

# DOKUMENT PROJEKTU

Wersja dokumentu: 1.1

Data utworzenia: 19.03.2025

Data ostatniej aktualizacji: 19.03.2025

## Gra internetowa Codenames

Organizacja i infrastruktura projektu

**Redaktor:** Zuzanna Nowak, s193165

**Współautorzy:** Agata Domasik, 193577  
Jakub Walasik, s193650  
Adam Chabraszewski, s193373

**Liczba stron:** 24





---

## Spis treści

---

<b>1</b>	<b>Wprowadzenie - o dokumencie</b>	<b>5</b>
1.1	Cel dokumentu . . . . .	5
1.2	Odbiorcy . . . . .	5
1.3	Terminologia . . . . .	5
<b>2</b>	<b>Opis projektu</b>	<b>7</b>
2.1	Nazwa projektu . . . . .	7
2.2	Adresowany problem . . . . .	7
2.3	Obszar zastosowania . . . . .	7
2.4	Rynek . . . . .	7
2.5	Interesariusze . . . . .	8
2.6	Użytkownicy i ich potrzeby . . . . .	8
2.7	Cel i zakres projektu . . . . .	8
2.8	Ograniczenia . . . . .	8
2.9	Inne współpracujące systemy . . . . .	9
2.10	Harmonogram . . . . .	9
2.10.1	Termin . . . . .	9
2.10.2	Główne etapy projektu . . . . .	9
<b>3</b>	<b>Interesariusze i użytkownicy</b>	<b>11</b>
3.1	Interesariusze . . . . .	11
3.2	Użytkownicy końcowi . . . . .	11
3.3	Klasyfikacja i krótki opis interesariuszy . . . . .	11
3.3.1	Interesariusze wewnętrzni . . . . .	11
3.3.2	Interesariusze zewnętrzni . . . . .	11
<b>4</b>	<b>Zespół</b>	<b>13</b>
4.1	Skład zespołu . . . . .	13
4.2	Kompetencje i umiejętności członków zespołu . . . . .	13
4.3	Zakresy odpowiedzialności . . . . .	15
4.4	Model pracy . . . . .	15
4.5	Dane kontaktowe . . . . .	15

<b>5</b>	<b>Komunikacja w zespole i z interesariuszami</b>	<b>17</b>
5.1	Harmonogram spotkań . . . . .	17
5.2	Miejsca spotkań . . . . .	17
5.3	Narzędzia komunikacji . . . . .	17
5.4	Komunikacja z interesariuszami zewnętrznymi . . . . .	17
5.5	Współpraca z opiekunem projektu . . . . .	17
<b>6</b>	<b>Współdzielenie dokumentów i kodu</b>	<b>19</b>
6.1	Metody wymiany dokumentów i kodu . . . . .	19
6.2	Repozytorium projektu . . . . .	19
6.2.1	Administracja repozytorium . . . . .	19
6.3	Zarządzanie dokumentacją . . . . .	19
6.3.1	Osoba odpowiedzialna . . . . .	19
6.3.2	Konwencje nazewnictwa . . . . .	20
6.3.3	Szablony dokumentów . . . . .	20
6.4	System wersjonowania . . . . .	20
6.4.1	Wersjonowanie kodu . . . . .	20
6.4.2	Wersjonowanie dokumentacji . . . . .	20
<b>7</b>	<b>Narzędzia</b>	<b>21</b>
7.1	Narzędzia komunikacyjne . . . . .	21
7.2	Narzędzia do zarządzania dokumentacją . . . . .	21
7.3	Narzędzia do zarządzania projektem . . . . .	21
7.4	Narzędzia developerskie . . . . .	21
7.5	Narzędzia testowe . . . . .	22

---

## Wprowadzenie - o dokumencie

---

### 1.1 Cel dokumentu

Celem dokumentu jest uporządkowanie podstawowych informacji o infrastrukturze projektu, organizacji pracy zespołu, rodzajach komunikacji czy sposobie wymiany dokumentów i kodu między członkami zespołu.

### 1.2 Odbiorcy

- Dr inż. Jakub Miler - prowadzący przedmiot *Realizacja projektu informatycznego*,
- Dr inż. Katarzyna Łukasiewicz - prowadzący zajęcia projektowe,
- Katedra Inżynierii Oprogramowania,  
Wydział Elektroniki, Telekomunikacji i Informatyki,  
Politechnika Gdańska,
- Członkowie zespołu projektowego:

Zuzanna Nowak, 193165 - kierownik projektu

Agata Domasik, 193577

Jakub Walasik, s193650

Adam Chabraszewski, s193373

### 1.3 Terminologia

- **Streaming** – transmitowanie rozgrywki na żywo za pośrednictwem platform internetowych, takich jak Twitch, YouTube czy Facebook Gaming. Umożliwia interakcję z widzami i popularyzację gry.

- **Turnieje online** – rywalizacja graczy w sieci w zorganizowanych zawodach, często z ustalonymi zasadami i nagrodami. Może być prowadzona na dedykowanych platformach turniejowych.
- **Rozproszona aplikacja** – aplikacja, której komponenty działają na różnych urządzeniach lub serwerach, komunikując się za pomocą sieci.
- **Sesja gry** – jednostka rozgrywki, w której gracze dołączają do wspólnej gry według określonych zasad.
- **Kapitan drużyny** – gracz odpowiedzialny za podawanie wskazówek w grze „Tajniacy”.
- **Interfejs użytkownika (UI)** – część aplikacji, z którą użytkownicy wchodzi w interakcję, np. menu gry, przyciski czy plansza gry.
- **Backend** – część systemu odpowiedzialna za przetwarzanie danych i logikę aplikacji, działająca po stronie serwera.
- **Frontend** – część systemu odpowiedzialna za interakcję użytkownika, działająca po stronie przeglądarki lub aplikacji klienckiej.
- **WebSocket** – technologia umożliwiająca dwukierunkową komunikację w czasie rzeczywistym pomiędzy klientem a serwerem.
- **Repozytorium Git** – system kontroli wersji używany do przechowywania i zarządzania kodem źródłowym projektu.

### 2.1 Nazwa projektu

Rozproszona aplikacja do rozgrywek w grę imprezową 'Tajniacy'.

### 2.2 Adresowany problem

Projekt odpowiada na potrzebę stworzenia cyfrowej wersji popularnej gry planszowej "Tajniacy", umożliwiającej rozgrywkę bez posiadania fizycznej wersji gry. Obecne na rynku implementacje nie oddają w pełni mechaniki oryginalnej gry, co utrudnia rozgrywkę i pogarsza wrażenia z gry.

### 2.3 Obszar zastosowania

Gra znajduje zastosowanie w obszarach:

- Rozrywka i gry towarzyskie online,
- Trening logicznego myślenia,
- Narzędzia do integracji zespołów,
- Streaming i turnieje online

### 2.4 Rynek

Docelowym rynkiem dla tego produktu są różnorodne grupy odbiorców:

- Gracze gier planszowych,
- Fani gier słownych i łamigłówek,
- Osoby poszukujące rozgrywki w formie cyfrowej,

- Zespoły pracownicze szukające narzędzi do integracji zespołu,
- Edukatorzy wykorzystujący gry jako narzędzia dydaktyczne,
- Rodziny, szukające rozrywki dla różnych grup wiekowych

## 2.5 Interesariusze

Interesariuszami projektu są:

- **Deweloperzy** odpowiedzialni za implementację,
- **Testerzy** weryfikujący funkcjonalności
- **Gracze**

## 2.6 Użytkownicy i ich potrzeby

Użytkownikami końcowymi produktu będą:

- gracze indywidualni - wymagają możliwości dołączania do publicznych rozgrywek,
- grupy znajomych - wymagają możliwości utworzenia rozgrywki prywatnej oraz zapraszania do niej graczy

Kluczowymi potrzebami wyżej wymienionych użytkowników są: prosty, intuicyjny interfejs oraz możliwość prowadzenia rozgrywki zdalnie.

## 2.7 Cel i zakres projektu

Celem projektu jest stworzenie rozproszonej wersji gry „Tajniacy”, umożliwiającej graczom rywalizację w czasie rzeczywistym z różnych lokalizacji. Gra będzie dostępna online, co pozwoli na granie w różnych środowiskach i na różnych urządzeniach. Rozgrzywka opiera się na współpracy drużynowej oraz analizie słów. Drużyny będą próbować odgadnąć słowa powiązane z podpowiedziami kapitana, zgodnie z zasadami oryginalnej gry. System zostanie zaprojektowany tak, by rozwijać strategiczne myślenie, umiejętność kojarzenia słów oraz pracę zespołową.

## 2.8 Ograniczenia

- **Czas realizacji:** Projekt ma ograniczony czas realizacji – dwa semestry.
- **Zasoby ludzkie:** Ograniczona liczba programistów.
- **Budżet:** Ograniczone fundusze na hosting i narzędzia.



## 2.9 Inne współpracujące systemy

Produkt będzie współpracował z serwerami obsługującymi rozgrywkę online, bazami danych przechowującymi informacje o użytkownikach i rozgrywkach oraz systemami autoryzacji zapewniającymi bezpieczny dostęp.

### 2.10 Harmonogram

#### 2.10.1 Termin

Projekt zostanie zrealizowany do 15 czerwca 2025 roku.

#### 2.10.2 Główne etapy projektu

Główne etapy realizacji projektu obejmują:

- Analiza i planowanie
  - Analiza wymagań - zbieranie wymagań funkcjonalnych i нефункциональных, opracowywanie scenariuszy użytkowników.
  - Planowanie techniczne - podjęcie decyzji na temat technologii i narzędzi. Projektowanie architektury aplikacji oraz przydzielenie zadań i ról w zespole.
- Faza prototypowania
  - Prototyp Backend - konfiguracja podstawowego serwera Spring Boot. Implementacja protokołu WebSockets do komunikacji w czasie rzeczywistym.
  - Prototyp Frontend - opracowanie ekranów gry, stworzenie prostego interfejsu dołączania do sesji gry.
- Etap realizacji
  - Podstawowa logika gry - implementacja mechaniki rozgrywki wieloosobowej, możliwości stworzenia sesji gry i logiki komunikacji między graczami.
  - System sesyjny - implementacja możliwości tworzenia i dołączania do sesji gry
- Testowanie i weryfikacja
  - Testy funkcjonalne - testy funkcjonalności oraz integracyjne
- Opracowanie dokumentacji



---

### Interesariusze i użytkownicy

---

#### 3.1 Interesariusze

Interesariuszami projektu są:

- **Deweloperzy** - odpowiedzialni za implementację systemu,
- **Testerzy** - odpowiedzialni za weryfikację funkcjonalności,
- **Gracze** - użytkownicy oprogramowania

#### 3.2 Użytkownicy końcowi

Użytkownikami końcowymi produktu będą:

- **Gracze indywidualni** - osoby korzystające z systemu do uczestnictwa w publicznych rozgrywkach,
- **Grupy znajomych** - zespoły osób pragnących prowadzić prywatne rozgrywki w zamkniętym gronie

#### 3.3 Klasyfikacja i krótki opis interesariuszy

##### 3.3.1 Interesariusze wewnętrzni

- **Deweloperzy** - zespół programistów zaangażowanych w tworzenie systemu,
- **Testerzy** - odpowiedzialni za weryfikację funkcjonalności

##### 3.3.2 Interesariusze zewnętrzni

- **Gracze indywidualni** - użytkownicy poszukujący możliwości uczestnictwa w publicznych rozgrywkach z innymi graczami.

- **Grupy znajomych** - zespoły użytkowników pragnących organizować prywatne rozgrywki.
- **Prawo UE i prawo narodowe** - zbiór regulacji prawnych, które określają zasady działania systemu i jego zgodność z przepisami.

#### 4.1 Skład zespołu

- Zuzanna Nowak, 193165 - kierownik projektu
- Agata Domasik, 193577
- Jakub Walasik, s193650
- Adam Chabraszewski, s193373

#### 4.2 Kompetencje i umiejętności członków zespołu

- Zuzanna Nowak:
  - *Umiejętności techniczne:*
    - \* **Języki programowania:** Java, C++, C, C#, JavaScript, TypeScript, PHP, Python, Bash, Język assemblera
    - \* **Frameworki:** Spring Framework, Spring Boot, React.js, AngularJS, Windows Presentation Foundation (WPF), Unity, Vue.js
    - \* **Bazy danych:** MySQL, MongoDB, Microsoft SQL Server, SQL, ELK Stack
    - \* **Front-end:** HTML5, CSS
    - \* **DevOps:** Ansible, Unix, Microsoft Azure, Git, Docker, Docker-Compose, Kubernetes
    - \* **Monitoring i serwery:** Grafana, ELK stack monitoring, Web Servers (Apache, Tomcat)
    - \* **Inne:** Algorytmy, Struktury danych
  - *Umiejętności dodatkowe:* Adobe Photoshop, Adobe Premiere Pro, Figma
  - *Języki obce:* Angielski (Certificate in Advanced English CAE)
  - *Edukacja:* Informatyka (studia inżynierskie – w trakcie)
  - *Doświadczenie zawodowe:* DevOps Engineer w firmie Vention

- **Agata Domasik:**

- *Umiejętności techniczne:*
  - \* **Języki programowania:** Java, C++, C, C#, JavaScript, TypeScript, PHP, Python, Bash, Język assemblera
  - \* **Frameworki:** Spring Framework, Spring Boot, React.js, AngularJS, Windows Presentation Foundation (WPF), Unity, Vue.js
  - \* **Bazy danych:** MySQL, MongoDB, Microsoft SQL Server, SQL
  - \* **Front-end:** HTML5, CSS
  - \* **DevOps:** Git
  - \* **Inne:** Algorytmy, Struktury danych
- *Umiejętności dodatkowe:* Figma
- *Języki obce:* Angielski (Certificate in Advanced English CAE), Niemiecki (poziom A1+)
- *Edukacja:* Informatyka (studia inżynierskie – w trakcie)

- **Jakub Walasik:**

- *Umiejętności techniczne:*
  - \* **Języki programowania:** Java, C++, C, C#, JavaScript, TypeScript, PHP, Python, Język assemblera
  - \* **Frameworki:** Spring Framework, Spring Boot, React.js, AngularJS
  - \* **Bazy danych:** MySQL, PostgreSQL, MongoDB, Microsoft SQL Server, SQL
  - \* **Front-end:** HTML5, CSS, SCSS
  - \* **DevOps:** Git
  - \* **Inne:** Algorytmy, Struktury danych
- *Języki obce:* Angielski (poziom C1)
- *Edukacja:* Informatyka (studia inżynierskie – w trakcie)

- **Adam Chabraszewski:**

- *Umiejętności techniczne:*
  - \* **Języki programowania:** Java, C++, C, C#, JavaScript, TypeScript, PHP, Python, Bash, Język assemblera
  - \* **Frameworki:** Spring Framework, Spring Boot, React.js, AngularJS, Unity, Vue.js
  - \* **Bazy danych:** MySQL, MongoDB, Microsoft SQL Server, SQL
  - \* **Front-end:** HTML5, CSS
  - \* **DevOps:** Git, Unix, Docker, Docker-Compose
  - \* **Inne:** Algorytmy, Struktury danych
- *Umiejętności dodatkowe:* konfiguracja CUDA
- *Języki obce:* Angielski (Certificate in Advanced English CAE)
- *Edukacja:* Informatyka (studia inżynierskie – w trakcie)
- *Doświadczenie zawodowe:* Software Developer w firmie Advanced Protection Systems Inc.

## 4.3 Zakresy odpowiedzialności

Imię i nazwisko	Zakres odpowiedzialności
Zuzanna Nowak	Backend, frontend, DevOps, design UI/UX, kierowanie zespołem, zarządzanie repozytorium
Agata Domasik	Backend, frontend, design UI/UX
Jakub Walasik	Backend, frontend, DevOps, testowanie
Adam Chabraszewski	Backend, frontend

## 4.4 Model pracy

Zespół pracuje w modelu zdalnym:

- Spotkania zespołowe odbywają się przez platformę Discord na dedykowanym serwerze projektu
- Regularne spotkania projektowe mają miejsce co tydzień we wtorek o godzinie 19:00
- Do zarządzania zadaniami zespół wykorzystuje GitHub Projects z tablicą Kanban
- Postępy prac są na bieżąco aktualizowane w systemie kontroli wersji GitHub
- Członkowie zespołu wykonują swoje zadania asynchronicznie, utrzymując stały kontakt przez Discord oraz Messenger

## 4.5 Dane kontaktowe

- Oficjalny e-mail kontaktowy zespołu: [codenames.contact@gmail.com](mailto:codenames.contact@gmail.com)
- Dane kontaktowe członków zespołu:
  - **Zuzanna Nowak:**
    - \* E-mail: [zuzanna.nowak.kontakt@gmail.com](mailto:zuzanna.nowak.kontakt@gmail.com)
    - \* LinkedIn: <https://www.linkedin.com/in/zuzanna-nowak-aa071125b/>
    - \* GitHub: <https://github.com/zuzqqa>
    - \* Facebook: <https://www.facebook.com/zuzia.nowak.50951/>
  - **Agata Domasik:**
    - \* E-mail: [agata.domasik@gmail.com](mailto:agata.domasik@gmail.com)
    - \* LinkedIn: <https://www.linkedin.com/in/agata-domasik-50277b2bb/>
    - \* GitHub: <https://github.com/agatadomasik>
    - \* Facebook: <https://www.facebook.com/agata.domasik>
  - **Jakub Walasik:**
    - \* E-mail: [walasikjakub@gmail.com](mailto:walasikjakub@gmail.com)
    - \* LinkedIn: <https://www.linkedin.com/in/jakub-walasik-9ba6b9348/>
    - \* GitHub: <https://github.com/jwalasik3>
    - \* Facebook: <https://www.facebook.com/jakub.walasik.5>

**– Adam Chabraszewski:**

- \* E-mail: [adam.chabraszewski@gmail.com](mailto:adam.chabraszewski@gmail.com)
- \* LinkedIn: <https://www.linkedin.com/in/adam-chabraszewski-574a1530a/>
- \* GitHub: <https://github.com/achabrasz>
- \* Facebook: <https://www.facebook.com/adam.chabraszewski.92>



---

### Komunikacja w zespole i z interesariuszami

---

#### 5.1 Harmonogram spotkań

Zespół spotyka się regularnie co tydzień we wtorek o godzinie 19:00 na serwerze Discord.

#### 5.2 Miejsca spotkań

Spotkania odbywają się na serwerze Discord.

#### 5.3 Narzędzia komunikacji

Komunikacja zespołu odbywa się głównie za pośrednictwem:

- Serwera Discord – wykorzystywanego zarówno do spotkań, jak i bieżącej komunikacji tekstowej i głosowej.
- Grupy na Messengerze – używanej do szybkich ustaleń i organizacji.

#### 5.4 Komunikacja z interesariuszami zewnętrznymi

Kontakt z interesariuszami zewnętrznymi odbywa się poprzez spotkania na żywo.

#### 5.5 Współpraca z opiekunem projektu

Spotkania z opiekunem projektu odbywają się nieregularnie, w zależności od potrzeby. Podczas tych spotkań zespół zdaje raport z postępów prac oraz uzyskuje odpowiedzi na ewentualne pytania.



---

### Współdzielenie dokumentów i kodu

---

#### 6.1 Metody wymiany dokumentów i kodu

Kod i dokumentacja projektu są współdzielone poprzez repozytorium GitHub. Każde zadanie jest rejestrowane jako **Issue**, po czym tworzona jest dla niego osobna gałąź (*branch*). Po zakończeniu pracy nad zadaniem składany jest **Pull Request** do głównej gałęzi (*main*).

#### 6.2 Repozytorium projektu

Projekt jest przechowywany w repozytorium na GitHubie, które zawiera zarówno kod źródłowy, jak i dokumentację.

##### 6.2.1 Administracja repozytorium

Administracją repozytorium zajmuje się **kierownik projektu**, który odpowiada za nadzór nad kodem, zatwierdzanie Pull Requestów oraz kontrolę dostępu. Gałąź *main* jest zablokowana i nie można do niej bezpośrednio dodawać zmian – każda zmiana musi przejść proces Pull Requesta.

#### 6.3 Zarządzanie dokumentacją

Dokumentacja projektu jest przechowywana w repozytorium w dedykowanym folderze docs.

##### 6.3.1 Osoba odpowiedzialna

Dokumentacją zajmują się wszyscy członkowie zespołu, jednak nadzór nad jej poprawnością sprawuje kierownik projektu.

### 6.3.2 Konwencje nazewnictwa

Pliki dokumentacji są nazywane zgodnie z następującymi zasadami:

- Nazwa pliku powinna odpowiadać tytułowi dokumentu, być w języku angielskim.
- Stosujemy format `snake_case` dla nazw plików (np. `project_organization.tex`).

### 6.3.3 Szablony dokumentów

Szablony dokumentów znajdują się w repozytorium w katalogu `docs/templates`.

## 6.4 System wersjonowania

Projekt wykorzystuje automatyczny system wersjonowania do zarządzania wersjami kodu.

### 6.4.1 Wersjonowanie kodu

Przed utworzeniem Pull Requesta należy uruchomić skrypt wersjonowania, który generuje nową wersję. Po zatwierdzeniu zmian, GitHub Actions automatycznie przypisuje nowy numer wersji do commita w gałęzi *main*.

### 6.4.2 Wersjonowanie dokumentacji

Dokumentacja jest wersjonowana ręcznie. Każda istotna zmiana powinna być odnotowana poprzez aktualizację numeru wersji dokumentu.

#### 7.1 Narzędzia komunikacyjne

W zespole wykorzystywane są następujące narzędzia do komunikacji:

- **Discord** – główna platforma do spotkań zespołu oraz bieżącej komunikacji głosowej i tekstowej.
- **Messenger** – używany do szybkich ustaleń oraz nieformalnej komunikacji zespołowej.

#### 7.2 Narzędzia do zarządzania dokumentacją

Dokumentacja projektu jest tworzona w formacie  $\text{\LaTeX}$  z wykorzystaniem następujących narzędzi:

- **GitHub** – repozytorium projektu zawiera dokumentację w katalogu docs.
- **Overleaf** – niektórzy członkowie zespołu korzystają z niego do edycji dokumentacji w  $\text{\LaTeX}$ .

#### 7.3 Narzędzia do zarządzania projektem

Zarządzanie projektem odbywa się za pomocą:

- **GitHub Projects** – tablica Kanban do śledzenia postępów prac, planowania zadań i zarządzania backlogiem.
- **GitHub Issues** – wykorzystywane do zgłaszania i monitorowania zadań oraz błędów.

#### 7.4 Narzędzia developerskie

Podczas pracy nad projektem wykorzystywane są następujące narzędzia programistyczne:

- **Visual Studio Code** – główny edytor dla większości języków używanych w projekcie.
- **IntelliJ IDEA** – środowisko programistyczne do pracy nad kodem w języku Java.
- **Docker Desktop** – platforma do zarządzania kontenerami Docker.
- **WSL (Windows Subsystem for Linux)** – umożliwia korzystanie z narzędzi linuksowych w systemie Windows.
- **Git** – system kontroli wersji używany do zarządzania kodem źródłowym.
- **MongoDB Compass** – narzędzie do przeglądania i zarządzania bazą danych MongoDB.

## 7.5 Narzędzia testowe

Testowanie aplikacji odbywa się za pomocą testów jednostkowych i integracyjnych:

- **JUnit** – framework do testów jednostkowych w języku Java.
- **Testy integracyjne** – realizowane w środowisku testowym z wykorzystaniem narzędzi do symulacji warunków produkcyjnych.



