Czas tworzenia wizualizacji powinien wynosić maks. 5 sekund • Użytkownik musi mieć możliwość zwalniania, cofania i przechodzenia przez wizualizację krok po kroku
	Częstotliwość	2
	Istotność	3
Scenariusz główny II	Warunek początkowy:	Użytkownik poprawnie zalogowany, rozwiązujący zadanie lub wygenerowany test
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik wybiera opcję “Zobacz wizualizację” 2. System generuje wizualizację wykonania

		danej metody dla parametrów podanych w zadaniu
	Efekty:	Wyświetla się wizualizacja, użytkownik może zobaczyć jak działa dany algorytm numeryczny krok po kroku
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Czas tworzenia wizualizacji powinien wynosić maks. 5 sekund • Użytkownik musi mieć możliwość zwalniania, cofania i przechodzenia przez wizualizację krok po kroku
	Częstotliwość	3
	Istotność	3

ID przypadku użycia	1.4, 2.3	
Nazwa	Rozwiąż test	
Uzasadnienie biznesowe	2.b, 3.b - Możliwość samodzielnej nauki z wykorzystaniem wizualizacji zadań i ich generowania	
Użytkownicy	Student, Student Premium	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany, test jest dostępny dla studenta
	Przebieg działań:	<ol style="list-style-type: none"> 1. Student przechodzi do okna testów 2. System wyświetla listę kategorii 3. Student wybiera interesującą go kategorię 4. System wyświetla mu listę dostępnych testów 5. Student wybiera test i przystępuje do rozwiązywania 6. System zapisuje wynik testu
	Efekty:	Student może rozwiązać test i poznać swoje umiejętności, nauczyciel może sprawdzać swoich uczniów
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Wyniki testu powinny być dostępne tylko dla ucznia i ewentualnie nauczyciela • W razie utraty połączenia aktualny wynik testu powinien być przechowywany w pamięci lokalnej
	Częstotliwość	2
	Istotność	3

ID przypadku użycia	1.5, 2.5	
Nazwa	Zobacz materiały dotyczące zagadnienia	
Uzasadnienie biznesowe	2.b, 3.b - Możliwość samodzielnej nauki z wykorzystaniem wizualizacji zadań i ich generowania 2.a, 3.a -Dostęp do materiałów	
Użytkownicy	Student, Student Premium, Nauczyciel	
Scenariusz główny	Warunek początkowy:	Użytkownik jest poprawnie zalogowany, w systemie istnieją materiały odnoszące się do podanego zagadnienia.
	Przebieg działań:	<ol style="list-style-type: none"> 1. Student przechodzi do okna testów 2. System wyświetla listę kategorii 3. Student wybiera interesującą go kategorię 4. System wyświetla mu dostępne materiały
	Efekty:	Użytkownik może zapoznać się z informacjami dotyczącymi danego zagadnienia
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Wyświetlenie materiałów musi nastąpić w ciągu 2 sekund
	Częstotliwość	3
	Istotność	3

ID przypadku użycia	1.6, 2.6	
Nazwa	Poproś o pomoc innego studenta	
Uzasadnienie biznesowe	2.c, 3.c - Wzajemna pomoc studentów	
Użytkownicy	Student, Student Premium	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany, student rozwiązuje zadanie
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik wybiera opcję zgłoszenia chęci otrzymania pomocy od innego użytkownika 2. System przyjmuje zgłoszenie 3. System przekazuje zgłoszenie do puli zgłoszeń dostępnych dla innych studentów
	Efekty:	Zgłoszenie studenta jest dostępne do przyjęcia przez innych studentów
	Wymagania niefunkcjonalne	-
	Częstotliwość	3
	Istotność	3
	Scenariusz alternatywny 1 – prośba o pomoc od konkretnego studenta (znajomego)	<ol style="list-style-type: none"> 1. Użytkownik wybiera opcję zgłoszenia chęci otrzymania pomocy od innego użytkownika 2. System przyjmuje zgłoszenie 3. System przekazuje zgłoszenie do konkretnego użytkownika i wyświetla mu powiadomienie

ID przypadku użycia	1.8, 2.10	
Nazwa	Dodaj znajomego	
Uzasadnienie biznesowe	2.c, 3.c - Wzajemna pomoc studentów	
Użytkownicy	Student, Student Premium	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany.
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik wyszukuje użytkownika po jego loginie lub wybiera użytkownika z którym jest w grupie. 2. Użytkownik wybiera opcję dodaj znajomego 3. System wysyła drugiemu użytkownikowi komunikat o dodaniu do znajomych i czeka na jego potwierdzenie 4. System wysyła pierwszemu użytkownikowi informację zwrotną
	Efekty:	Użytkownik może zostać dodany do znajomych
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Lista użytkowników powinna być jawna tylko dla danego użytkownika • Wyświetlenie powiadomienia i samo dodanie do znajomych nie powinno trwać dłużej niż 2 sekundy
	Częstotliwość	2
	Istotność	2

ID przypadku użycia	1.7, 2.8	
Nazwa	Pomóż innemu studentowi	
Uzasadnienie biznesowe	2.c, 3.c - Wzajemna pomoc studentów	
Użytkownicy	Student, Student Premium	
Scenariusz główny I	Warunek początkowy:	Student jest poprawnie zalogowany, w systemie jest zgłoszenie chęci otrzymania pomocy od studenta
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik otwiera okno ze zgłoszeniami chęci otrzymania pomocy 2. System wyświetla listę obecnych zgłoszeń 3. Nauczyciel wybiera zgłoszenie 4. System łączy studenta i nauczyciela rozmową tekstową, która może być zamieniona w rozmowę głosową.
	Efekty:	Student ma możliwość udzielić pomocy innemu studentowi.
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Żadna niepożądana osoba nie może dołączyć do rozmowy • Przesyłane rozwiązania zostają zachowane w bazie danych do późniejszego wglądu
	Częstotliwość	3
	Istotność	3
	Scenariusz alternatywny - nieudana próba zestawienia połączenia	<ol style="list-style-type: none"> 1.-3. jak w scenariuszu głównym 4. Nieudana próba zestawienia połączenia pomiędzy studentami 5. System wyświetla komunikat o błędzie studentom 6. System przenosi studenta pomagającego

		ponownie do okna zgłoszeń chęci otrzymania pomocy
Scenariusz główny II	Warunek początkowy:	Student jest poprawnie zalogowany, w systemie jest zgłoszenie chęci otrzymania pomocy od studenta
	Przebieg działań:	5. Użytkownik otwiera okno ze zgłoszeniami chęci otrzymania pomocy 6. System wyświetla listę obecnych zgłoszeń 7. Nauczyciel wybiera zgłoszenie 8. System łączy studenta i nauczyciela rozmową tekstową, która może być zamieniona w rozmowę głosową.
	Efekty:	Student ma możliwość udzielić pomocy innemu studentowi.
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Żadna niepożądana osoba nie może dołączyć do rozmowy • Przesyłane rozwiązania zostają zachowane w bazie danych do późniejszego wglądu
	Częstotliwość	3
	Istotność	3
	Scenariusz alternatywny - nieudana próba zestawienia połączenia	1.-3. jak w scenariuszu głównym 4. Nieudana próba zestawienia połączenia pomiędzy studentami 5. System wyświetla komunikat o błędzie studentom 6. System przenosi studenta pomagającego ponownie do okna zgłoszeń chęci otrzymania pomocy

ID przypadku użycia	2.4	
Nazwa	Zobacz wykład	
Uzasadnienie biznesowe	2.a - Dostęp do materiałów i wykładów	
Użytkownicy	Student Premium	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany, istnieją udostępnione wykłady
	Przebieg działań:	Użytkownik przechodzi w okno wykłady System wyświetla listę kategorii Użytkownik wybiera kategorię System wyświetla listę wykładów w danej kategorii Użytkownik wybiera wykład System wyświetla wykład użytkownikowi
	Efekty:	Użytkownik może obejrzeć nagrany wcześniej wykład.
	Wymagania нефункционалне	<ul style="list-style-type: none"> System musi być w stanie wyświetlić fiilm w rozdzielczości przynajmniej 720p Materiał musi być przesyłany w postaci skompresowanej, tak aby nie było lagów dla prędkości połączenia powyżej 20Mb/s
	Częstotliwość	3
	Istotność	3

ID przypadku użycia	2.7	
Nazwa	Poproś o pomoc nauczyciela	
Uzasadnienie biznesowe	2.e - Uzyskiwanie pomocy od nauczycieli	
Użytkownicy	Student premium	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany, student rozwiązuje zadanie
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik wybiera opcję zgłoszenia chęci otrzymania pomocy od innego użytkownika 2. System przyjmuje zgłoszenie 3. System przekazuje zgłoszenie do puli zgłoszeń dostępnych dla nauczycieli
	Efekty:	Zgłoszenie studenta jest dostępne do przyjęcia przez nauczycieli
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Zgłoszenie musi być przetworzone i wysłane do nauczyciela w ciągu 5 sekund
	Częstotliwość	3
	Istotność	3

ID przypadku użycia	2.9	
Nazwa	Dołącz do lekcji	
Uzasadnienie biznesowe	2.d - Uczestniczenie w e-lekcjach	
Użytkownicy	Student Premium	
Scenariusz główny	Warunek początkowy:	Student jest poprawnie zalogowany, nauczyciel rozpoczął lekcję
	Przebieg działań:	<ol style="list-style-type: none"> 1. Student przechodzi do okna lekcji 2. System wyświetla dostępne lekcje 3. Student wybiera konkretną lekcję 4. System ustanawia połączenie pomiędzy studentem a nauczycielem i innymi studentami obecnymi na lekcji
	Efekty:	Student może brać udział w lekcji online.
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Tylko studenci z danej grupy mogą brać udział w danej lekcji • Połączenie musi być stabilne
	Częstotliwość	2
	Istotność	4

ID przypadku użycia	3.3	
Nazwa	Wygeneruj test	
Uzasadnienie biznesowe	1.b - Generowanie testów sprawdzających	
Użytkownicy	Nauczyciel	
Scenariusz główny	Warunek początkowy:	Nauczyciel jest poprawnie zalogowany
	Przebieg działań:	<ol style="list-style-type: none"> 1. Nauczyciel przechodzi w okno wygeneruj test 2. Nauczyciel wybiera parametry testu: zakres materiału, ilość pytań, czas na rozwiązanie. 3. System generuje zadania. 4. Nauczyciel może edytować wygenerowane zadania 5. Nauczyciel zatwierdza test i ustawia czas przesłania studentom 6. System przetwarza test i wysyła go do studentów w odpowiednim czasie 7. Po wykonaniu testu przez studentów system zwraca wyniki.
	Efekty:	Zostaje wygenerowany test, który może być rozwiązany przez studentów. Może być to forma ocenianego sprawdzianu lub po prostu sprawdzenia umiejętności. Nauczyciel otrzymuje informację zwrotną.
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Wygenerowane zadania muszą być możliwe do rozwiązania. • Test musi być przesłany studentom w ustawionym czasie - maksymalnie 10 s opóźnienia • Informacja zwrotna może być jawna tylko dla nauczyciela i studenta, który udydził danych odpowiedzi

	Częstotliwość	3
	Istotność	4

ID przypadku użycia	3.4	
Nazwa	Prowadź lekcję	
Uzasadnienie biznesowe	1.d - Prowadzenie lekcji	
Użytkownicy	Nauczyciel	
Scenariusz główny	Warunek początkowy:	Nauczyciel jest poprawnie zalogowany, utworzona jest grupa studentów
	Przebieg działań:	<ol style="list-style-type: none"> 1. Nauczyciel przechodzi w okno rozpoczęcia lekcji 2. Nauczyciel wybiera grupę dla której chce poprowadzić lekcję 3. System rozpoczyna wideorozmowę 4. System wysyła powiadomienia do członków grupy o rozpoczętej lekcji
	Efekty:	Utworzona zostaje wideokonferencja, podczas której nauczyciel może uczyć uczniów
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Tylko studenci z danej grupy mogą brać udział w danej lekcji • Połączenie musi być stabilne
	Częstotliwość	2
	Istotność	4

ID przypadku użycia	3.5	
Nazwa	Dodaj wykład	
Uzasadnienie biznesowe	1.a - Udostępniania materiałów studentom	
Użytkownicy	Nauczyciel	
Scenariusz główny	Warunek początkowy:	Nauczyciel jest poprawnie zalogowany,
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik przechodzi do okna dodawania wykładów 2. System wyświetla formularz 3. Użytkownik wpisuje nazwę wykład, opis i kategorię której on dotyczy. 4. Użytkownik załącza plik wideo zawierający wykład 5. System sprawdza poprawność plików 6. System pobiera pliki 7. System udostępnia wykład studentom premium.
	Efekty:	Wykłady są dodane i widoczne dla studentów zalogowanych w systemie.
	Wymagania niefunkcjonalne	· Materiały muszą być dostępne dla studentów w ciągu 300 sekund od zakończenia procesu wysyłania plików
	Częstotliwość	2
	Istotność	3
	Scenariusz alternatywny 1 – błąd pliku	<ol style="list-style-type: none"> 1.-5. Jak w scenariuszu głównym 6. System odrzuca pliki ze względu na rozszerzenie bądź błąd odczytu 7. System wyświetla komunikat o błędzie i prosi o ponowne wybranie pliku

ID przypadku użycia	3.6	
Nazwa	Dodaj materiały	
Uzasadnienie biznesowe	1.a - Udostępniania materiałów studentom	
Użytkownicy	Nauczyciel	
Scenariusz główny	Warunek początkowy:	Nauczyciel jest poprawnie zalogowany.
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik przechodzi do okna dodawania materiałów 2. System wyświetla formularz 3. Użytkownik wpisuje nazwę materiału i kategorię której on dotyczy. 4. Użytkownik załącza plik zawierający przygotowane materiały 5. System sprawdza poprawność plików 6. System pobiera pliki 7. System udostępnia materiały studentom.
	Efekty:	Materiały są dodane i widoczne dla studentów zalogowanych w systemie.
	Wymagania niefunkcjonalne	· Materiały muszą być dostępne dla studentów w ciągu 30 sekund od zakończenia procesu wysyłania plików
	Częstotliwość	2
	Istotność	3
	Scenariusz alternatywny 1 – błąd pliku	<ol style="list-style-type: none"> 1.-5. Jak w scenariuszu głównym 6. System odrzuca pliki ze względu na rozszerzenie bądź błąd odczytu 7. System wyświetla komunikat o błędzie i prosi o ponowne wybranie pliku

ID przypadku użycia	3.7	
Nazwa	Pomóż studentowi	
Uzasadnienie biznesowe	1.c - Pomoc studentom	
Użytkownicy	Nauczyciel	
Scenariusz główny	Warunek początkowy:	Nauczyciel jest poprawnie zalogowany, w systemie jest zgłoszenie chęci otrzymania pomocy od studenta
	Przebieg działań:	<ol style="list-style-type: none"> 1. Użytkownik otwiera okno ze zgłoszeniami chęci otrzymania pomocy 2. System wyświetla listę obecnych zgłoszeń 3. Nauczyciel wybiera zgłoszenie 4. System łączy studenta i nauczyciela rozmową tekstową, która może być zamieniona w rozmowę głosową.
	Efekty:	Nauczyciel ma możliwość udzielić pomocy studentowi.
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Żadna niepożądana osoba nie może dołączyć do rozmowy • Przesyłane rozwiązania zostają zachowane w bazie danych do późniejszego wglądu
	Częstotliwość	3
	Istotność	3
	Scenariusz alternatywny - nieudana próba zestawienia połączenia	1.-3. jak w scenariuszu głównym 4. Nieudana próba zestawienia połączenia pomiędzy studentem a nauczycielem 5. System wyświetla komunikat o błędzie nauczycielowi 6. System przenosi nauczyciela ponownie do okna zgłoszeń chęci otrzymania pomocy

ID przypadku użycia	3.8	
Nazwa	Utwórz grupę	
Uzasadnienie biznesowe	1.d - Prowadzenie lekcji	
Użytkownicy	Nauczyciel	
Scenariusz główny	Warunek początkowy:	Nauczyciel jest poprawnie zalogowany.
	Przebieg działań:	<ol style="list-style-type: none"> 1. Nauczyciel wybiera opcję utwórz grupę 2. Nauczyciel wybiera parametry, takie jak nazwa grupy 3. System dodaje grupę do bazy danych
	Efekty:	Zostaje utworzona grupa studentów.
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Żadna niepożądana osoba nie może dołączyć do grupy • Grupa powinna zostać utworzona w systemie w ciągu 10 s
	Częstotliwość	1
	Istotność	3

ID przypadku użycia	3.9	
Nazwa	Dodaj studenta do grupy	
Uzasadnienie biznesowe	1.d - Prowadzenie lekcji	
Użytkownicy	Nauczyciel	
Scenariusz główny	Warunek początkowy:	Nauczyciel jest poprawnie zalogowany, istnieje grupa.
	Przebieg działań:	4. Nauczyciel wyszukuje studenta 5. Nauczyciel dodaje go do grupy 6. System przydziela studentowi dostęp do grupy
	Efekty:	Student jest członkiem grupy i może korzystać z udostępnionych dla jej członków materiałów, brać udział w lekcjach i rozwiązywać testy.
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Żadna niepożądana osoba nie może dołączyć do grupy
	Częstotliwość	2
	Istotność	4

ID przypadku użycia	4.1	
Nazwa	Zarejestruj konto	
Uzasadnienie biznesowe	4.a - Możliwość rejestracji	
Użytkownicy	Niezalogowany	
Scenariusz główny	Warunek początkowy:	Użytkownik nie jest zalogowany, aplikacja jest uruchomiona
	Przebieg działań:	<ol style="list-style-type: none"> 1. Niezalogowany wybiera opcję rejestracji 2. Aplikacja wyświetla formularz 3. Niezalogowany podaje e-mail, hasło, login 4. Aplikacja wysyła informację na serwer 5. Serwer potwierdza, że ani e-mail ani login nie znajdują się w bazie. 6. Serwer tworzy rekord w bazie danych z informacjami o użytkownikach 7. Serwer zwraca informację do klienta 8. Klient wyświetla komunikat o pomyślnej rejestracji
	Efekty:	Użytkownik jest zarejestrowany i może zalogować się w aplikacji
	Wymagania niefunkcjonalne	<ul style="list-style-type: none"> • Przesyłane dane muszą być zabezpieczone, hasło zahasowane • Średni czas operacji nie powinien wynosić więcej niż 1 sekunda
	Częstotliwość	4
	Istotność	2
Scenariusz alternatywny - login lub e-mail już w bazie		<ol style="list-style-type: none"> 1.-4. tak jak w scenariuszu głównym 5. Serwer wykrywa, że dany login/e-mail istnieje w bazie 6. Serwer wysyła informację zwrotną 7. Klient wyświetla informację i prosi o ponowne

		zalogowanie
--	--	-------------

ID przypadku użycia	4.2	
Nazwa	Zaloguj się	
Uzasadnienie biznesowe	4.b - Możliwość logowania	
Użytkownicy	Warunek początkowy:	Użytkownik nie jest zalogowany, aplikacja jest uruchomiona
Scenariusz główny	Przebieg działań:	<ol style="list-style-type: none"> 1. Niezalogowany wpisuje login i hasło 2. Aplikacja haszuje hasło i przesyła je na serwer 3. Serwer odbiera informację i potwierdza że użytkownik jest w bazie 4. Serwer zwraca informację do klienta 5. Klient autoryzuje użytkownika
	Efekty:	Użytkownik jest zalogowany i może korzystać z aplikacji
	Wymagania нефункционалне	<ul style="list-style-type: none"> • Hasło musi być ukryte dla wszystkich • Hasło nie może zostać przechwycone podczas transmisji do serwera • Hasło podczas wpisywania musi być niewidoczne
	Częstotliwość	2
	Istotność	2
	Scenariusz alternatywny - brak loginu w bazie lub złe hasło	<ol style="list-style-type: none"> 1.-2. tak jak w scenariuszu głównym 3. Serwer odbiera informację, ale loginu nie ma w bazie/hasło jest nieprawidłowe 4. Serwer wysyła informację zwrotną 5. Klient wyświetla informację i prosi o ponowne zalogowanie

5.3 Wymagania niefunkcjonalne

wobec całego systemu

1. **Wydajność** -Dostęp do materiałów zajmuje 10ms.

- Najwolniejszy czas do wizualizacji wyników algorytmów jest 1s.
- Stabilność lekcji będzie słabnąć po przekroczeniu 24 użytkowników
- Aplikacja będzie umożliwiać korzystanie przez 12 tys. użytkowników w tym samym czasie.

2. **Bezpieczeństwo**

- Możliwość zgłaszania błędów.
- Wszystkie błędy będą naprawiane w ciągu tygodnia.
- Okresowy zapis bazy danych co 3 miesiące.
- Okresowy serwis aplikacji co 6 miesięcy.
- Okresowa kopia bazy danych co 6 miesięcy.
- Serwisy będą wykonywane w godzinach od 1:00 do 3:00

3. **Zabezpieczenia**

- Wysyłanie raz na miesiąc prośby o zmianę hasła.
- Dostępna możliwość 2-stupniowej(mobilnej) weryfikacji.
- Jeśli użytkownik będzie się chciał zalogować z innego urządzenia zostanie wysłana do niego wiadomość email o potwierdzenie tożsamości.

4. **Inne cechy jakości**

- Zwiększenie maksymalnej aktywnej liczby użytkowników z 12 tys. do 24 tys.
- Aktualizowanie materiałów ze względu na szybki rozwój tej dziedziny.
- Wraz z powiększeniem się aplikacji oraz liczby użytkowników, zatrudnienie dodatkowych osób do zespołu zarządzania aplikacją i jej dalszy rozwój.

6 Zarządzanie projektem

6.1 Zasoby ludzkie

(rzeczywiste lub hipotetyczne) – przy realizacji projektu

Należy założyć, że projekt byłby realizowany w całości jako projekt komercyjny a nie tylko częściowo w ramach zajęć na uczelni

LP	Stanowisko	Liczba osób w zespole	Opis
1.	Kierownik projektu	1	Koordynuje pracę zespołu, zarządza przepływem informacji pomiędzy jego członkami, zarządza dokumentacją.
2.	Programista aplikacji - backend	2	Osoby zajmujące się oprogramowaniem działania aplikacji po stronie klienta, w tym implementacją wszelkich algorytmów numerycznych.
3.	Programista aplikacji - frontend	2	Jego zadaniem jest oprogramowanie klienta pod kątem interakcji z użytkownikiem
4.	Programista serwera	1	Osoba zajmująca się stworzeniem bazy danych, oprogramowaniem działania serwerowej strony aplikacji.
5.	Tester	3	Mają na celu wykrycie wszystkich błędów, zarówno dotyczących działania samej aplikacji jak i (przy udziale konsultanta merytorycznego) poprawności działania wizualizacji i algorytmów obliczeniowych.
6.	Konsultant merytoryczny	1	Osoba, której zadaniem byłoby wsparcie merytoryczne z dziedziny analizy numerycznej, dotyczące dokładnego obliczenia i zwizualizowania algorytmów numerycznych.
7.	Grafik/UI Designer	1	Zajmuje się opracowaniem projektu graficznego wyglądu i działania aplikacji.
8.	Marketingowiec	1	Zajmuje się marketingiem aplikacji.

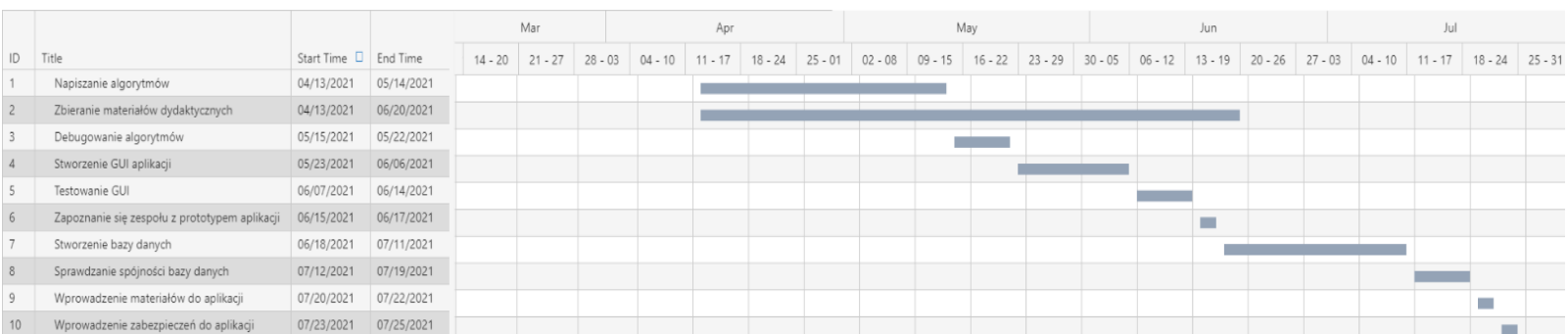
6.2 Harmonogram prac

Etapy mogą się składać z zadań.

Wskazać czasy trwania poszczególnych etapów i zadań – wykres Gantta.

obejmuje również harmonogram wdrożenia projektu – np. szkolenie, rozruch, konfiguracja, serwis – może obejmować różne wydania (tj. o różnej funkcjonalności – personal, professional, enterprise) i wersje (1.0, 1.5, itd.)

- Napisanie algorytmów.
- Debugowanie algorytmów.
- Stworzenie GUI aplikacji
- Debugowanie GUI
- Zapoznanie się z GUI z resztą zespołu.
- Stworzenie bazy danych
- Debugowanie bazy danych
- Zebranie materiałów.
- Sprawdzenie poprawności materiałów.
- Wprowadzenie materiałów do aplikacji
- Wprowadzenie zabezpieczeń do aplikacji



Z czego wynika długi czas prac nad jakimś etapem projektu.

Napisanie algorytmów: Nad etapem pracuje mała liczba programistów, a sama liczba oraz złożoność niektórych algorytmów powoduje czasochłonność ich napisania.

Zbieranie materiałów dydaktycznych: Długi czas pracy nad zakończeniem tego etapu jest spowodowany małą liczbą osób odpowiedzialnych za zebranie dużej ilości materiałów dydaktycznych.

Stworzenie bazy danych: Nad stworzeniem bazy danych pracuje mały zespół, który ma na celu stworzenie bazy danych, która po ukończeniu nie będzie wymagała naprawy jakichkolwiek błędów.

6.3 Etapy/kamienie milowe projektu

dla głównych etapów projektu

1. Napisanie poprawnie działających rozwiązujących problemy matematyczne algorytmów.
2. Stworzenie GUI.
3. Zaprojektowanie relacyjnej bazy danych dla aplikacji
4. Wprowadzenie materiałów do aplikacji
5. Stworzenie funkcji do przeprowadzania e-lekcji.
6. Wprowadzenie zabezpieczeń dla aplikacji i bazy danych.
7. Wdrożenie aplikacji na rynek.

7 Zarządzanie ryzykiem

7.1 Lista czynników ryzyka

1. Techniczne
 1. Awaria zewnętrznej serwerowni
 2. Wykorzystanie całej dostępnej przestrzeni dyskowej
 3. Przeciążenie serwera spowodowane dużą liczbą użytkowników podłączonych jednocześnie
2. Biznesowe
 1. Mała liczba studentów chętnych do pomocy innym studentom. (na tyle mała, że oczekiwanie na chętnego do pomocy trwa ponad godzinę)
 2. Mała liczba użytkowników usługi (mniej niż 5000)
 3. Mały odsetek użytkowników, którzy zakupili wersję Premium (mniej niż 10%)
3. Ludzkie
 1. Odejście kluczowych członków zespołu
4. Funkcjonalne działanie aplikacji
 1. Błędy w algorytmach obliczeniowych
 2. Błędy w generowaniu zadań - generowanie zadań niemożliwych do rozwiązania

7.2 Ocena ryzyka

Wpływ oceniany jest w skali 5-stopniowej, gdzie 1 to negatywny wpływ przypadku jest bardzo niski, wręcz pomijalny, a 5 to bardzo wysoki

ID	Nazwa przypadku	Prawdopodobieństwo	Wpływ
1.1	Awaria zewnętrznej serwerowni	20%	4
1.2	Wykorzystanie całej dostępnej przestrzeni dyskowej	30%	2
1.3	Przeciążenie serwera spowodowane dużą liczbą użytkowników podłączonych jednocześnie	40%	3
2.1	Mała liczba studentów chętnych do pomocy innym studentom	60%	3
2.2	Mała liczba użytkowników usługi (mniej niż 5000)	40%	4
2.3	Mały odsetek użytkowników, którzy zakupili wersję Premium (mniej niż 10%)	50%	3
3.1	Odejście członków zespołu	20%	2
4.1	Błędy w algorytmach obliczeniowych	15%	4
4.2	Błędy w generowaniu zadań - generowanie zadań niemożliwych do rozwiązania	25%	4

7.3 Plan reakcji na ryzyko

Ryzyko	Reakcja
Techniczne	
Awaria zewnętrznej serwerowni	Wysłanie komunikatu o awarii serwerowni. Naprawienie serwerowni. Jeśli naprawa nie pomoże zakupić nowy sprzęt i zamienić go z zepsutym.
Wykorzystanie całej dostępnej przestrzeni dyskowej	Usunięcie nieaktywnych użytkowników, którzy nie logowali się od 6 miesięcy. Jeśli to nie pomoże. Dokupienie dodatkowej przestrzeni dyskowej.
Przeciążenie serwera spowodowane dużą liczbą użytkowników podłączonych jednocześnie	Chwilowe ograniczenie możliwości nowych połączeń do serwera. Włączenie kolejki. Wyłączenie braku możliwości na nowe połączenia. Dokupienie dodatkowych serwerów do odciążenia aktualnych serwerów.
Biznesowe	
Mała liczba studentów chętnych do pomocy innym studentom	Zatrudnienie dodatkowych osób do zespołu jako osoba pomagająca studentom
Mała liczba użytkowników usługi	Wprowadzenie ogłoszeń na kierunkach, w których znajdują się przedmiot, które obejmują zakres aplikacji.
Mały odsetek użytkowników, którzy zakupili wersję Premium	Zmniejszenie ceny zakupu Premium oraz wprowadzenie 7-dniowej darmowej wersji Premium.
Ludzkie	
Odejście członków zespołu	Zatrudnienie odpowiednich nowych członków do zespołu
Funkcjonalne działanie aplikacji	

Błędy w algorytmach obliczeniowych	Wyłączenie funkcji wykorzystującej dany algorytm oraz wyświetlenie komunikatu o błędach w obliczaniu danej funkcji. Debugowanie algorytmu.
------------------------------------	---

8 Zarządzanie jakością

1.1 Rozwiązywanie równań nieliniowych

- Metoda bisekcji
- Metoda Newtona
- Metoda siecznych

1.2 Rozwiązywanie układów równań liniowych

- Eliminacja Gaussa z wyborem elementów głównych
- Rozkład LU

1.3 Aproksymacja funkcji

- Interpolacja wielomianowa(wielomian Lagrange'a, wielomian Newtona)
- Interpolacja funkcji wielu zmiennych

1.4 Różniczkowanie i całkowanie numeryczne

- Różniczkowanie numeryczne
- Metoda prostokątów
- Metoda trapezów
- Metoda Monte Carlo

1.5 Programowanie liniowe

- Programowanie liniowe
- Algorytm sympleks

1.6 Optymalizacja

- Metoda spadku
- Metoda połowienia
- Metoda złotego podziału
- Aproksymacja kwadratowa
- Aproksymacja sześcienna
- Metoda Newtona

8.1 Scenariusze i przypadki testowe

ID	1														
Nazwa	Poprawne obliczanie wyników przez algorytmy obliczające równania układów nieliniowych.														
Kategoria	Algorytmy:Rozwiązywanie równań nieliniowych														
Opis	Sprawdzenie czy grupa algorytmów zwraca poprawny wynik.														
Tester	Tester														
Termin	05/15/2021														
Narzędzia	Visual Studio 2019														
Przebieg działań	<table><tr><th>Lp.</th><th>Tester</th><th>System</th></tr><tr><td>1.</td><td>Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi</td><td>Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik</td></tr><tr><td>2.</td><td>Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem</td><td></td></tr><tr><td>3.</td><td>Tester wprowadza źle przygotowane dane wejściowe</td><td>System zwraca komunikat o błędzie oraz powodu błędu</td></tr></table>			Lp.	Tester	System	1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik	2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem		3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu
Lp.	Tester	System													
1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik													
2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem														
3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu													
Założenia	Algorytm w systemie powinien się zatrzymać oraz zwrócić poprawny wynik.														
Środowisko															
Warunki wstępne	Przygotowanie nieliniowych układów równań Przygotowanie wyników do poprawnych danych wejściowych														
Dane wejściowe	Tester wpisuje przygotowaną funkcje wraz z podanymi przedziałami. układ równań nieliniowych(co dokładnie tester ma wpisać)														
Zestaw danych	f = cos(x), przedział <0, 2>														

testowych	$f = (x+2)^5$, przedział $<-3, 0>$
-----------	-------------------------------------

ID	2														
Nazwa	Poprawne obliczanie wyników przez algorytmy obliczające równania układów liniowych.														
Kategoria	Algorytmy:Rozwiązywanie równań liniowych														
Opis	Sprawdzenie czy grupa algorytmów zwraca poprawny wynik.														
Tester	Tester														
Termin	05/16/2021														
Narzędzia	Visual Studio 2019														
Przebieg działań	<table><tr><th>Lp.</th><th>Tester</th><th>System</th></tr><tr><td>1.</td><td>Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi</td><td>Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik</td></tr><tr><td>2.</td><td>Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem</td><td></td></tr><tr><td>3.</td><td>Tester wprowadza źle przygotowane dane wejściowe</td><td>System zwraca komunikat o błędzie oraz powodu błędu</td></tr></table>			Lp.	Tester	System	1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik	2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem		3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu
Lp.	Tester	System													
1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik													
2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem														
3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu													
Założenia	Algorytm w systemie powinien się zatrzymać oraz zwrócić poprawny wynik.														
Środowisko															
Warunki wstępne	Przygotowanie liniowych układów równań. Przygotowanie wyników do poprawnych danych wejściowych.														
Dane wejściowe	Zmiana liczba wierszy w zależności od ilości równań. Tester wprowadza jedno równanie do jednego wiersza w aplikacji, do momentu aż wpisze cały zestaw równań.														
Zestaw danych testowych	$2 \cdot x + 2 \cdot y + z = 3$ $x + 4 \cdot y + z = -1$														

	$y + 2z = 1$ $4x + y + 4z = 2$ $2x - y + 2z = -4$ $x + y + 2z = -1$ $2x + 2y + z - v = 6$ $3y + z + 2v = -1$ $x + 2y + 2z + v = 4$ $x + y + z = 4$
--	---

ID	3														
Nazwa	Poprawne obliczanie wyników przez algorytmy obliczające numeryczne całkowanie.														
Kategoria	Algorytmy:Różniczkowanie i całkowanie numeryczne														
Opis	Sprawdzenie czy grupa algorytmów zwraca poprawny wynik.														
Tester	Tester														
Termin	05/18/2021														
Narzędzia	Visual Studio 2019														
Przebieg działań	<table><tr><th>Lp.</th><th>Tester</th><th>System</th></tr><tr><td>1.</td><td>Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi</td><td>Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik</td></tr><tr><td>2.</td><td>Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem</td><td></td></tr><tr><td>3.</td><td>Tester wprowadza źle przygotowane dane wejściowe</td><td>System zwraca komunikat o błędzie oraz powodu błędu</td></tr></table>			Lp.	Tester	System	1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik	2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem		3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu
Lp.	Tester	System													
1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik													
2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem														
3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu													
Założenia	Algorytm w systemie powinien się zatrzymać oraz zwrócić poprawny wynik.														
Środowisko															
Warunki wstępne	Przygotowanie danych wejściowych oraz wyniki.														
Dane wejściowe	Tester wybiera ilość wektorów do wprowadzenia oraz jakiej osi jest odpowiednikiem wektorem czy osi x czy osi y.														

Zestaw danych testowych	$x = [0\ 10\ 20\ 30\ 40\ 50\ 60\ 70\ 80\ 90\ 100\ 110\ 120\ 130\ 140\ 150];$ $y = [0\ 25\ 30\ 28\ 20\ 12\ 8\ 11\ 18\ 20\ 16\ 15\ 20\ 27\ 30\ 0];$ $x_2 = [0\ -23\ -30\ -29\ -24\ -13\ -10\ -6\ -5\ -9\ -16\ -31\ -41\ -37\ -30\ 0];$
----------------------------	--

ID	4														
Nazwa	Poprawne obliczanie wyników przez algorytmy obliczające numeryczne różniczkowanie.														
Kategoria	Algorytmy:Różniczkowanie i całkowanie numeryczne														
Opis	Sprawdzenie czy grupa algorytmów zwraca poprawny wynik.														
Tester	Tester														
Termin	05/18/2021														
Narzędzia	Visual Studio 2019														
Przebieg działań	<table><tr><th>Lp.</th><th>Tester</th><th>System</th></tr><tr><td>1.</td><td>Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi</td><td>Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik</td></tr><tr><td>2.</td><td>Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem</td><td></td></tr><tr><td>3.</td><td>Tester wprowadza źle przygotowane dane wejściowe</td><td>System zwraca komunikat o błędzie oraz powodu błędu</td></tr></table>			Lp.	Tester	System	1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik	2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem		3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu
Lp.	Tester	System													
1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik													
2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem														
3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu													
Założenia	Algorytm w systemie powinien się zatrzymać oraz zwrócić poprawny wynik.														
Środowisko															
Warunki wstępne	przygotowanie danych wejściowych oraz wyniku.														
Dane wejściowe	Tester wybiera według dowolnej kolejności 3 sposoby różniczkowania numerycznego oraz wprowadza do nich														

	funkcje, przedział oraz krok.
Zestaw danych testowych	$f = \cos(2 \cdot x)$, przedział $\langle 0, 6 \rangle$, krok = 0.5

ID	5		
Nazwa	Poprawne obliczanie wyników przez algorytmy programowania liniowego		
Kategoria	Algorytmy:Programowanie liniowe		
Opis	Sprawdzenie czy grupa algorytmów zwraca poprawny wynik.		
Tester	Tester		
Termin	05/19/2021		
Narzędzia	Visual Studio 2019		
Przebieg działań	Lp.	Tester	System
	1.	Tester wywołuje algorytm z wcześniej przygotowanymi danymi wejściowymi	Algorytm w systemie zatrzymuję się oraz zwraca poprawny wynik
	2.	Tester porównuje zwrócony wynik przez system z wcześniej przygotowanym wynikiem	
	3.	Tester wprowadza źle przygotowane dane wejściowe	System zwraca komunikat o błędzie oraz powodu błędu
Założenia	Algorytm w systemie powinien się zatrzymać oraz zwrócić poprawny wynik.		
Środowisko			
Warunki wstępne	Przygotowanie danych wejściowych oraz wynik.		
Dane wejściowe	Tester wprowadza układ równań wraz z funkcją celu(Z)		

Zestaw danych testowych	$2x + y + z = 10$ $3x + 3y + v = 24$ $2x + w = 8$ $Z = 300x + 200y + 0z + 0v + 0w$
----------------------------	---

9 Projekt techniczny

9.1 Opis architektury systemu

Architekturę naszego systemu można podzielić na dwie główne części:

- serwerową
- po stronie klienta - aplikacja zainstalowana na komputerze użytkownika

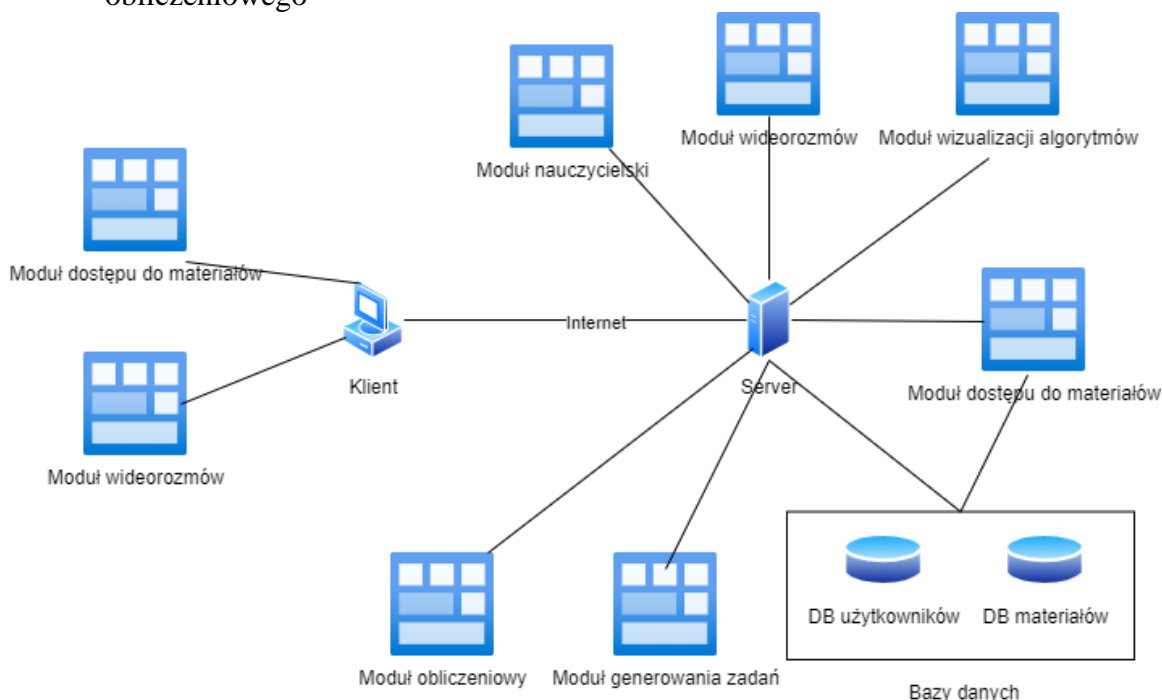
Część serwerowa powinna składać się z modułów:

- udostępniania dostępu do umieszczonych wcześniej materiałów i nagrań wideo - moduł ten ma udostępniać dostęp do zasobów bazy danych materiałów konkretnym użytkownikom. Na tym etapie moduł musi sprawdzić uprawnienia dostępu danego użytkownika do danego materiału.
- tworzenia i utrzymywania połączenia do wideorozmów - po otrzymaniu polecenia od użytkownika moduł ma utworzyć lekcję, kontrolować użytkowników którzy mogą w niej uczestniczyć, wysłać powiadomienia do użytkowników.
- bazy danych użytkowników (użytkownicy, grupy, przesyłane wiadomości) - baza musi przechowywać dane kont użytkowników oraz utworzone przez nauczycieli grupy. Poza tym, musimy tu przechowywać relacje między użytkownikami (znajomi) oraz przesyłane wiadomości. Baza ta musi być szczególnie mocno chroniona, ponieważ zawiera wrażliwe informacje.
- bazy danych materiałów i nagrań wideo - baza danych, która ma przechowywać wszystkie umieszczone w programie materiały i nagrania wideo. Poza samymi materiałami baza musi przechowywać informację, jaki rodzaj użytkownika ma dostęp do danego materiału.

Część kliencka powinna składać się z następujących modułów:

- brania udziału w wideorozmowach - moduł powinien pozwolić na połączenie się do istniejącej wideorozmowy (lekcji) za pośrednictwem serwera. Moduł ma za zadanie wysłać żądanie dostępu do rozmowy oraz połączenia się z nim.
- dostępu do materiałów, nagrań i zadań - moduł wysyła żądanie dostępu do serwera, które zostaje obsłużone przez moduł udostępniania dostępu do umieszczonych wcześniej materiałów i nagrań wideo
- generowania zadań obliczeniowych - moduł ten przyjmuje na wejście od użytkownika rodzaj zadania numerycznego, następnie na podstawie zaimplementowanych algorytmów i z uwzględnieniem nałożonych ograniczeń (takich, aby nie dopuścić do wygenerowania zadań sprzecznych/nierozwiązywalnych) zwraca zadanie, parametry dane i początkowo ukrytą dla użytkownika odpowiedź, realizowaną przy wsparciu modułu obliczeniowego.
- moduł wizualizacji - moduł ten ma za zadanie przeprowadzenie wizualizacji. Musi on współpracować z modułem obliczeniowym. Dzięki danym uzyskanym od tego modułu powinien on wyświetlić interaktywną wizualizację, umożliwić jej przesuwanie krok po kroku i eksport do pliku graficznego
- Moduł obliczeniowy - moduł, który wykonuje wszelkie niezbędne obliczenia, realizując tym samym algorytmy numeryczne zaimplementowane w programie.

- Moduł nauczycieli - służący do zarządzania grupami studentów, dodawania materiałów. Obejmuje to też generowanie testów przy udziale modułu obliczeniowego



9.2 Technologie implementacji systemu

tabela z listą wykorzystanych technologii, każda z uzasadnieniem

Backend	
Nazwa	Uzasadnienie
Python	Język python zostanie wykorzystany do stworzenia części back-end z powodu prostych w użyciu bibliotek umożliwiających obliczanie i wizualizację algorytmów oraz łatwość tworzenia aplikacji webowych.
Numpy	Biblioteka Numpy języka Python zostanie wykorzystana w celu ułatwienia wykonywania wszelkich obliczeń dotyczących zarówno wykonywania algorytmów jak i generowania zadań.
Matplotlib	Biblioteka Matplotlib języka Python zostanie wykorzystana w celu wizualizacji wszystkich algorytmów, które zostaną wykonane.
MySQL	Sprawdzony DBMS, który zostanie wykorzystany w celu przechowywania i udostępniania danych użytkowników oraz materiałów dydaktycznych. System ten jest powszechnie dostępny i znany co przemawia za jego użyciem.

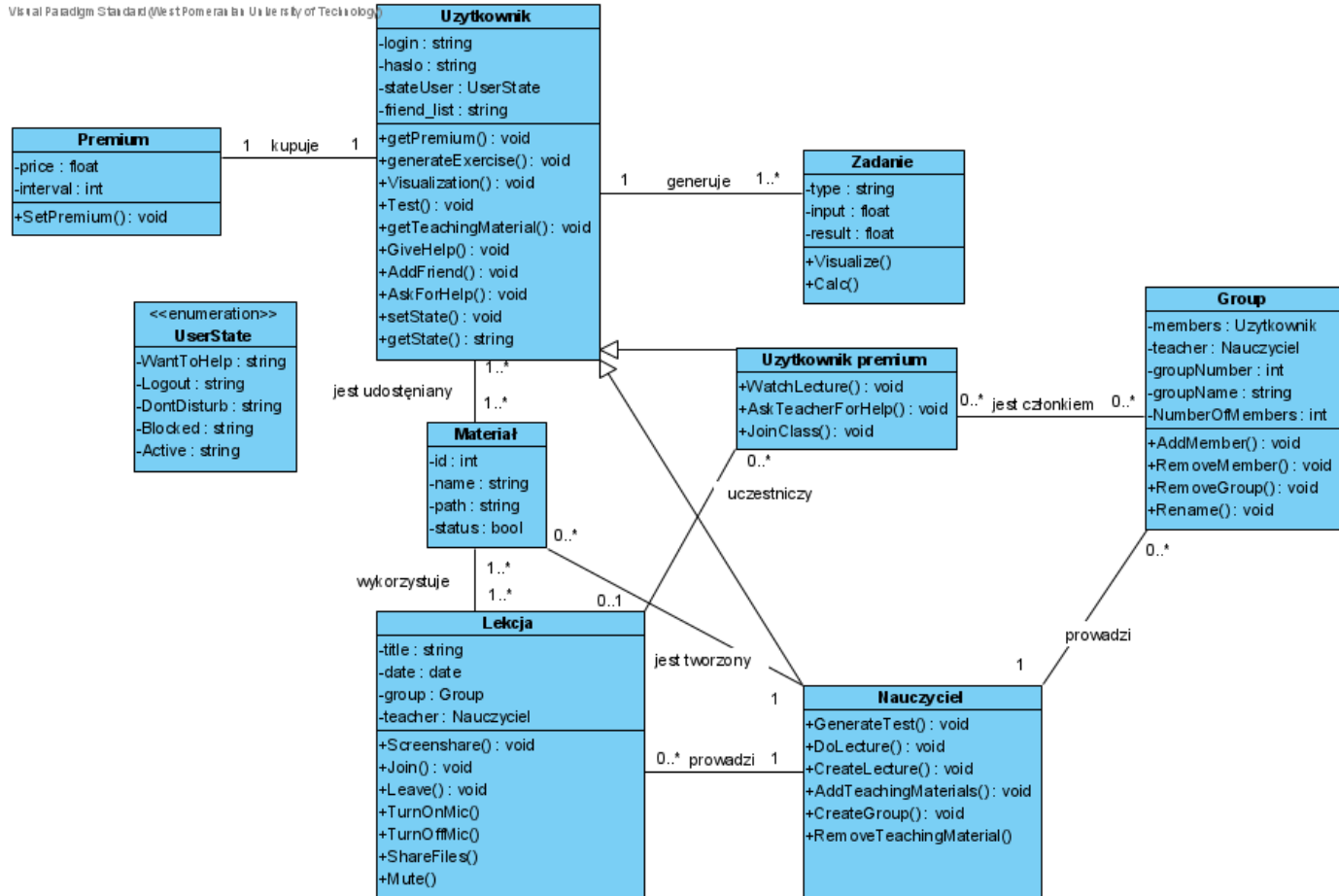
Django	Biblioteka Django posłuży do prawidłowego działania części serwerowej odpowiedzialnej za komunikację z klientem.
---------------	--

Frontend	
HTML	Język HTML zostanie użyty do zbudowania struktury aplikacji klienckiej.
CSS	Język CSS zostanie użyty do skonfigurowania wyglądu aplikacji.
JS	Język JS będzie obsługiwał wszelkie formularze i dynamiczne elementy na stronie.

9.3 Diagramy UML

każdy diagram ma mieć tytuł oraz ma być na osobnej stronie diagramy przypadków użycia umieszczone w punkcie 5.2.2, a nie tutaj.

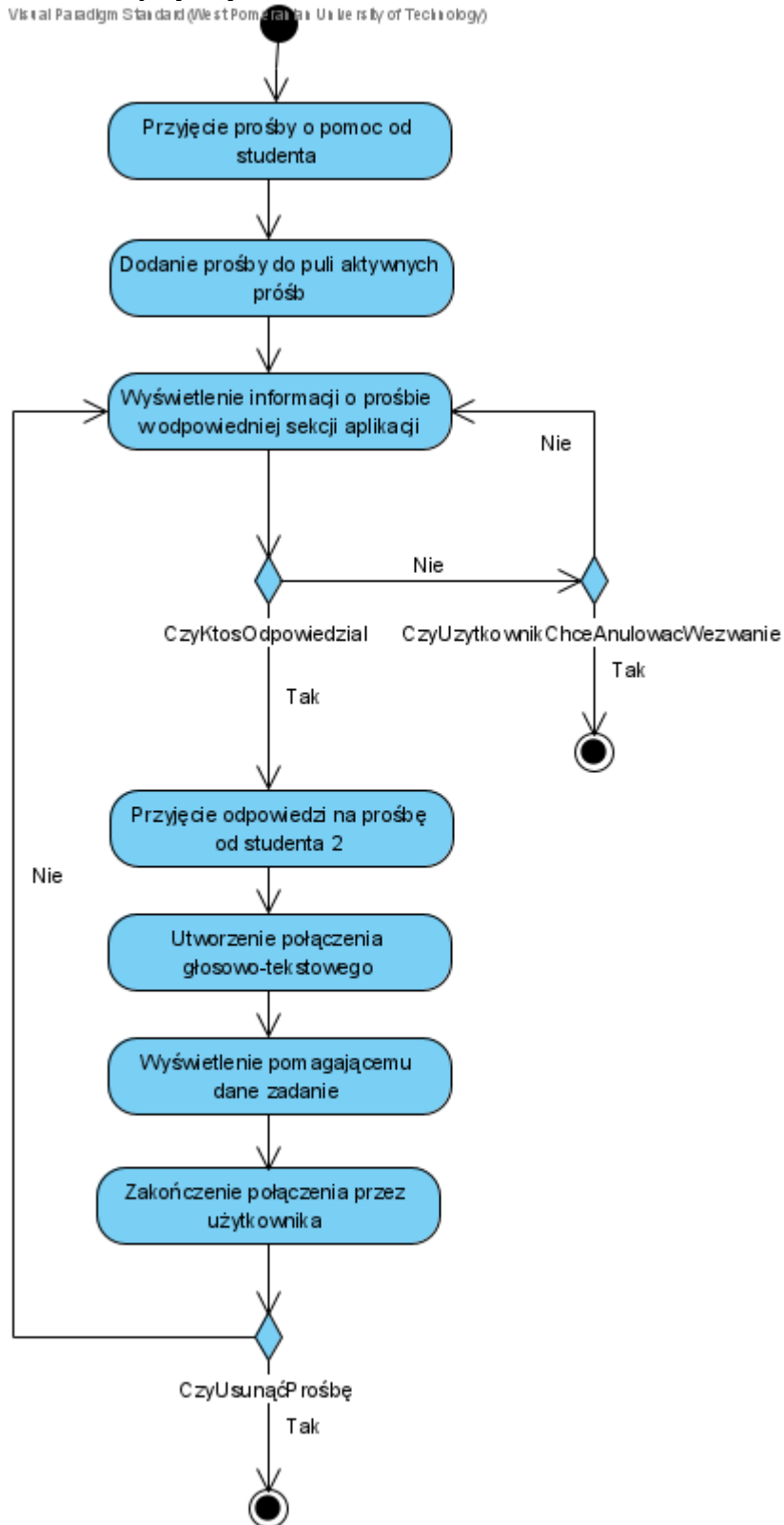
9.3.1 Diagram(-y) klas(najbardziej czasochłonne)



9.3.2 Diagram(-y) czynności

9.3.2.1 Pomoc między użytkownikami

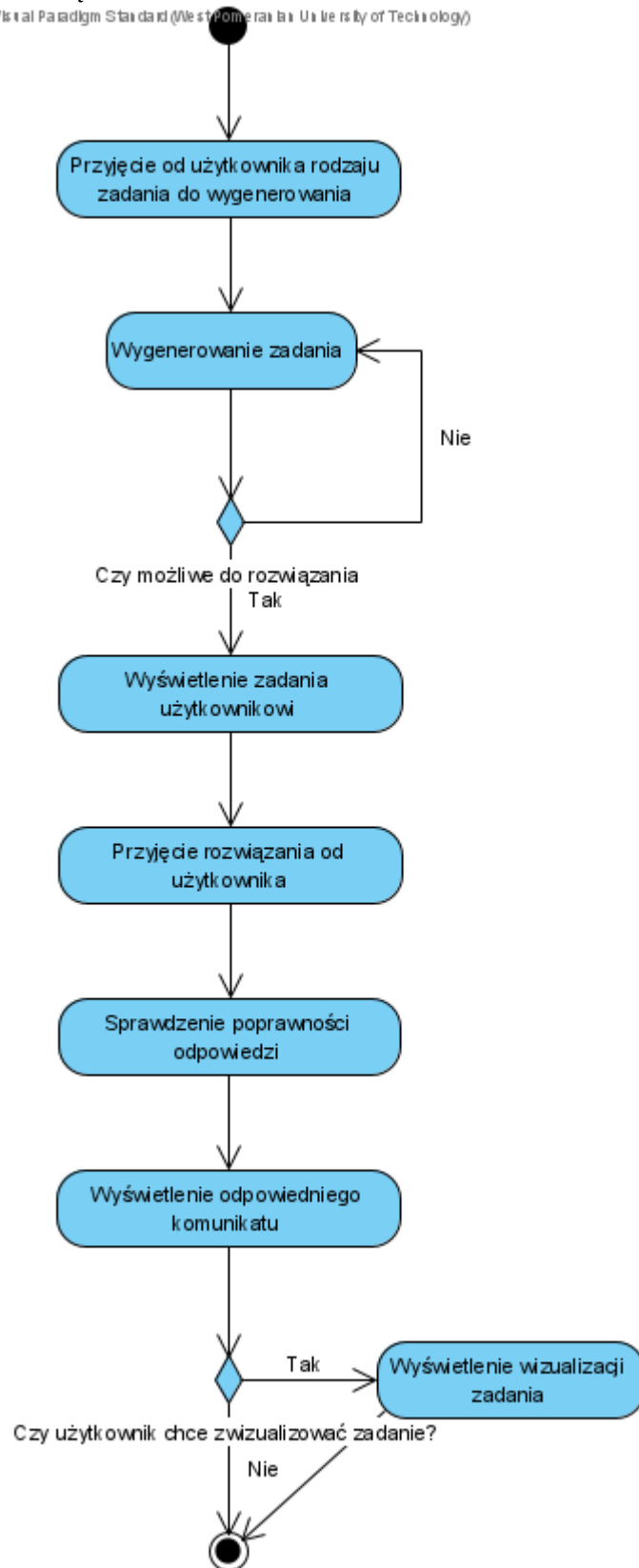
Visual Paradigm Standard (West Pomeranian University of Technology)



9.3.2.2 Rozwiązanie

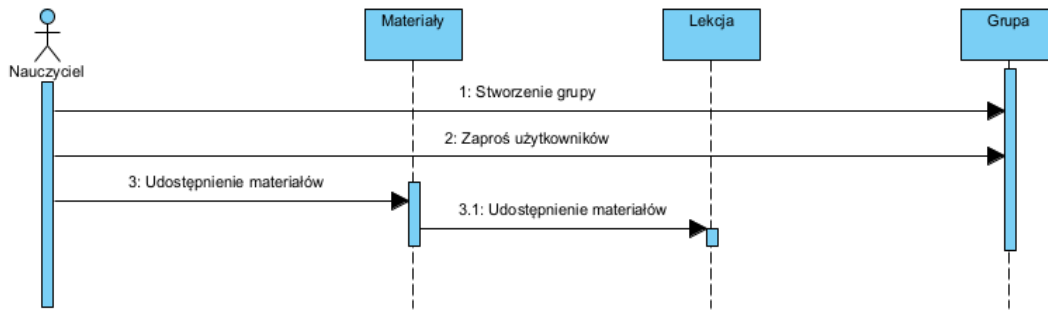
Visual Paradigm Standard (Wrocław: Wrocław University of Technology)

zadania

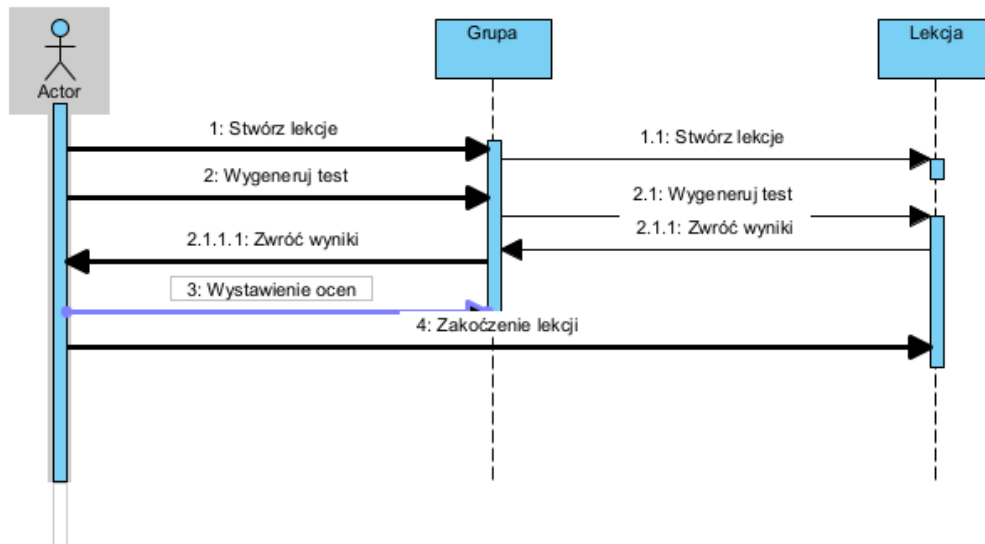


9.3.3 Diagramy sekwencji

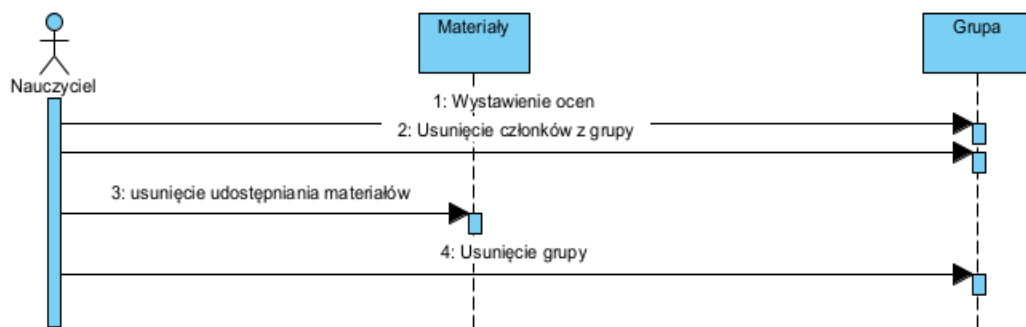
Tworzenie grupy



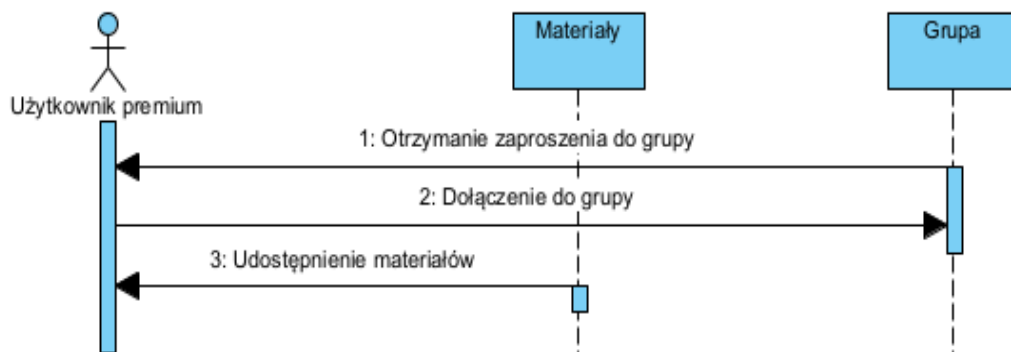
Prowadzenie zajęć



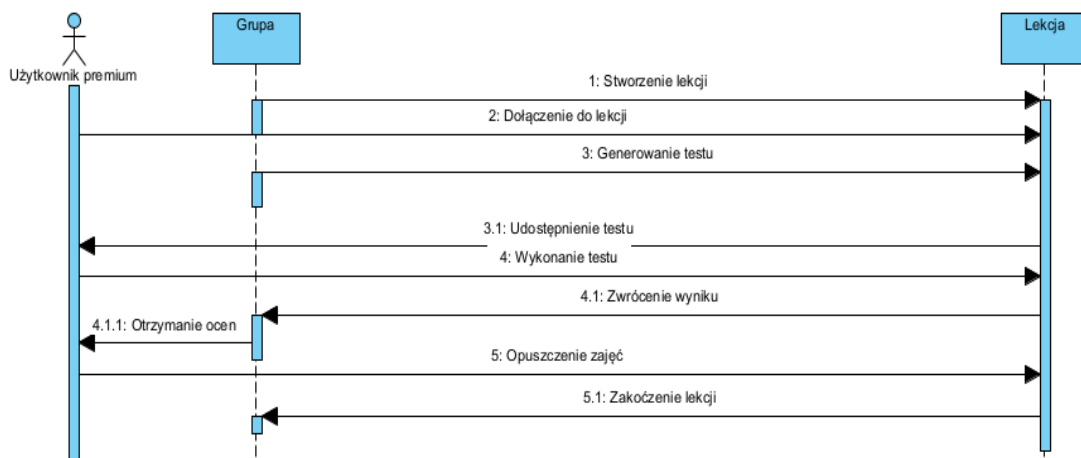
Zakończenie semestru zajęć



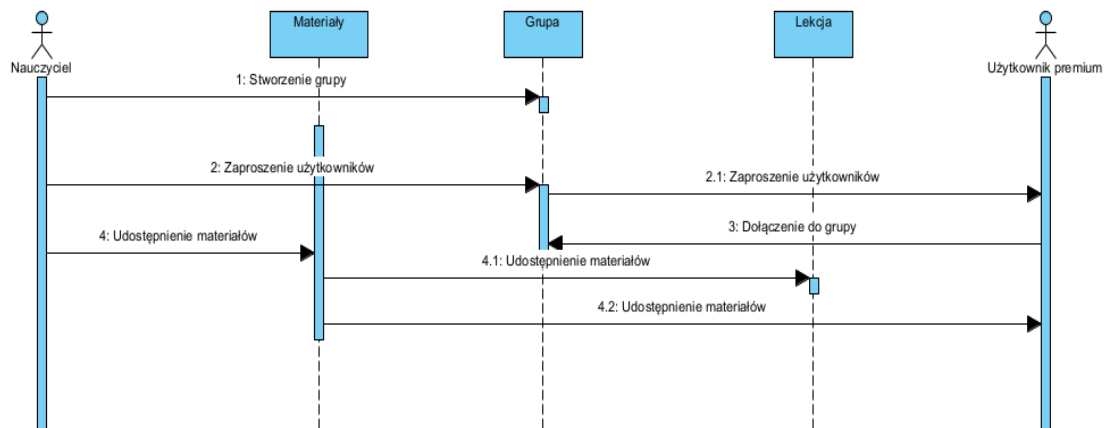
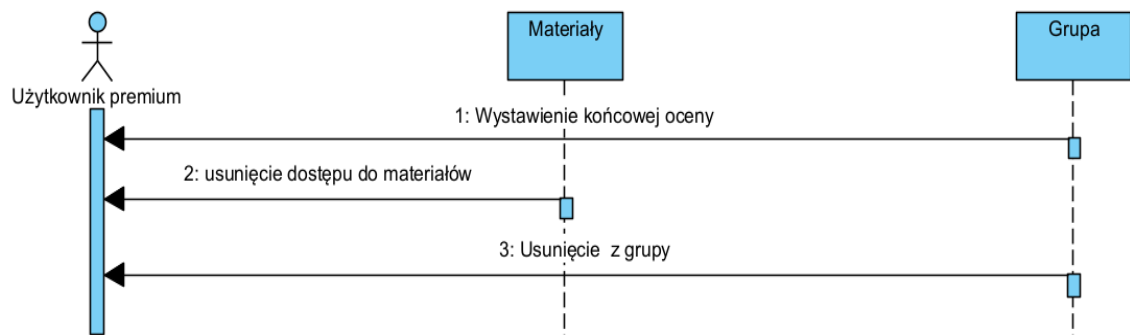
Dołączanie do grupy

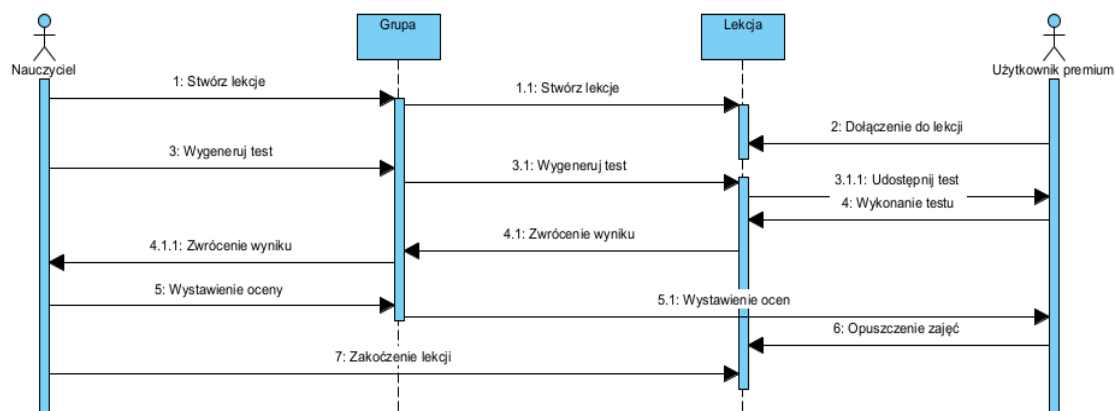
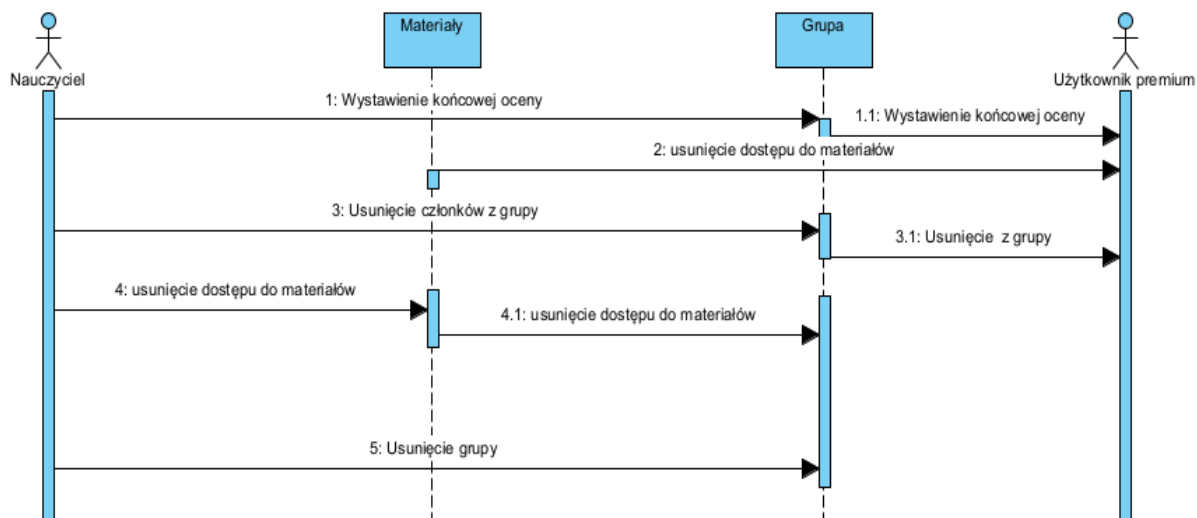


Uczestniczenie na zajęciach



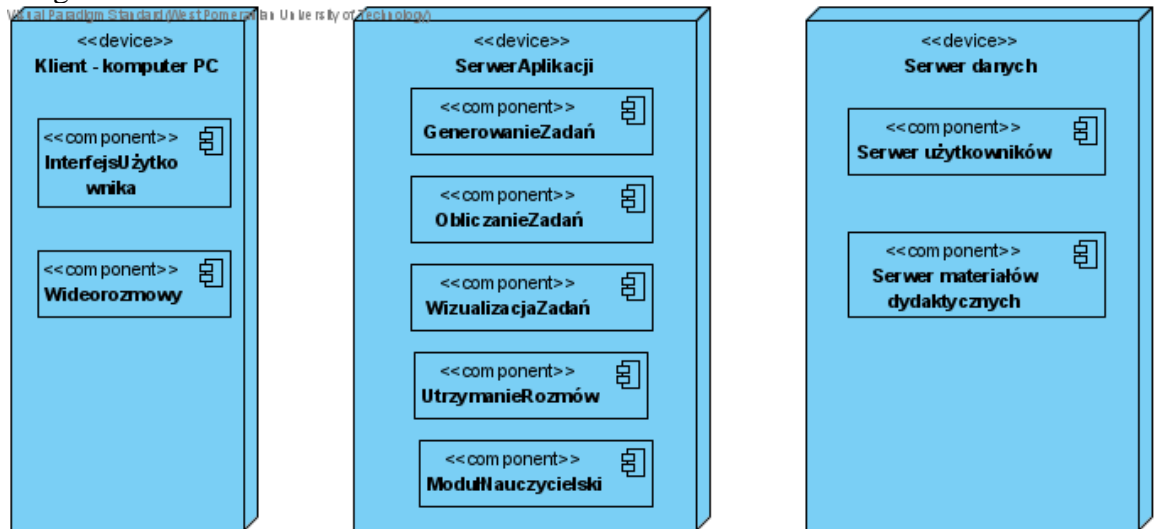
Koniec semestru





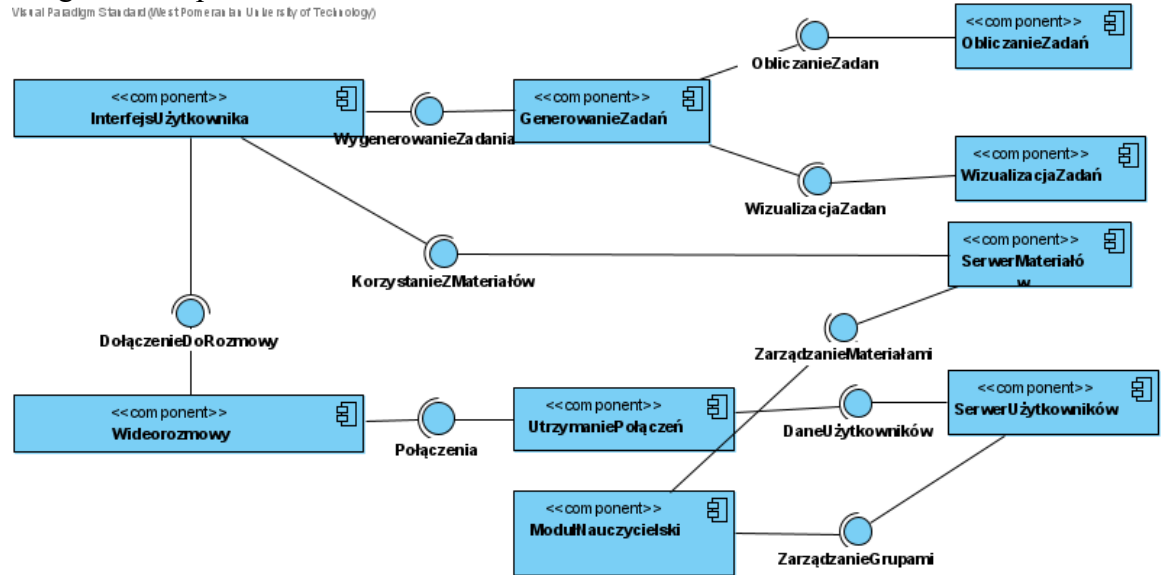
9.3.4 Inne diagramy

9.3.4.1 Diagram Rozmieszczenia



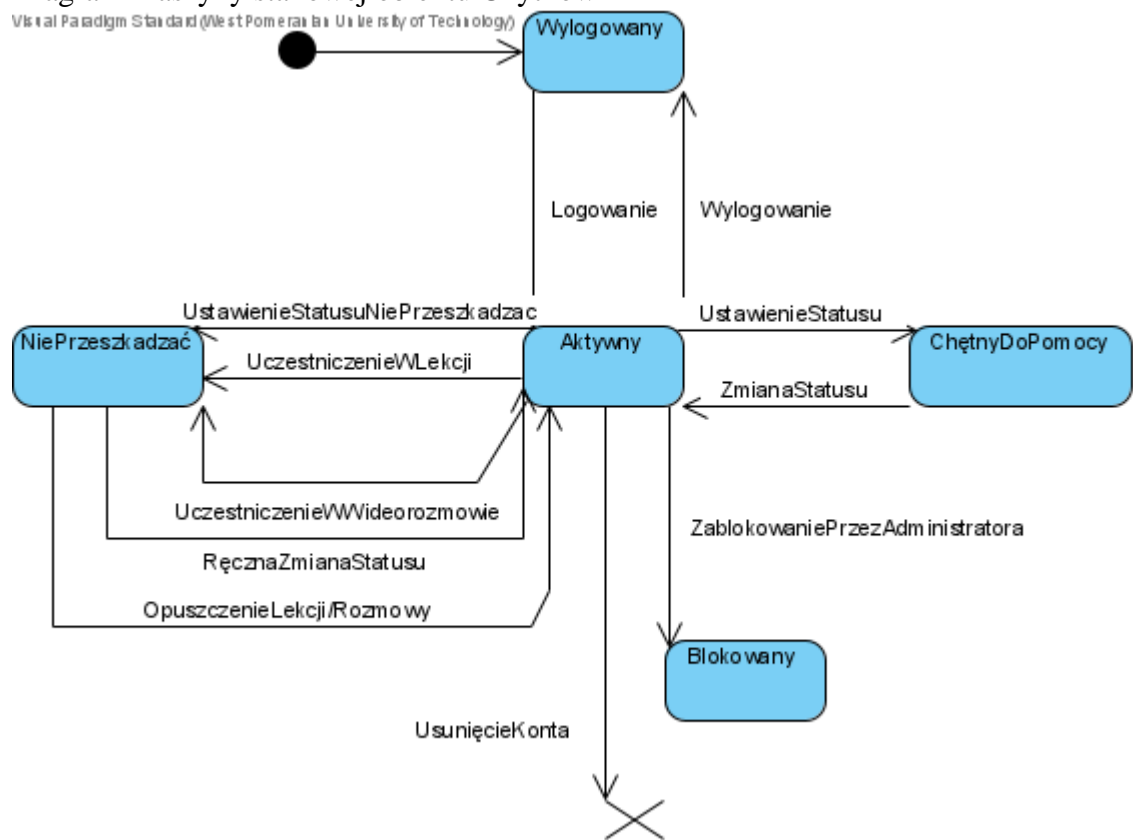
9.3.4.2 Diagram Komponentów

Visual Paradigm Standard (West Pomeranian University of Technology)



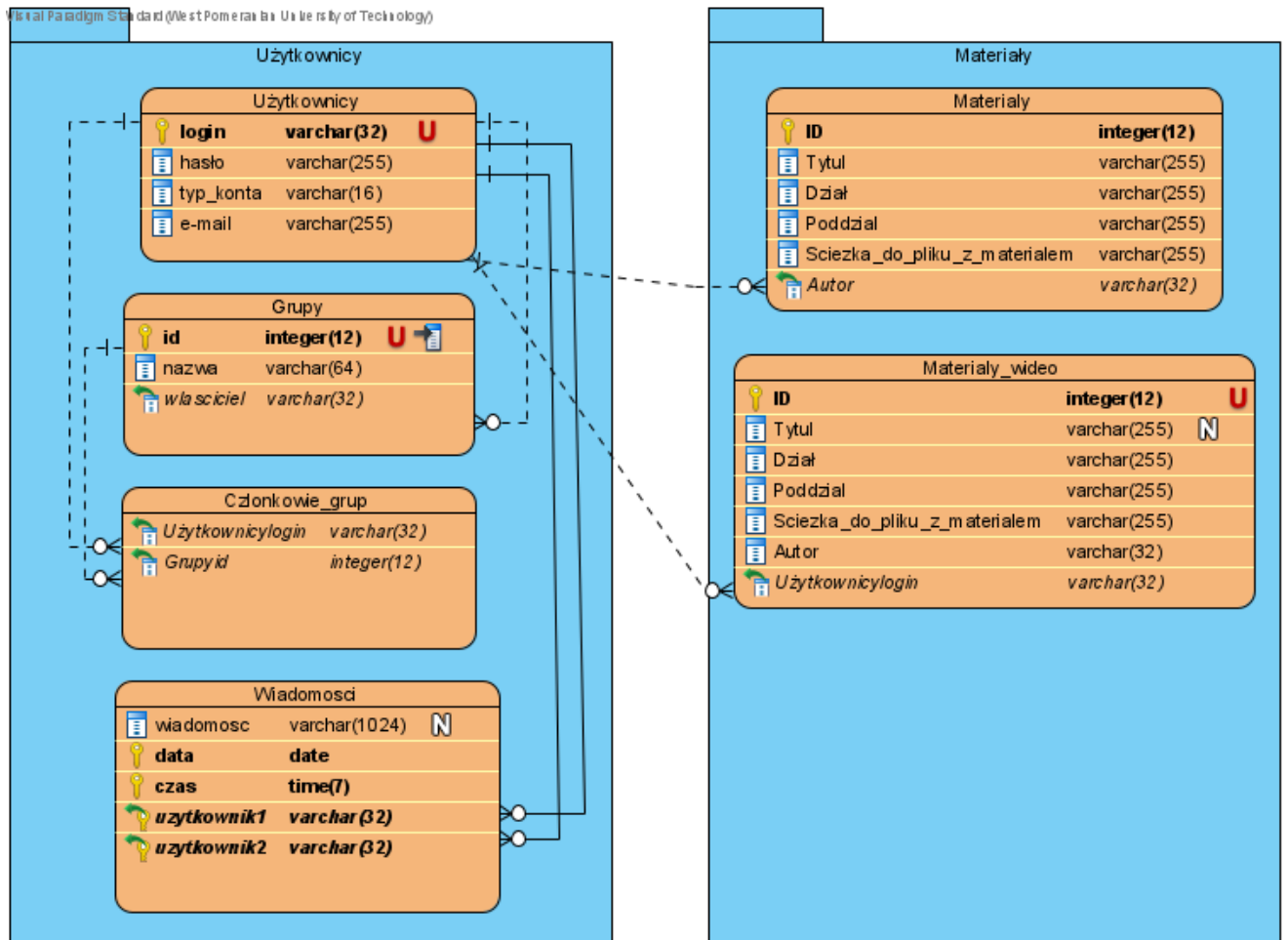
9.3.4.3 Diagram maszyny stanowej obiektu Uzytkownik

Visual Paradigm Standard (Next Generation Usability of Technology)



9.4 Projekt bazy danych

9.4.1 Schemat



9.4.2 Projekty szczegółowe tabel

Użytkownicy		
login	tomek34	Login użytkownika
hasło	149c190811c6c6cea4b51567d4314348	Hasło użytkownika w postaci zahaszkowanej
typ_konta	student	Jedna z trzech wartości: 'student', 'nauczyciel', 'student_premium'
e-mail	tomek34@gmail.com	Adres mailowy użytkownika

Grupy		
id	22	unikalny numer identyfikacyjny grupy
nazwa	12B_UJ	nazwa grupy, nie musi być unikalna
wlasciciel	J_Kowalski	login uzytkownika (nauczyciela) który utworzył grupę, klucz obcy z tabeli Uzytkownicy

Czlonkowie_grup		
Uzytkownicylogin	tomek34	login uzytkownika nalezacego do grupy o danym id, klucz obcy z tabeli Uzytkownicy
Grupyid	22	numer identyfikacyjny nalezacy do grupy do ktorej nalezy dany uzytkownik, klucz obcy z tabeli Grupyid

Wiadomosci		
wiadomosc	Pomozesz mi z interpolacja?	Tresc przeslanej wiadomosci
data	2021-05-30	Data danej wiadomosci
czas	16:16:05	Godzina, minuta, sekunda przeslania danej wiadomosci
uzytkownik1	tomek34	Login nadawcy wysylanej wiadomosci, klucz obcy z tabeli uzytkownicy
uzytkownik2	asia_opo	Login odbiorcy wysylanej wiadomosci, klucz obcy z tabeli uzytkownicy

Materialy		
ID	43	Numer identyfikacyjny danego materiału tekstowego
Tytuł	Metoda Newtona	Tytuł materiału
Dział	Optymalizacja	Dział analizy numerycznej, której dotyczy materiał
Poddział	Ekstremum funkcji jednej zmiennej	Poddział analizy numerycznej, której dotyczy materiał
Sciezka_do_pliku_z_materialem	//data/materialy/optimalizacja/ekstrema/newton.html	Sciezka dostępu do pliku .html na serwerze z danym materiałem
Autor	J_Kowalski	Login użytkownika (nauczyciela) który dodał dany materiał, klucz obcy z tabeli uzytkownicy

Materialy_wideo		
ID	43	Numer identyfikacyjny danego materiału tekstowego
Tytuł	Metoda Newtona	Tytuł materiału
Dział	Optymalizacja	Dział analizy numerycznej, której dotyczy materiał
Poddział	Ekstremum funkcji jednej zmiennej	Poddział analizy numerycznej, której dotyczy materiał
Sciezka_do_pliku_z_materialem	//vids/materialy/optimalizacja/ekstrema/newton.html	Sciezka dostępu do pliku .mp4 na serwerze z danym materiałem
Autor	J_Kowalski	Login użytkownika (nauczyciela) który dodał dany materiał, klucz obcy z tabeli uzytkownicy

9.5 Projekt interfejsu użytkownika

Co najmniej dla głównej funkcjonalności programu – w razie wątpliwości, uzgodnić z prowadzącym zajęcia

9.5.1 Lista głównych elementów interfejsu

okien, stron, aktywności (Android)

Kontrolka wprowadzania tekstu

- logowanie - wprowadzenie loginu oraz hasła
- rejestracja - wprowadzenie loginu, hasła, email oraz wybranie pomiędzy użytkownikiem premium a podstawowym.
- wprowadzania danych do narzędzi służących do liczenia.

Komponenty nawigacji

- przemieszczanie się pomiędzy narzędziami strony internetowej

Komponenty informacyjne

- wydzielona część strony służąca do przekazywania aktualnych informacji dotyczących aktualności na stronie internetowej.

Przyciski

- element służący do wykonywania akcji.

Okienko

- odpowiedzialne za wyświetlanie wykresu

Hiperłącza

- odnośniki do zewnętrznych stron.

9.5.2 Przejścia między głównymi elementami

Za przejścia pomiędzy elementami strony internetowej będzie służył komponent nawigacyjny. Osobny przycisk znajdujący się w górnym prawym rogu do przejścia do zarządzania kontem.

przycisk zarządzania kontem -> wyloguj

przycisk zarządzania kontem->ustawienia

przycisk zarządzania kontem->zarządzaj kontem

przycisk zarządzania kontem->zakup premium

komponent nawigacyjny->strona główna

komponent nawigacyjny->materiały

komponent nawigacyjny->E-lekcje

komponent nawigacyjny->kalkulatory rozwiązywania zadań

komponent nawigacyjny->Zadania

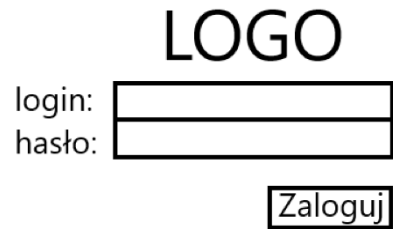
logowanie->strona główna

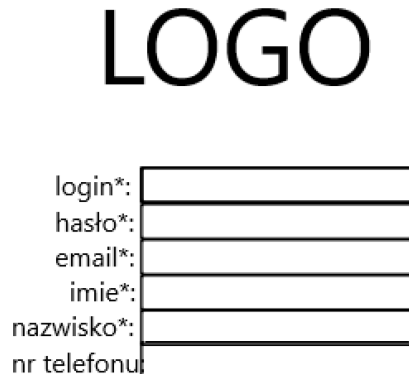
rejestracja->logowanie->strona główna


9.5.3 Projekty szczegółowe poszczególnych elementów


każdy element od nowej strony z następującą minimalną zawartością:

- numer – ID elementu
- nazwa – np. formularz danych produktu
- projekt graficzny – wystarczy schemat w narzędziu graficznym lub zrzut ekranu – z przykładowymi informacjami (nie pusty!!!)
- opcjonalnie:
- opis – dodatkowe opcjonalne informacje o przeznaczeniu, obsłudze – jeśli nazwa nie będzie wystarczająco czytelna
- wykorzystane dane – jakie dane z bazy danych są wykorzystywane
- opis działania – tabela pokazująca m.in. co się dzieje po kliknięciu przycisku, wybraniu opcji z menu itp.

numer	1
nazwa	logowanie
projekt graficzny	 <p>The graphic shows a login form layout. At the top is the word 'LOGO' in large, bold, black capital letters. Below it, the text 'login:' is followed by a rectangular input field. Below that, the text 'hasło:' is followed by another rectangular input field. At the bottom right, there is a rectangular button with the text 'Zaloguj' inside it.</p>

numer	2
nazwa	rejestracja
projekt graficzny	 <p>The graphic shows a registration form layout. At the top is the word 'LOGO' in large, bold, black capital letters. Below it, there is a list of labels followed by input fields: 'login*', 'hasło*', 'email*', 'imie*', 'nazwisko*', and 'nr telefonu'. Each label is aligned to the left of its corresponding input field.</p>

numer	3
nazwa	komponent nawigacyjny
projekt graficzny	
opis	<p>Panel ten jest elementem odpowiedzialnym do poruszania się po stronie internetowej.</p> <p>Znajduje się zawsze po lewej stronie ekranu.</p>

numer	4
nazwa	przycisk zarządzania kontem
projekt graficzny	
opis	<p>Element, który daje możliwość przejścia do panelu danych użytkownika oraz ustawień profilu użytkownika.</p> <p>Znajduje się zawsze w górnym prawym rogu.</p>

numer	5
-------	---

nazwa	Ogólne informacje
projekt graficzny	<div data-bbox="873 296 1339 606" style="border: 2px solid black; padding: 10px; margin: 10px auto; width: fit-content;"> <p>Informacja nr 6 data v</p> <p>Artykuł nr 3 data v</p> <p>Informacja nr 5 data v</p> <p>Informacja nr 4 data v</p> <p>Artykuł nr 2 data v</p> </div>
opis	<p>Element służący do przekazywania informacji użytkownikowi.</p> <p>Znajduje się na głównej stronie</p>

9.6 Procedura wdrożenia

- Zatrudnienie nauczycieli
- Zapoznanie nauczycieli z aplikacją
- Zakup serwera
- Umieszczenie aplikacji na serwer
- Wypuszczenie aplikacji na rynek.
- Reklamowanie produktu.
- Współpraca z uczelniami.

10 Dokumentacja dla użytkownika

11 Podsumowanie

11.1 Szczegółowe nakłady projektowe członków zespołu

Zadanie	Igor Stodolny	Marcin Włodarczyk
Stworzenie harmonogramu	50%	50%
Wprowadzenie do dokumentacji	50%	50%
Opracowanie charakterystyki ogólnej	70%	30%
Wymagania funkcjonalne	0%	100%
Wymagania нефunkcjonalne	100%	0%
Stworzenie przypadków użycia	0%	100%
Zarządzanie projektem	60%	40%
Zarządzanie ryzykiem	50%	50%
Opracowanie scenariuszy testowych	100%	0%
Projekt techniczny - część opisowa	0%	100%
Diagram klas	50%	50%
Diagramy UML - pozostałe	50%	50%
Opisanie wzorców projektowych	50%	50%
Zaprojektowanie bazy danych	0%	100%
Zaprojektowanie interfejsu graficznego	100%	0%
Opracowanie procedury wdrożenia	50%	50%
Nakład pracy w całości dokumentacji		
Ogólnie	50%	50%

12 Inne informacje

przydatne informacje, które nie zostały ujęte we wcześniejszych punktach