**Wojskowa Akademia Techniczna**

**Sprawozdanie**

**Programowanie zdarzeniowe**

**Projekt**

**Temat**: System korygujący ruchy myszki w celu skorygowania skupienia strzału w grach FPS

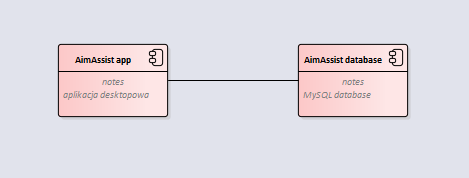
Prowadzący: por. mgr inż. Dawid Bugajewski

Wykonał: Rafał Geresz

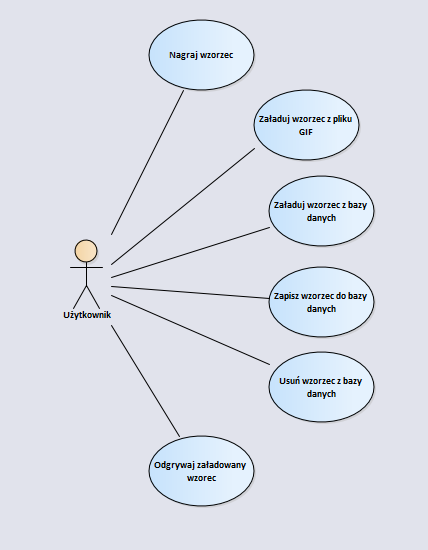
Nr indeksu: 63013

Grupa: I5B4S1, I\*PZ

1. **Diagram architektury**

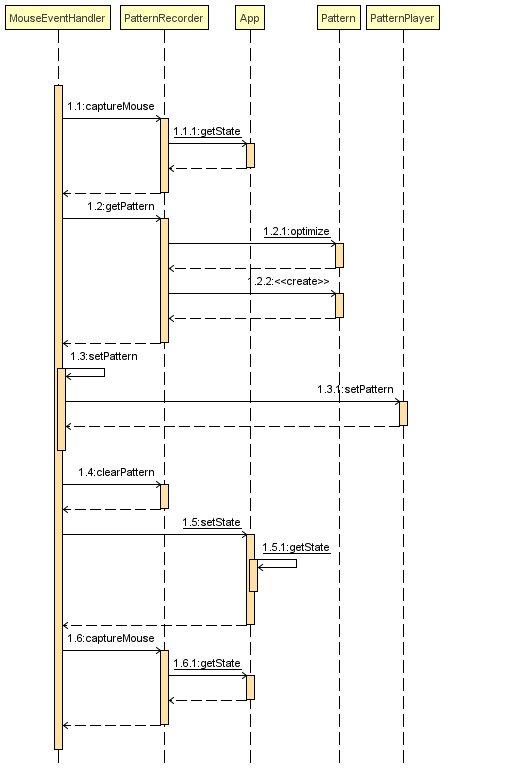


1. **Przypadki użycia**

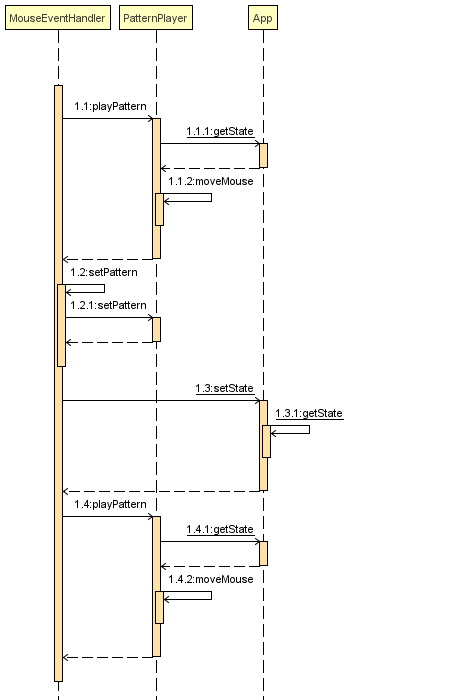


1. **Diagramy sekwencji**

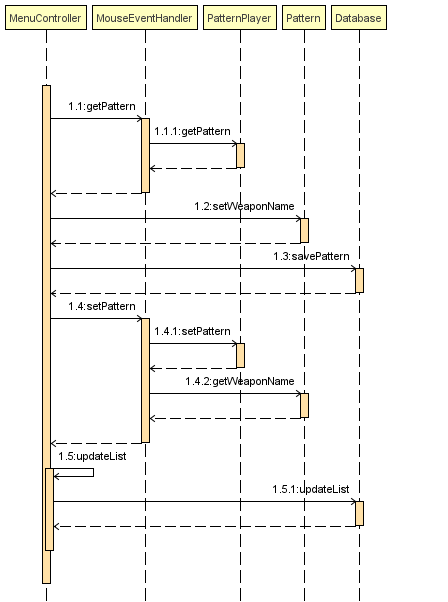
* Nagraj wzorzec



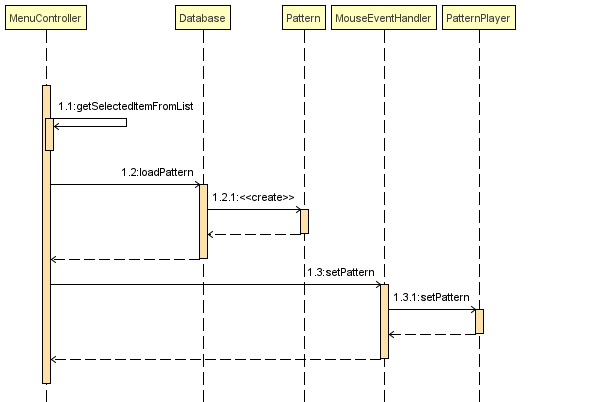
* Odtwórz wzorzec



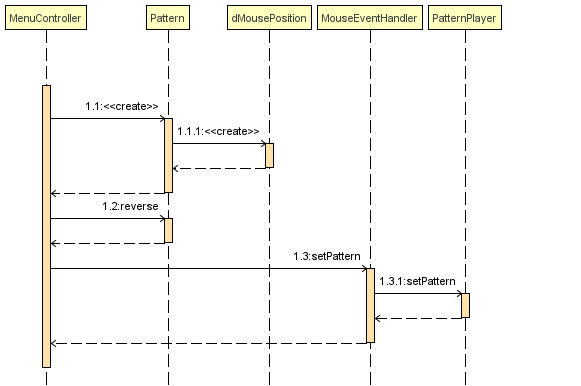
* Zapisz wzorzec w bazie



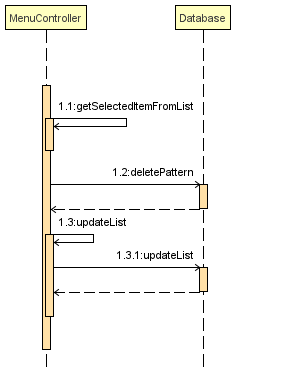
* Załaduj wzorzec z bazy



* Załaduj wzorzec z pliku GIF



* Usuń wzorzec z bazy



1. **Przypadki testowe**

**Tytuł:**

Nagraj wzorzec

**Kroki:**

1. Naciśnij F1 na klawiaturze.
2. Naciśnij LPM.
3. Poruszaj kursorem myszy.
4. Puść LPM.

**Oczekiwany rezultat:**

Ruchy myszy wykonane pomiędzy naciśnięciem a puszczeniem LPM zostaną załadowane do aplikacji oraz stan aplikacji zmieni się na PLAY.

**Tytuł:**

Odtwórz wzorzec

**Kroki:**

1. Naciśnij F2 na klawiaturze.
2. Naciśnij LPM.

**Oczekiwany rezultat:**

Kursor myszy zacznie się poruszać zgodnie ze załadowanym wzorem.

**Tytuł:**

Zapisz wzorzec w bazie danych

**Kroki:**

1. Naciśnij przycisk „Save Pattern” w oknie aplikacji.
2. Podaj nazwę wzorca w wyświetlonym oknie oraz wciśnij przycisk „OK”.

**Oczekiwany rezultat:**

Zostanie wyświetlone okno z informacji o pomyślnym zapisaniu wzorca załadowanego w aplikacji oraz wzorzec zostanie zapisany w bazie danych.

**Tytuł:**

Załaduj wzorzec z bazy danych

**Kroki:**

1. Wybierz wzorzec z listy w oknie aplikacji.
2. Naciśnij przycisk „Load pattern from database” w oknie aplikacji.

**Oczekiwany rezultat:**

Zostanie wyświetlone okno z informacji o pomyślnym załadowaniu wzorca z bazy danych oraz wybrany wzorzec zostanie załadowany do aplikacji.

**Tytuł:**

Załaduj wzorzec z pliku GIF

**Kroki:**

1. Naciśnij przycisk „Load pattern from GIF” w oknie aplikacji.
2. Wybierz plik .gif w wyświetlonym oknie oraz zatwierdź.
3. Podaj szybkostrzelność broni w strzałach na minutę w wyświetlonym oknie oraz wciśnij przycisk „OK”.

**Oczekiwany rezultat:**

Zostanie wyświetlone okno z informacji o pomyślnym załadowaniu wzorca z pliku GIF oraz wybrany wzorzec zostanie załadowany do aplikacji.

**Tytuł:**

Usuń wzorzec z bazy danych

**Kroki:**

1. Wybierz wzorzec z listy w oknie aplikacji.
2. Naciśnij przycisk „Delete pattern” w oknie aplikacji.
3. Naciśnij przycisk „OK” w wyświetlonym oknie aby potwierdzić.

**Oczekiwany rezultat:**

Zostanie wyświetlone okno z informacji o pomyślnym usunięciu wzorca z bazy danych oraz wybrany wzorzec zostanie usunięty z bazy danych.

**Tytuł:**

Zatrzymaj odtwarzanie wzorca

**Kroki:**

1. Naciśnij przycisk F3 na klawiaturze.

**Oczekiwany rezultat:**

Stan aplikacji zmieni się na „STOPPED” oraz naciśnięcie LPM nie będzie inicjowało żadnych akcji związanych ze wzorcami.

**5. Podsumowanie**

Najlepszym sposobem na wyeliminowanie odrzutu w grze byłoby oczywiście nadpisywanie pamięci procesu danej gry, ale nie to było tematem projektu oraz takie rozwiązanie jest mocno narażone na wykrycie przez systemy przeciwko oszustwom. Jednakże moja aplikacja pozwala zdobyć małą przewagę bez tego ryzyka, lecz nie jest zbyt wygodna w obsłudze. Ponadto nie udało mi się naprawić 2 błędów w działaniu aplikacji:

1. Nagrywania wzorca podczas gry (testowałem aplikacje w grze Counter Strike: Global Offensive). Występuje on, ponieważ gdy aktywne jest okno gry kursor jest cały czas w jego środku a ruchy myszy powodują zmianę kątów widzenia gracza.
2. Dodawanie wzorca z pliku GIF działa tylko w wypadku konkretnych plików GIF (np. ze strony <http://csgoskills.com/academy/spray-patterns/>). Jednak nie w każdym przypadku udaje się poprawnie rozpoznać wzorzec.

Gdybym miał od początku napisać tego rodzaju aplikację prawdopodobnie nie wybrał bym języka Java tylko C++. Związane jest to z łatwiejszą obsługą zdarzeń systemowych z pomocą bibliotek C++.

Aplikację można by rozwinąć o kilka przydatnych funkcji:

* Automatyczne wykrywanie wybranej broni oraz dopasowywanie wzorca.
* Algorytm szukający optymalnego wzorca na podstawie obrazów z gry.