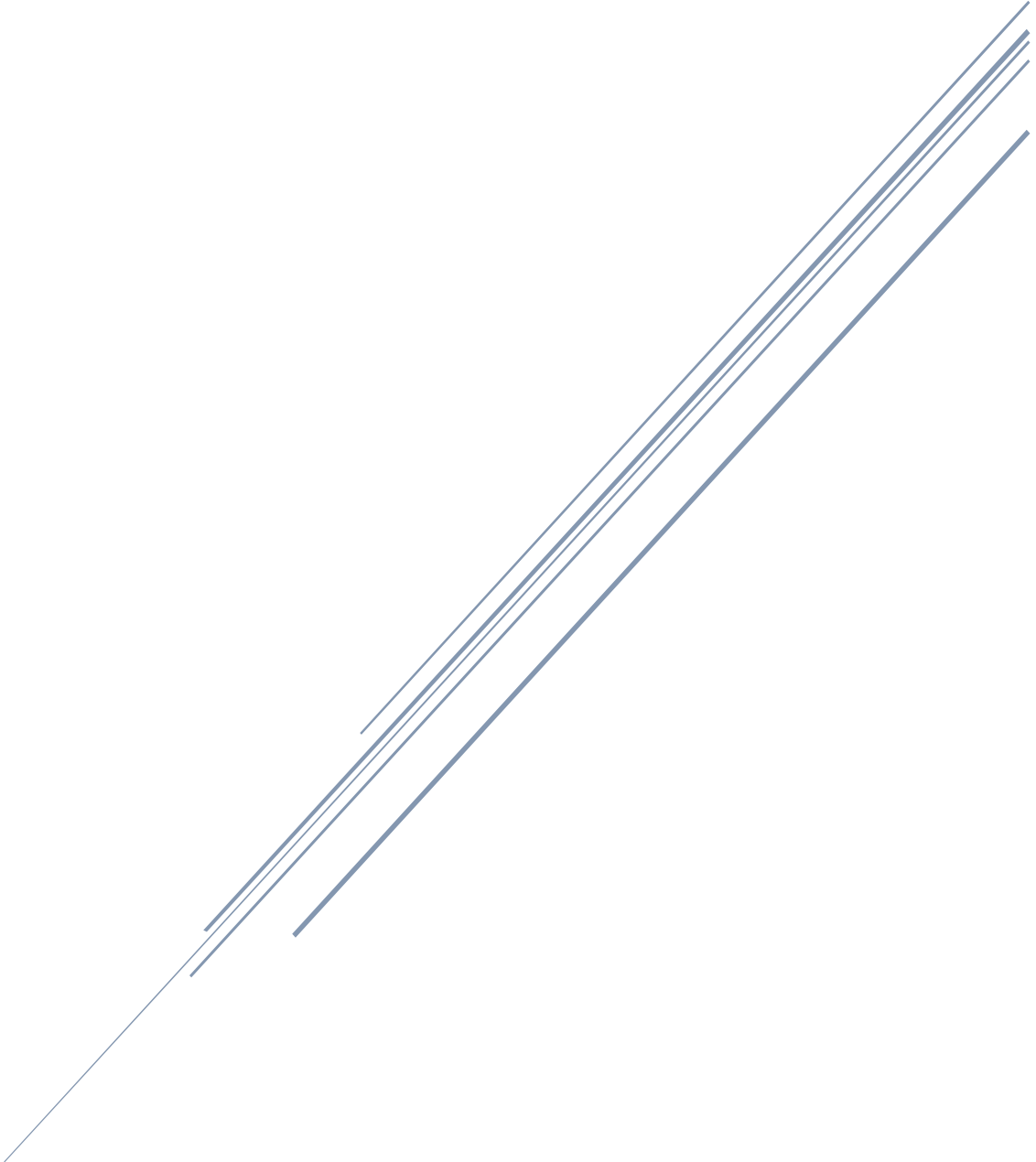


PODZIAŁ PRACY

Coin Hunter



Uniwersytet Jagielloński, Wydział Matematyki i Informatyki
Inżyniera Oprogramowania

Spis treści

1. Wstęp.....	2
2. Grupy i struktura pracy.....	3
2.1. Serwer oraz strona internetowa.....	3
2.1.1. Aleksandra Wnuk.....	3
2.1.2. Julia Pająk.....	3
2.2. Dokumentacja.....	4
2.2.1. Agata Przybyła.....	4
2.2.2. Mikołaj Oberda.....	4
2.3. Gra oraz testy.....	5
2.3.1. Dawid Baran	5
2.3.2. Tomek Piękosz	6

1. Wstęp

Niniejszy dokument przedstawia podział na zespoły, które miał miejsce podczas tworzenia projektu „Coin Hunter” oraz wykaz zadań wykonywanych przez poszczególnych członków znajdujących się w obrębie tychże grup.

2. Grupy i struktura pracy

2.1. Serwer oraz strona internetowa

2.1.1. Aleksandra Wnuk

1. UX/UI design strony,
2. Restrukturyzacja i zarządzanie kolekcjami w bazie danych,
3. Implementacja oraz refactoring w HTML oraz CSS:
 - Profilu użytkownika,
 - Rankingu,
 - Forum,
4. Utworzenie mechanizmu reportowania błędów przez użytkowników (moduł nodemailer),
5. Utworzenie i redagowanie pliku README i instrukcji uruchomienia lokalnego,
6. Dodanie ograniczeń do systemu logowania użytkowników,
7. Implementacja szyfrowania haseł (bcrypt),
8. Restrukturyzacja systemu rankowania użytkowników bazującego na ilości monet i przebytej odległości,
9. Restrukturyzacja i implementacja kodu gry zgodnie z zasadami OOP i SOLID, włączenie wzorców projektowych,
10. Przygotowanie i wygłoszenie prezentacji,
11. Konsultacje i współpraca z zespołem ds. dokumentacji odnośnie do struktury kodu i zastosowanych rozwiązań,
12. Stworzenie dokumentu „Opis klas”,
13. Zaprojektowanie diagramu klas – Gra,
14. Nadzór nad architekturą gry.

2.1.2. Julia Pająk

1. UX/UI design strony,
2. Implementacja w HTML oraz CSS:
 - Profilu użytkownika,
 - Sklepu (infrastruktura i logistyka),
 - Katalogu użytkowników (mechanizm wyszukiwania użytkowników po loginach),
 - Zgłaszania błędów,
 - Strony gry,
 - Ogólnego layoutu strony,
3. Utworzenie serwera, bazy danych oraz połączenia między nimi,
4. Uwierzytelnianie użytkownika, system logowania i rejestracji (włączenie oprogramowania passport.js),
5. Katalog wszystkich użytkowników w bazie, wyszukiwanie użytkownika po username,

6. Utworzenie i redagowanie pliku README i instrukcji uruchomienia lokalnego,
7. Komunikacja gry z bazą danych, aktualizowanie danych użytkownika po zakończonej grze,
8. Implementacja systemu rankowania użytkowników bazującego na ilości monet i przebytej odległości,
9. Restrukturyzacja i implementacja kodu gry wraz z dostosowaniem silnika gry zgodnie z zasadami OOP i SOLID, włączenie wzorców projektowych,
10. Przygotowanie prezentacji,
11. Stworzenie dokumentu „Opis klas”,
12. Nadzór nad architekturą gry.

2.2. Dokumentacja

2.2.1. Agata Przybyła

1. Konsultacja i współpraca z zespołem odpowiedzialnym za serwer i stronę internetową oraz z zespołem odpowiedzialnym za grę oraz testy,
2. Branie udziału w dyskusji na temat założeń projektu,
3. Branie udziału w konstruowaniu wymagań aplikacji,
4. Prezentacja wymagań na zajęciach,
5. Utworzenie i opracowanie plików dokumentacji:
 - Dokument wizji i zakresu,
 - Wymagania użytkownika,
6. Utworzenie diagramów:
 - Diagram przypadków użycia,
 - Diagram czynności,
 - Diagram sekwencji,
7. Testowanie aplikacji poprzez jej używanie i zgłaszanie ewentualnych błędów.

2.2.2. Mikołaj Oberda

1. Redagowanie i nadzór nad dokumentacją,
2. Konsultacja i współpraca z zespołem odpowiedzialnym za serwer i stronę internetową oraz z zespołem odpowiedzialnym za grę oraz testy,
3. Stworzenie szablonu dokumentacji,
4. Analiza i opis wymagań funkcjonalnych oraz нефункциональных,
5. Stworzenie następujących dokumentów m.in. na podstawie informacji uzyskanych od członków innych zespołów:
 - Podział pracy,
 - Instrukcja obsługi,
 - Interfejs użytkownika,
 - Metodologia, architektura i wzorce,
 - Realizacja wymagań,
 - Software Requirements Specification,
 - Wykorzystane technologie,

- Grafik wykonanych zadań,
- 6. Zredagowanie i reedycja dokumentacji otrzymanej od członków innych zespołów:
 - Dokumentacja testowa,
 - Instrukcja uruchomienia lokalnego,
 - Diagram klas – Gra,
 - Opis klas,
- 7. Zreferowanie pracy zespołu od spraw dokumentacji,
- 8. Ostateczna poprawa dokumentacji wraz z kierownikiem projektu.

2.3. Gra oraz testy

2.3.1. Dawid Baran

1. Wprowadzanie kolejnych animacji:
 - Serca,
 - Skok,
 - Moneta,
 - Flash,
2. Udoskonalanie poruszania się postaci (poprawa mechaniki),
3. Naprawa błędów animacji,
4. Naprawa błędów implementacyjnych,
5. Naprawa błędów podczas integracji,
6. Poprawa interakcji postaci z pozostałymi elementami gry,
 - Reakcja na kolizje z monetą,
 - Reakcja na kolizje z przeciwnikiem,
7. Refaktoryzacja kodu:
 - Dostosowanie szkieletu do rozgrywki,
 - Wgranie animacji,
 - Wgranie mechaniki,
 - Poprawa czytelności kodu,
8. Utworzenie środowiska do sprawnej komunikacji członków zespołu,
9. Testowanie jednostkowe,
10. Wykonanie części testów systemowych,
11. Wykonanie części testów integracyjnych,
12. Sporządzenie dokumentacji testowej,
13. Korespondencja z prowadzącą ćwiczenia,
14. Częściowe zarządzanie projektem,
 - Wprowadzanie własnych pomysłów,
 - Konsultacja zastosowanych rozwiązań,
15. Kontrola plików dokumentacyjnych i eliminacja błędów,
16. Konsultacje i współpraca z zespołem ds. dokumentacji odnośnie do struktury kodu i zastosowanych rozwiązań,
17. Zreferowanie pracy zespołu programistów gry,
 - Wygłoszenie z jakich technologii korzystamy i dlaczego.

2.3.2. Tomek Piękosz

1. Pomysł na projekt,
2. Konsultacje i współpraca z zespołem ds. dokumentacji odnośnie do struktury kodu i zastosowanych rozwiązań,
3. Design gry oraz jej interfejsów w HTML i CSS,
4. Dodanie animacji wszystkich elementów gry,
5. Szukanie optymalnych parametrów mechaniki dla jak najlepszej rozgrywki,
6. Implementacja całej gry w JavaScript'cie,
7. Powierzchowne testowanie rozgrywki,
8. Restrukturyzacja kodu gry na OOP,
9. Poprawienie błędów oraz niedociągnięć w kodzie po testach.