**Sprawozdanie z projektu z przedmiotu Narzędzia Pracy Grupowej**

**Gra Mastermind**

Obraz zawierający logo, Czcionka, tekst, symbol

Zawartość wygenerowana przez AI może być niepoprawna.

Rok 2025

Akademia Górniczo-Hutnicza, WEAIiIB, Automatyka i Robotyka, semestr 2, Gr 1

Wykonane przez:

Szymon Cempura, Urszula Cisek, Krzysztof Ceklarz, Mateusz Cieślak, Leon Bukowy

**1. Udostępnienie zasobów projektu**

Projekt został zrealizowany na platformie GitHub.  
**Link do repozytorium:** https://github.com/SzymCe2/Mastermind

W repozytorium znajduje się:

* Kod źródłowy aplikacji, podzielony na moduły: generator kodu, logika porównania, walidacja, interfejs użytkownika, integracja.
* Dokumentacja techniczna i opis modułów w plikach README.
* Tablice zadań i raporty z postępów w zakładce Projects.
* Raporty zebranych spotkań zespołu (folder /reports).

**2. Opis organizacji pracy**

Projekt był realizowany zgodnie z metodyką Scrum w uproszczonej formie. Praca odbywała się w 2 sprintach.

**Narzędzia używane podczas projektu:**

* GitHub – zarządzanie kodem i dokumentacją, zarządzanie repozytorium, tracking commitów i gałęzi
* messenger – komunikacja zespołu, organizacja spotkań i standupów
* GitHub Projects – zarządzanie backlogiem i zadaniami
* Pycharm – środowisko programistyczne

Artefaktami projektu były: backlog produktu, repozytorium Git z kodem, wiki z dokumentacją oraz raporty standupów.

**3. Przebieg sprintów**

**Sprint 1**

* **Cel:** Stworzenie wstępnie działającej wersji gry Mastermind
* **Role:**
  + Krzysiek – generator kodu
  + Mateusz – logika porównania
  + Urszula – prace nad interfejsem (tekstowym)
  + Leon – testy jednostkowe
  + Szymon – koordynacja repozytorium i integracja modułów, oraz kod na podpowiedzi
  + Leon – walidacja danych
* **Kamienie milowe:**
  + Wstępny kod gotowy
  + Dodanie podpowiedzi do gry
  + Ukończenie działającej gry
* **Czasy realizacji:** Sprint trwał od 14 maja do 30 maja
* **Zebrania:** 1 zebranie 15 maja

**Sprint 2**

* **Cel:** ulepszenie gry o elementy graficzne interfejsu, jak i dodatkowe poziomy trudności i funkcje
* **Role:**
  + Krzysiek – ulepszenie generatora kodu
  + Mateusz – wzmocnienie logiki porównań
  + Urszula – prace nad interfejsem (konsola)
  + Szymon – koordynacja repozytorium i integracja modułów
  + Leon – walidacja danych
* **Kamienie milowe:**
  + Wstępny kod generatora kodu gotowy
  + Podstawowy interfejs konsoli uruchomiony
* **Czasy realizacji:** Sprint trwał od 31 maja do 23 czerwca
* **Zebrania:** 1 zebranie 5 czerwca

**4. Backlog produktu**

Początkowy backlog zawierał następujące wymagania:

* Moduł generowania losowego kodu z opcjami konfiguracyjnymi (długość, unikalność, kolory/liczby)
* Moduł porównujący kod użytkownika z tajnym kodem i generujący wynik
* Moduł pozwalający na podpowiedzi kodu
* Walidacja danych wejściowych
* Interfejs użytkownika (tekstowy)
* Integracja modułów i zarządzanie pętlą gry
* Dokumentacja i testy

W trakcie realizacji backlog ewoluował, zostały dodane dodatkowe testy jednostkowe oraz usprawnienia interfejsu.

**5. Wkład indywidualny członków zespołu**

**Krzysztof Ceklarz – Backend (Generator kodu)**

**https://github.com/Krzysztof-droid**

**Zrealizowane zadania:**

*Zadania projektowe:*

• Zaprogramowanie generatora kodu

*Zadania programistyczne:*

• Wykonanie Code Review dla członków zespołu

• Zaprojektowanie oraz trzykrotna modyfikacja i optymalizacja generatora kodu, który umożliwia: o konfigurację długości kodu, o wybór trybu: kolory lub liczby, o określenie, czy znaki w kodzie mogą się powtarzać, o wybór poziomu trudności: łatwy, średni, trudny

• Finalna wersja generatora została dopracowana z myślą o elastyczności oraz łatwym wyborze poziomu trudności

*Zadania organizacyjne:*

• Udział w trzech spotkaniach zespołowych, w tym aktywne omawianie kierunku rozwoju projektu i podziału obowiązków. Zadania dokumentacyjne:

• Edycja i uzupełnienie dokumentacji projektu, pliku README w zakresie opisu działania generatora i zasad gry. Parametry oceny pracy

• Rola: stworzenie i rozwój funkcji generatora kodu z parametrami: o długość kodu, o unikalność znaków, o trybie kolor/liczby, o poziom trudności (łatwy/średni/trudny).

• Repozytorium: 5 commitów (w tym 3 znaczące modyfikacje pliku generator\_kodu.py), o praca w gałęzi master, o aktywność przez 3 dni.

• Czas pracy: około 8 godzin

• Wkład procentowy w projekt: 17,5%

**Mateusz Cieślak – Backend (Logika porównania)**

* https://github.com/MatCieslak

[matcieslak@student.agh.edu.pl](mailto:matcieslak@student.agh.edu.pl)

* Zaimplementowanie logiki porównującej, która miała sprawdzać poprawność odpowiedzi gracza i w wypadku nieprawidłowej kombinacji zwracać ilość trafionych miejsc oraz trafionych kolorów na złych miejscach.
* Branie udziału w konwersacji dotyczącej projektu na platformie Messenger
* Czas pracy: 6 godzin; Ilość commitów: 3
* Wkład procentowy: 15%

**Urszula Cisek – Interfejs**

[**https://github.com/ulllaaa**](https://github.com/ulllaaa)

|  |  |  |
| --- | --- | --- |
| Aspekt | Parametry | Wkład |
| Role | **Wymienić** | **Urszula Cisek, urszulacisek@student.agh.edu.pl** |
| Kodowanie | **Liczba linii kodu** | **172** |
|  | **Funkcje (wymienić)** | **Stworzenie interfejsu (GUI) z użyciem biblioteki tkinter, który jest połączony z plikami odpowiedzialnymi za generowanie kodu do gry i porównywania. Interfejs obsługuje przebieg gry – zliczanie prób, wyświetlanie podpowiedzi i komunikatów o wygranej lub przegranej. Dodatkowo umożliwia wybór poziomu trudności gry, rozpoczęcie nowej gry, wybór kolorów w trakcie gry oraz zakończenie zgadywania. Również daje informacje zwrotna o błędach w przypadku braku wprowadzenia danych - braku wyboru poziomu trudności.** |
| Repozytorium | **Liczba commit-ów** | **W głównym repozytorium projektu są tylko 2 commity utworzone bezpośrednio przeze mnie, ponieważ pracowałam nad swoją częścią w osobnym forku repozytorium. Utworzyłam 3 commity w swoim repozytorium, a następnie koordynator wrzucił moje pliki do głównego repozytorium.** |
|  | **Liczba utworzonych gałęzi** | **3 - w sforkowanym repozytorium.** |
|  | **Gałąź (używana – nazwa)** | **Gałąź Master w repozytorium projektu, gałąź patch-1 w sforkowanym repozytorium.** |
|  | **Liczba połączonych gałęzi** | **1** |
|  | **Liczba dni aktywności GIT** | **3 dni** |
| Dokumentowanie | **Liczba standup-ów** | **1 raport na Githubie, zgłaszanie problemów i dyskusja nad projektem w konwersacji w aplikacji Messenger.** |
|  | **Opisy na Wiki** | **Nie tworzyłam osobnych wpisów w Wiki.** |
| Aktywność | **Liczba zrealizowanych zadań** | **Stworzenie całości interfejsu, w tym: zaprojektowanie wyglądu, integracja GUI z generatorem i porównywarką, testowanie gry i wprowadzanie poprawek, aby zwiększyć czytelność.** |
|  | **Szacowana liczba godzin** | **16h** |
|  | **Ocena procentowego wkładu** | **17.5%** |

**Szymon Cempura – Koordynacja**

Mail: [scempura@student.agh.edu.pl](mailto:scempura@student.agh.edu.pl)

Link na githubie: <https://github.com/SzymCe2>

***Zrealizowane:***

*Zadania Projektowe:*

- Pomoc w ustaleniu zakresu funkcjonalności gry

- Udział w spotkaniu

- Propozycja mechanizmu podpowiedzi

- Propozycja mechanizmu różnej długości kodu do zgadnięcia

*Zadania Programistyczne:*

- Utworzenie projektu na Githubie (<https://github.com/SzymCe2/Mastermind>)

- Scalanie kodu napisanego przez resztę grupy

- Code Review dla osób z grupy

*Zadania Organizacyjne:*

- Koordynacja pracy innych osób

- Organizacja spotkania grupy

*Zadania dokumentacyjne:*

- Stworzenie Sprawozdania z Projektu

- Stworzenie dokumentacji do projektu (Readme)

- Stworzenie raportu ze spotkania

- Utworzenie w projects backlogu dla reszty osób z grupy

Dodawałem commity lokalnie z komputera gdzie mam ustawioną nazwę Szyme, i można je znaleźć w projekcie pod tą nazwą (w <https://github.com/SzymCe2/Mastermind/commits/master/>)

Parametry oceny pracy:

Rola: koordynator oraz filnalizacja kodu

Repozytorium:

- X commitów

- 1 gałąź master

- 5 dni aktywności git

- 1 spotkanie grupowe

- 1 zrealizowane zadanie

- około 20 godzin pracy

Wkład procentowy: 30%

**Leon Bukowy – Backend (Walidacja i testy)**

[**https://github.com/mapilogo**](https://github.com/mapilogo)

*Zadania Projektowe:*

* Walidacja wejścia i przygotowanie testów jednostkowych
* Aktywna komunikacja z zespołem dotycząca wprowadzonych rozwiązań

*Zadania Programistyczne:*

* Implementacja testów jednostkowych dla klasy porównywania oraz generowania kodu przy użyciu biblioteki ***unittest***.
* Modyfikacja testów w zależności od zmian mechanizmu wprowadzanych przez zespół.
* Dzięki dobrej komunikacji testuje się wyłącznie używane i istotne części programu, co optymalizuje debugging w razie dalszego rozwijania projektu.

*Podsumowanie pracy:*

* Ilość commitów dla gałęzi programistycznej : **3**
* Czas pracy: **8h**
* Udział: **15%**

Tabela podsumowująca wkład zespołu:

| **Aspekt** | **Krzysztof** | **Mateusz** | **Urszula** | **Leon** | **Szymon** |
| --- | --- | --- | --- | --- | --- |
| zadanie | Generator kodu | Logika porównania | Interfejs | Walidacja i testy | Koordynacja |
| funkcje |  |  |  |  |  |
| Liczba commitów |  |  |  |  |  |
|  |  |  |  |  |  |
| Dni aktywności git |  |  |  |  | 5 |
| Udział % | 17,5% | 17,5% | 17,5% | 17,5% | 30% |