



AKADEMIA GÓRNICZO-HUTNICZA  
IM. STANISŁAWA STASZICA W KRAKOWIE

Akademia Górniczo-Hutnicza w Krakowie  
Wydział Fizyki i Informatyki Stosowanej  
Informatyka Stosowana  
14 kwietnia 2021

## Inżynieria Oprogramowania

### **deja vu - prosty serwis internetowy z grami**

Lista wymagań klienta, ograniczeń systemu.  
Podział ról, koncepcja systemu.

Skład zespołu:

**Bartosz Rogowski**

Kinga Pyrek

Natalia Szczerba

Katarzyna Romańczyk

Mateusz Barnacki

Rafał Jakubczyk

Katarzyna Dobreńko

## Spis treści

1	Wizja projektu	2
2	Podział ról w zespole	2
3	Lista wymagań klienta	2
4	Ograniczenia systemu	3
5	Kamienie milowe	3
6	Koncepcja systemu	4
6.1	Architektura logiczna . . . . .	4
6.2	Zakres usług do zrealizowania . . . . .	4
6.3	Główne funkcje, grupy funkcji . . . . .	5
6.4	Założenia technologiczne . . . . .	5

# 1 Wizja projektu

Projekt zakłada realizację prostej aplikacji webowej (strony internetowej), na której zawartych jest kilka gier (3-4). Aplikacja będzie posiadać system logowania i autoryzacji użytkowników; po zalogowaniu jego wyniki będą zapisane w bazie; oraz system odznak, które będą przyznawane za zdefiniowane osiągnięcia. Strona jest dedykowana szczególnie osobom urodzonym w latach 1980-1990, ponieważ ma przypomnieć im o ich ulubionych grach z dzieciństwa. Ponadto wygląd również będzie nawiązywał do tamtych czasów i stylistyki retro.

Wspólnie ustalono, że nazwa strony będzie nosić nazwę *deja vu*, dlatego *proponowanym przez nas kodem jest DJV*.

## 2 Podział ról w zespole

Osoba	Rola
Bartosz Rogowski	teamleader, organizacja zespołu, dokumentacja, wizja projektu oraz strona graficzna, tester UX
Kinga Pyrek	front-end, pomoc w dokumentacji, wizja projektu
Natalia Szczerba	back-end coordinator
Katarzyna Romańczyk	database manager, back-end
Mateusz Barnacki	game developer
Rafał Jakubczyk	back-end
Katarzyna Dobreńko	testowanie, game developer

Tabela 1: Przyporządkowanie ról w zespole.

## 3 Lista wymagań klienta

Poniższa lista została wykonana poprzez uwzględnienie najważniejszych punktów, które zostały ustalone grupowo.

Aplikacja powinna:

- zawierać co najmniej 3 gry (kojarzące się z latami młodości osób urodzonych w latach 80.):
  - Memo
  - Snake
  - Sudoku
- mieć wygląd odpowiedni do tematyki gier, a także nawiązywać do estetyki retro
- być przejrzysta i intuicyjna dla użytkownika
- posiadać system logowania oraz autoryzacji do 200 użytkowników
- zawierać system odznak nagradzających użytkownika i zachęających do dalszego grania
- być przystosowana do działania w trybie 24/7 dla przynajmniej kilkudziesięciu osób naraz na różnych platformach systemowych

- bazę danych, w której będą przechowywane dane użytkowników oraz ich osiągnięcia
- zakończyć sesję po 15 minutach bezczynności.

## 4 Ograniczenia systemu

- około 4000 równoczesnych połączeń na godzinę na Heroku
- ograniczenie do 550 godz. miesięcznie dla aplikacji (liczone, gdy api jest włączone i wykorzystywane przez użytkowników)
- ograniczenia bazodanowe:
  - limit liczby rekordów w kolekcji bazy mongoDB
  - limit równoczesnych połączeń użytkowników z bazą mongoDB (ok. 500)

## 5 Kamienie milowe

Projekt został podzielony na 3 części tak, aby lepiej zorganizować pracę i śledzić postępy. Sprints oraz ich kamienie milowe:

1. Opracowanie prototypu strony internetowej z rozbudowanym front-endem, jedną grą oraz podstawowym wdrożeniem back-endu; przygotowanie wizji odznak.

Tablica Sprint 1 (19 zgłoszeń)			0	19	0	Rozpocznij sprint
✓	IP-16 Prepare Jira and add all needed tasks	CREATE DOCUMENTATION, REPOR...	W TOKU	BB		
✓	IP-12 Figure out page design	CREATE FRONT-END	W TOKU	KP		
✓	IP-4 Prepare mongoDB script and add to gamepage-api repo	CREATE BACKEND OF THE PAGE (S...	W TOKU	KP		
✓	IP-6 Make decision about app design (front end)	CREATE FRONT-END	W TOKU	KP		
✓	IP-17 Prepare Flask	CREATE BACKEND OF THE PAGE (S...	W TOKU	NS		
✓	IP-7 Develop main page	CREATE FRONT-END	W TOKU	KP		
✓	IP-9 Create session system	CREATE BACKEND OF THE PAGE (S...	W TOKU	NS		
✓	IP-3 Create 2 collections on mongoDB: users and memo, and insert sample records	CREATE BACKEND OF THE PAGE (S...	W TOKU	NS		
✓	IP-2 Create logo	CREATE FRONT-END	W TOKU	BB		
✓	IP-8 Create menu and footer to MEMO game	CREATE FRONT-END	W TOKU	KP		
✓	IP-11 Prepare MEMO game code	CREATE FIRST GAME PROTOTYPE	W TOKU	MB		
✓	IP-13 Add repo gamepage to Heroku	CREATE BACKEND OF THE PAGE (S...	W TOKU	RJ		
✓	IP-14 Add repo gamepage-api to Heroku	CREATE BACKEND OF THE PAGE (S...	W TOKU	RJ		
✓	IP-10 Create function 'saveMemoScore'	CREATE BACKEND OF THE PAGE (S...	W TOKU	NS		
✓	IP-15 Figure out automatic deployment	CREATE BACKEND OF THE PAGE (S...	W TOKU	RJ		
✓	IP-1 Create database in mongoDB	CREATE BACKEND OF THE PAGE (S...	W TOKU	KP		
✓	IP-5 Change mongoDB configuration in 'api.py' file	CREATE BACKEND OF THE PAGE (S...	W TOKU	KP		
✓	IP-23 Style MEMO game and verify the code	CREATE FIRST GAME PROTOTYPE	W TOKU	KP		
✓	IP-25 Prepare two project reports (3 ideas, requirements etc.)	CREATE DOCUMENTATION, REPOR...	W TOKU	BB		

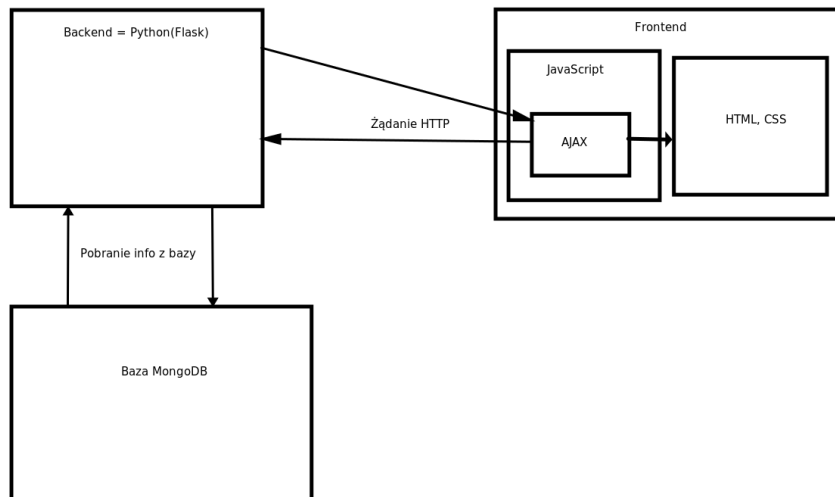
Rysunek 1: Zrzut ekranu z Jiry prezentujący zadania do zrealizowania - sprint 1.

2. Rozbudowa prototypu o dwie kolejne gry, system odznak, gotowy front-end; rozpoczęcie testów.
3. Wdrożenie wszelkich poprawek, pielęgnacja, ewentualna rozbudowa strony; oddanie projektu.

Zadania z kolejnych sprintów nie zostały zawarte w tym dokumencie, ponieważ wciąż trwają dyskusje nad listą obowiązków wraz z ich przydziałem, niemniej jednak kamienie milowe sugerują następne etapy prac.

## 6 Koncepcja systemu

### 6.1 Architektura logiczna



Rysunek 2: Diagram architektury logicznej aplikacji.

### 6.2 Zakres usług do zrealizowania

- stworzenie bazy danych, pozwalającej na rejestrację ograniczonej liczby użytkowników
- stworzenie co najmniej 3 prostych koncepcyjnie gier
- umożliwienie równoczesnego dostępu do serwisu dla około 20 osób
- wdrożenie systemu nagradzającego zarejestrowanych użytkowników
- umożliwienie zapisu najlepszych wyników, aby gracze mogli ze sobą konkurować.

## 6.3 Główne funkcje, grupy funkcji

- obsługa użytkownika:
  - rejestracja
  - logowanie
  - zmiana danych konta (nazwa, hasło)
  - wyświetlanie odznak
  - wyświetlanie rankingu
  - poprawne wyświetlanie interaktywnych gier
- obsługa gier:
  - zapisywanie wyników gier
  - przyznawanie odznak użytkownikom
- baza danych:
  - zapisywanie odczytywanie danych użytkowników
  - zapisywanie odczytywanie wyników gier

## 6.4 Założenia technologiczne

Narzędzia, systemy użyte podczas realizowania projektu:

- środowiska:
  - Windows
  - Linux
- deployment:
  - Heroku
  - GitHub Pages
- bazy danych:
  - mongoDB Atlas
- IDE:
  - VS Code, Sublime, notatnik lub inny dowolny edytor tekstowy
- repozytoria kodu:
  - GitHub (<https://github.com/I0-Proj>)
- dokumentacja:
  - Overleaf
- zarządzanie zespołem oraz projektem:
  - Jira
  - Discord (własny serwer)
  - MS Teams (składanie raportów, spotkania i konsultacje z Prowadzącą)