**Platforma kursów**

Platforma e-learningowa a’la moodle

Skład grupy:

Filip Domachowski

Paweł Godlewski

Wojciech Marciniak

Link do repozytorium:

<https://github.com/Pagan254081/inzopr>

**Scenariusze**

**Przypadek nr 1 : Rejestracja w systemie**

1. Użytkownik wypełnia formularz rejestracji

2. Zatwierdza wprowadzone dane

3. Potwierdza prawdziwość podanej skrzynki pocztowej klikając w link

otrzymany na podany adres e-mail

**Przypadek nr 2 : Zapis na kurs**

1. Wybiera kurs z listy

2. Klika zapisz mnie

3.1. Jeśli kurs jest bezpłatny i są wolne miejsca zostanie zapisany

3.2. Jeśli kurs jest płatny i są wolne miejsca przejdź do panelu płatności

**Przypadek nr 3 : Aktualizacja szczegółów kursu**

1. Admin loguje się do systemu

2. Wybiera kurs do aktualizacji

3. Aktualizuje dane kursu

4. Zatwierdza zmiany

**Przypadek nr 4 : Usunięcie konta**

1. Zalogowany użytkownik wchodzi na swój profil

2. Przechodzi na zakładkę Ustawienia

3. Wybiera usuń konto i zatwierdza wybór

**Przypadek nr 5 : Panel płatności**

1. Zalogowany użytkownik wybiera sposób płatności

2. Dokonuje płatności

**Przypadek nr 6 : Dodawanie zasobów do kursu**

1. Właściciel kursu wybiera dodaj zasób

2. Wybiera typ zasobu

3. Uzupełnia wymagane dane

4. Zatwierdza dodanie zasobu

**Opis**

Singleton Platform to serce tego systemu.

My jako klient (User) mamy dostęp jedynie do funkcjonalności udostępnianej przez klasę Platform, tak więc wszystkie operacje mogą korzystać tylko i wyłącznie z udostępnionego interfejsu.

Taki podział umożliwia dwutorowy rozwój oprogramowania:

Naszym frontendem byłaby klasa User i jej rozszerzenia.

Backendem klasa Platform i przez nią używane.

Co sprawiłoby, że dodanie funkcjonalności do frontendu jednocześnie nie miałoby wpływu na działanie backendu i vice versa.

Do tworzenia kursów i zasobów wykorzystane są klasy CourseCreator i ResourceCreator zgodnie ze wzorcem FactoryMethod

Tak mogłaby wyglądać przykładowa metoda na dodanie kursu:

bool User::addCourse(){

bool wynik = getPlatform().addCourse();

return wynik;

}