



Space Game

Specyfikacja wymagań

Autorzy: Adam Gryczka, Dariusz Dybka, Karol Tamowicz, Michał Jereczek

Spis treści

1	Wstęp	2
2	Technologie	2
3	Wymagania funkcjonalne	2
3.1	Administracja kontem	2
3.2	Obsługa interfejsu mapy	3
3.3	Obsługa interfejsu planety gracza	4
3.4	Obsługa interfejsu planety innego gracza	5
4	Wymagania niefunkcjonalne	6
4.1	Bezpieczeństwo	6
4.2	Wygoda użytkowania	6
4.3	Odporność na awarię	6
4.4	Skalowalność	6

1 Wstęp

Celem tego projektu jest zaimplementowanie gry przeglądarkowej. Użytkownik po utworzeniu konta uzyskuje dostęp do panelu użytkownika, gdzie może zarządzać utworzoną przez siebie planetą w konkretnym miejscu. Na każdej planecie będzie możliwość wydobywania dwóch surowców: Gadolin i Ununtium. Z czego Ununtium jest surowcem, który pełni rolę paliwa, zaś surowiec Ununtium pozwoli nam rozbudowywanie budynków i tworzenie floty. Użytkownik będzie miał możliwość wysyłania floty na inne planety w celu zaatakowania jej i zagrabienia surowców.

2 Technologie

Projekt postanowiliśmy zaimplementować przy użyciu następujących technologii.

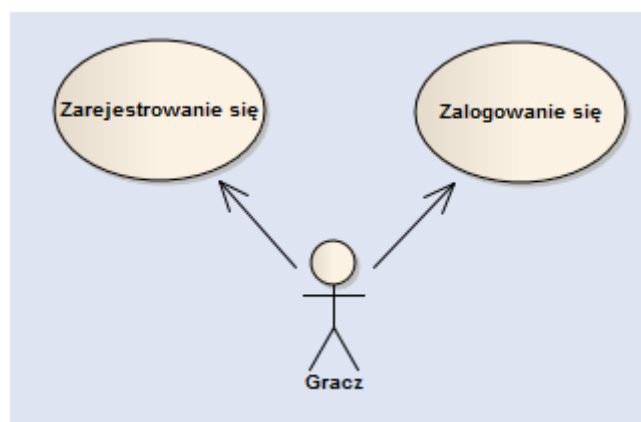
- **Back-end** - Spring MVC
- **Front-end** - Angular JS + Bootstrap
- **Baza danych** - MySQL

3 Wymagania funkcjonalne

3.1 Administracja kontem

Pierwszym krokiem rejestracji jest wybór nazwy użytkownika, hasła oraz podanie adresu email. Kolejnym krokiem jest wybór planety na mapie gdzie miałyby się znajdować planety. Mapa będzie przedstawiona jako siatka z wyszczególnionymi kratkami, gdzie użytkownik będzie mógł wybrać sobie miejsce swojej planety. Ostatnim etapem będzie kliknięcie przez użytkownika przycisku zarejestruj. W tym momencie system wyśle na podany adres email link z kodem aktywacyjnym. Użytkownik ma 1 dzień na aktywowanie swojego konta, w przeciwnym wypadku konto zostanie usunięte.

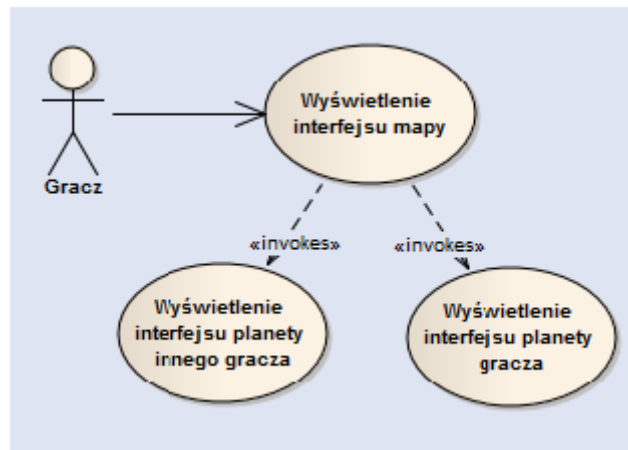
Podczas logowania, użytkownik może wpisać swoją nazwę użytkownika lub adres email i hasło. Po zalogowaniu zostaje przeniesiony na stronę główną panelu użytkownika.



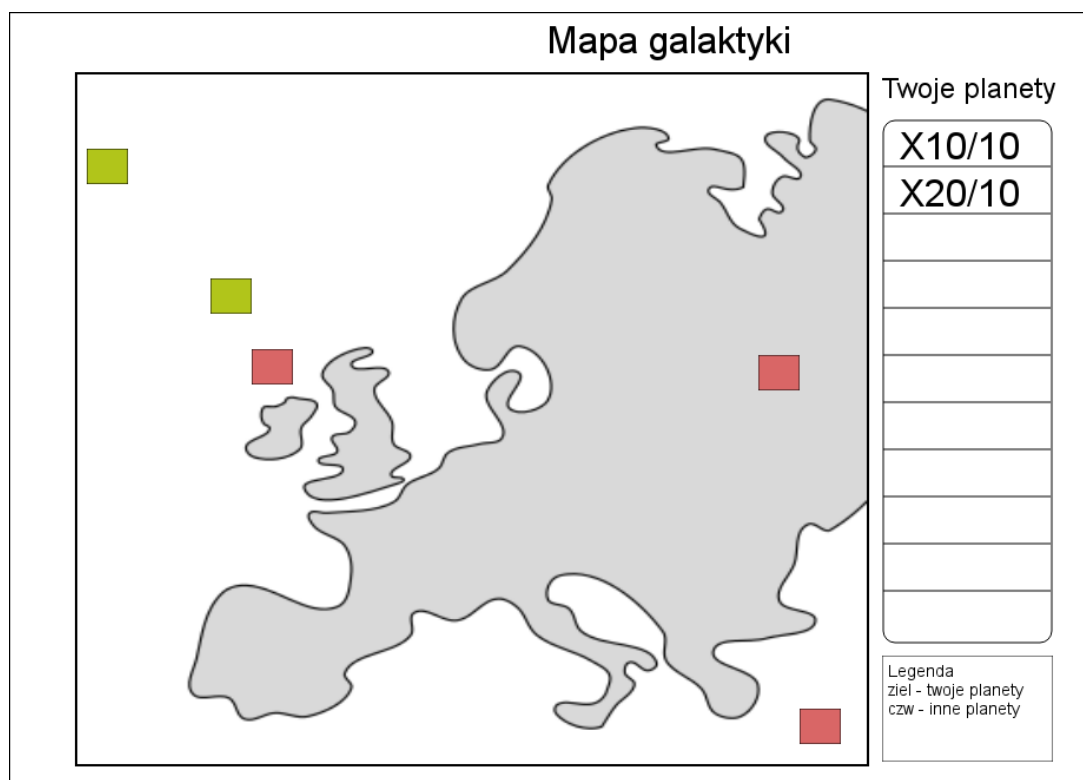
3.2 Obsługa interfejsu mapy

Po załogowaniu się użytkownik widzi interfejs mapy. Projekt interfejsu mapy przedstawia Rysunek 1. Rysunek 2 jest szkicem; mapa bowiem zawiera siatkę małych kwadratów w kilku kolorach (zielone i czerwone). Każdy kwadrat symbolizuje planetę. Kolorem czerwonym planety oznaczono planety innych graczy. Natomiast kolor zielony symbolizuje planety gracza.

Kliknięcie w odpowiedni kwadrat powoduje wyświetlenie interfejsu osoby klikającej albo wyświetlenie interfejsu innego gracza.

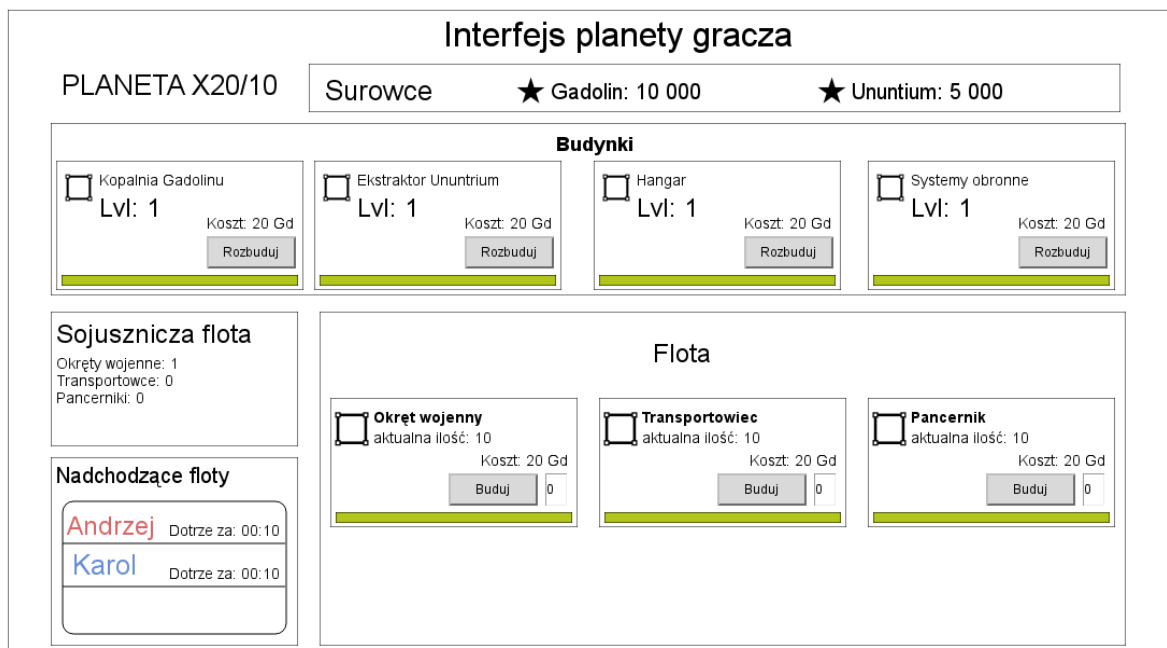
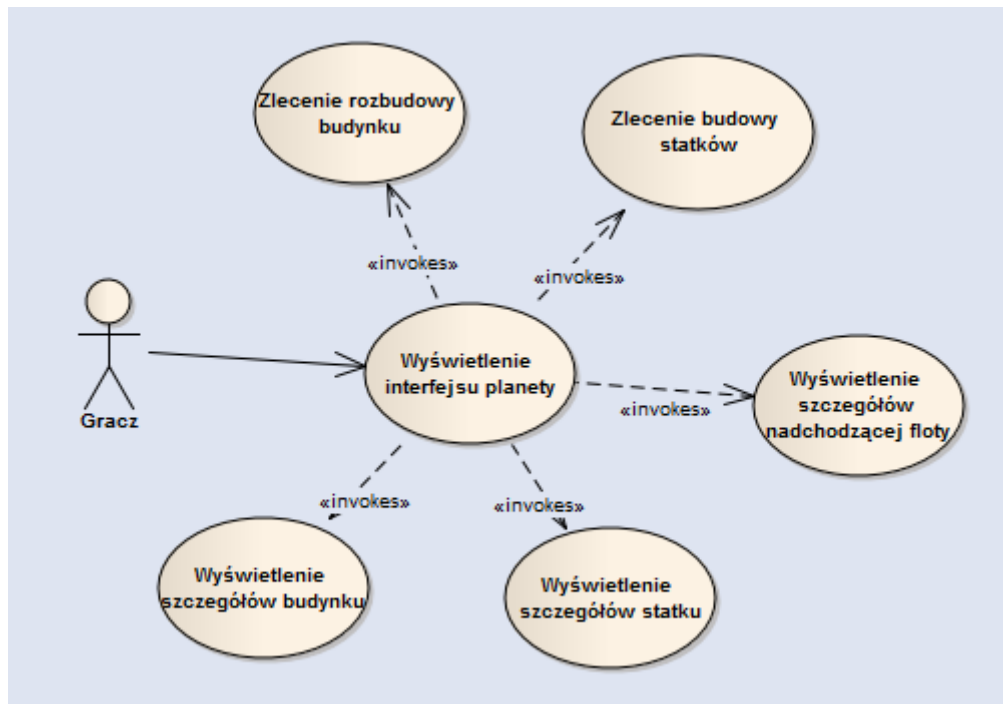


Rysunek 1: Przypadki użycia.



Rysunek 2: Interfejs mapy.

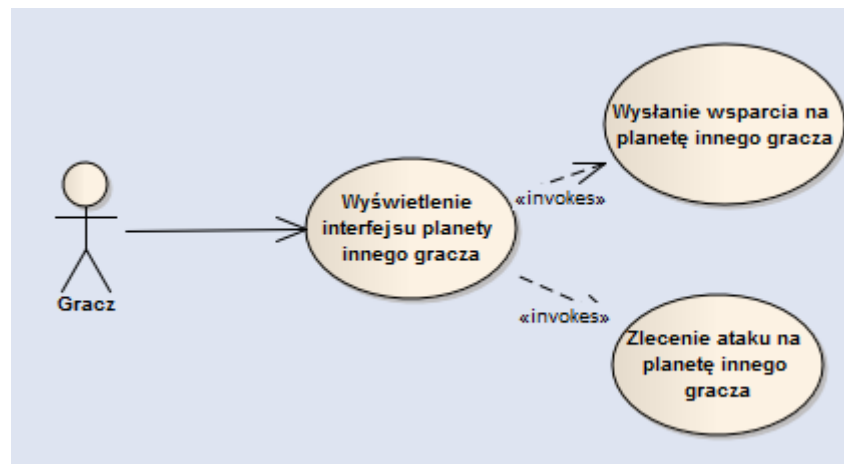
3.3 Obsługa interfejsu planety gracza



Rysunek 3: Rysunek przedstawia poglądowy projekt interfejsu planety gracza. Jest to możliwe po zalogowaniu się.

3.4 Obsługa interfejsu planety innego gracza

W interfejsie innego gracza jest możliwość, po jego wyświetleniu, wsparcia innego gracza. Chodzi o to, iż można wesprzeć atak (działać w sojuszu) innego gracza na inną planetę. Odminnym krokiem będzie możliwość zaatakowania planety innego gracza. Robi się to (zgodnie ze szkicem interfejsu planety innego gracza) poprzez wybór odpowiedniego radio-buttona.



Rysunek 4: Akcje możliwe do wykonania na planecie innego gracza

Interfejs planety innego gracza

PLANETA X20/10

Surowce★ Gadolin: 10 000★ Ununtium: 5 000

Budynki

☐ Kopalnia Gadolinu
Lvl: 1

☐ Ekstraktor Ununtium
Lvl: 1

☐ Hangar
Lvl: 1

☐ Systemy obronne
Lvl: 1

Wysłanie statków

Planeta X10/10

☒ Atak

Okręty wojenne: 0

Transportowce: 10

Pancerniki: 0

☐ Wsparcie

Wyslij

Rysunek 5: Projekt interfejsu przy wykonywaniu akcji na planecie innego gracza

4 Wymagania niefunkcjonalne

4.1 Bezpieczeństwo

Aplikacja będzie bezpiecznie przechowywać dane użytkowników. Dostęp do odpowiednich danych będzie możliwy dopiero po autoryzacji. Hasła użytkowników będą haszowane z użyciem soli.

4.2 Wygoda użytkowania

Aplikacja będzie posiadała przejrzysty interfejs, będzie działać w stylu single-page application. Dostęp do aplikacji będzie poprzez przeglądarkę internetową.

4.3 Odporność na awarię

Aplikacja będzie walidować dane przed wykonaniem żądania. Wszystkie błędy oraz sytuacje awaryjne będą logowane do osobnego pliku. Użytkownik otrzyma odpowiednią informację o błędzie gdy taki nastąpi.

4.4 Skalowalność

Aplikacja umożliwi dodawanie nowych komponentów takich jak: nowe budynki, rodzaje statków. Aplikację będzie można łatwo rozwijać i dodawać nowe funkcjonalności.