

# Dokumentacja projektu zaliczeniowego

Przedmiot: Inżynieria oprogramowania

Temat: „Sklep z grammi”  
Autorzy: Jagodziński Cezary, Kaźmierczak Filip, Kotoński Kamil,  
Michał Świeczak  
Grupa: I1-224A  
Kierunek: informatyka  
Rok akademicki: ...  
Poziom i semestr: I/4  
Tryb studiów: stacjonarne

*Należy pozostawić wszelkie nagłówki tego dokumentu, a umieszczać treść w odpowiednich miejscach zamiast obecnych objaśnień.*

*Stronę tytułową można sformatować w dowolny sposób, ale należy pozostawić zawartość informacyjną w układzie pokazanym powyżej.*

*Praca powinna zostać złożona wyłącznie w formacie pdf. Przed wygenerowaniem ostatecznej wersji należy zaktualizować spis treści – wyświetlane dwa poziomy.*

*Niniejszą informację należy również usunąć z wersji końcowej.*

# **1. Spis treści**

2	Odnośniki do innych źródeł	4
3	Słownik pojęć	5
4	Wprowadzenie	6
4.1	Cel dokumentacji	6
4.2	Przeznaczenie dokumentacji	6
4.3	Opis organizacji lub analiza rynku	6
4.4	Analiza SWOT organizacji	6
5	Specyfikacja wymagań	7
5.1	Charakterystyka ogólna	7
5.2	Wymagania funkcjonalne	7
5.3	Wymagania niefunkcjonalne	8
6	Zarządzanie projektem	9
6.1	Zasoby ludzkie	9
6.2	Harmonogram prac	9
6.3	Etapy/kamienie milowe projektu	9
7	Zarządzanie ryzykiem	10
7.1	Lista czynników ryzyka	10
7.2	Ocena ryzyka	10
7.3	Plan reakcji na ryzyko	10
8	Zarządzanie jakością	11
8.1	Scenariusze i przypadki testowe	11
9	Projekt techniczny	12
9.1	Opis architektury systemu	12
9.2	Technologie implementacji systemu	12
9.3	Diagramy UML	12
9.4	Charakterystyka zastosowanych wzorców projektowych	12
9.5	Projekt bazy danych	12
9.6	Projekt interfejsu użytkownika	12
9.7	Procedura wdrożenia	13
10	Dokumentacja dla użytkownika	14
11	Podsumowanie	15
11.1	Szczegółowe nakłady projektowe członków zespołu	15
12	Inne informacje	16



## **2. Odnośniki do innych źródeł**

tj. do wykorzystywanych narzędzi / projektów w tych narzędziach

- Zarządzania projektem – Jira, Trello, itp.
- Wersjonowanie kodu – sugerowany Git (hosting np. na Bitbucket lub Github), ew. SVN
- System obsługi defektów – np. Bitbucket, Github, Bugzilla.

diagramy - <https://www.visual-paradigm.com>

### **3. Słownik pojęć**

Tabela lub lista z pojęciami, które wymagają wyjaśnienia, wraz z tymi wyjaśnieniami – w szczególności synonimy różnych pojęć używanych w dokumentacji.

Gość - Osoba niezalogowana na stronie.

Game dev - deweloper gier.

## 4. Wprowadzenie

### 4.1 *Cel dokumentacji*

po co ją robimy i co zawiera (poziom szczegółowości)

Celem dokumentacji jest przedstawienie koncepcji oprogramowania związanego z rynkiem IT, zwłaszcza gier. Dokumentacja ma za zadanie jasno określić cele projektu, funkcje, oraz dostarczyć informacji potrzebnych do zrozumienia i efektywnego wdrożenia systemu. Poziom szczegółowości dokumentacji będzie zróżnicowany, obejmując zarówno ogólne cele, jak i bardziej techniczne aspekty projektu.

### 4.2 *Przeznaczenie dokumentacji*

dla kogo ona jest

Dla zespołu deweloperskiego: Zawierać będzie szczegółowe informacje na temat architektury systemu, interfejsów programistycznych, oraz instrukcje dotyczące implementacji.

Dla menedżerów projektu: Skupi się na celach biznesowych, harmonogramie prac, zasobach potrzebnych do realizacji projektu.

Dla klientów: Oferować będzie prostą i zrozumiałą prezentację funkcji systemu, instrukcje obsługi, oraz informacje o korzyściach wynikających z korzystania z oprogramowania.

### 4.3 *Opis organizacji lub analiza rynku*

Jedna z dwóch opcji:

1. Jeśli dla konkretnej organizacji: Czym jest organizacja, dla której realizowany będzie system; jak działa (lub będzie działała), kiedy system będzie wdrożony – tutaj nie odwołujemy się do samego systemu, tylko opisujemy samo działanie organizacji i role. W szczególności – jak wyglądają główne procesy biznesowe.
2. Jeśli na masowy rynek: Pobieźna analiza rynku. Dla kogo będzie przydatny taki system. Ile jest organizacji, które będą mogły z niego skorzystać, użytkowników w poszczególnych organizacjach. Czy te organizacje stanowią jednorodną grupę czy są różne rodzaje. Co one mają ze sobą wspólnego. Jak ta liczba będzie się zmieniała w najbliższej przyszłości.

## Rynek masowy

**Dla kogo będzie przydatny taki system:**

Nasz system będzie przydatny dla szerokiego spektrum użytkowników, w tym:

**Indywidualni gracze:** Osoby poszukujące nowych gier oraz platformy do interakcji z innymi graczami.

**Deweloperzy gier/Studia:** Twórcy gier, którzy chcą promować, sprzedawać i zarządzać swoimi tytułami na jednej zintegrowanej platformie.

### **Liczba organizacji i użytkowników:**

**Organizacje:** Oczekujemy, że nasza platforma przyciągnie uwagę wielu użytkowników(klient/gracz) w tym także różnych organizacji w branży gier, obejmujących zarówno małe niezależne studia, jak i duże korporacje.

**Użytkownicy:** Szacujemy, że nasza platforma może zyskać miliony użytkowników indywidualnych oraz setki organizacji w ciągu pierwszego roku od wprowadzenia.

### **Różnorodność organizacji/klientów:**

Organizacje korzystające z naszej platformy będą się dzielić głównie na dwie grupy: twórców gier, studia, które chcą upublicznić swoje dzieła i użytkowników, którzy poszukują gier do zakupów.

### **Wspólne cechy organizacji:**

Wszystkie organizacje korzystające z naszego systemu mają wspólny cel – efektywne zarządzanie, promocja i dostarczanie gier na coraz bardziej rozwijającym się rynku. Z drugiej strony klienci/użytkownicy szukają gier do zakupu dla rozrywki.

### **Prognoza zmian w przyszłości:**

Przewidujemy dynamiczny wzrost liczby użytkowników i organizacji korzystających z naszej platformy w najbliższych latach. Rozwój technologii, rosnące zainteresowanie grami wideo oraz rozwijająca się kultura e-sportu są czynnikami, które mogą przyspieszyć ten wzrost.

## **4.4 Analiza SWOT organizacji**

- jeśli system dla konkretnej organizacji:
  - wystarczy sama tabela 2x2 (silne-słabe-szanse-zagrożenia)
- jeśli system na masowy rynek:
  - szanse i zagrożenia

### **Szanse:**

**Współpraca** - nawiązanie współpracy z różnymi twórcami gier może poszerzyć bibliotekę dostępnych tytułów.

**Rozwój** - ekspansja, czyli wejście na rynki zagraniczne.

**Technologia** - korzystanie z najnowszych technologii może przynieść konkurencyjne przewagi, np. poprawić jakość usług takich jak szybkość pobierania gier.

**Wzrost rynku gier wideo** - rośnie popularność gier wideo, co stwarza możliwości dla nowych platform i usług.

**Zagrożenia:**

**Związane z bezpieczeństwem** - zagrożenia związane z bezpieczeństwem danych użytkowników może zaszkodzić reputacji i zaufaniu do platformy.

**Konkurencja** - rynek jest konkurencyjny, a istniejące platformy mogą wprowadzać nowe funkcje lub rozwiązania co przyciągnie użytkowników.

**Zmiany prawne** - ewolucje w przepisach i regulacjach branżowych mogą wpływać na sposób prowadzenia biznesu.



## 5. Specyfikacja wymagań

### 5.1 Charakterystyka ogólna

#### 5.1.1 Definicja produktu

jedno zdanie o systemie – nazwa i rodzaj

System umożliwiający publikację gier przez twórców/studia, które później mogą być zakupione przez innych użytkowników.

Nazwa produktu maets.

#### 5.1.2 Podstawowe założenia

do czego będzie służył ten system – kilka/kilkanaście zdań wprowadzających

System będzie służył jako platforma do zakupu, sprzedaży i dystrybucji gier online. Użytkownicy będą mieli możliwość tworzenia konta, przeglądania ofert gier, dodawania ich do koszyka lub listy życzeń, dokonywania płatności online za zakupione tytuły oraz korzystania z różnorodnych funkcji społecznościowych, takich jak dodawanie znajomych, prowadzenie czatów, udział w dyskusjach na forum, ocenianie gier i udzielanie zwrotów pod określonymi warunkami. Dodatkowo, system będzie automatycznie aktualizował gry, oferując użytkownikom najnowsze wersje, oraz umożliwia dodawanie i przeglądanie opisów, trailery oraz zrzutów ekranu do poszczególnych tytułów. Możliwe będzie również subskrybowanie gier za pomocą abonamentów oraz korzystanie z kuponów rabatowych i promocji.

#### 5.1.3 Cel biznesowy

co organizacja docelowo chce osiągnąć wdrażając system

**Zwiększenie sprzedaży:** Głównym celem może być zwiększenie sprzedaży gier poprzez platformę online. Poprzez umożliwienie łatwego dostępu do szerokiego wyboru gier, organizacja może przyciągnąć większą liczbę klientów i zwiększyć obroty.

**Zbudowanie zaufania wśród klientów:** Zbudowanie lojalnej bazy klientów poprzez oferowanie atrakcyjnych promocji i dostęp do szybkiego kontaktu z pomocą.

**Poprawa doświadczenia użytkownika:** Poprzez zapewnienie intuicyjnego interfejsu, szybkich czasów ładowania, wsparcia dla wielu platform czy bogatej funkcjonalności społecznościowej, organizacja może dążyć do poprawy doświadczenia użytkownika i przyciągnięcia nowych klientów.

**Rozwój społeczności:** Platforma może służyć jako miejsce spotkań dla społeczności graczy, umożliwiając wymianę doświadczeń, recenzji, czy udział w turniejach. Budowanie aktywnej społeczności może przyczynić się do zwiększenia zaangażowania użytkowników i promocji platformy.

**Zwiększenie udziału w rynku:** Poprzez oferowanie konkurencyjnych cen, ekskluzywnych tytułów lub innowacyjnych funkcji, organizacja może dążyć do zwiększenia swojego udziału w rynku cyfrowej dystrybucji gier.

**Zapewnienie bezpieczeństwa i ochrony danych:** Zabezpieczenie platformy przed atakami cybernetycznymi oraz ochrona danych klientów może być kluczowym celem organizacji, szczególnie w kontekście przechowywania danych osobowych i finansowych użytkowników.

**Wsparcie dla deweloperów:** Organizacja może dążyć do zapewnienia wsparcia dla niezależnych deweloperów gier poprzez udostępnianie narzędzi i zasobów, promowanie ich tytułów oraz oferowanie korzystnych warunków współpracy.

#### 5.1.4 Użytkownicy

lista – ew. wyjaśnienia dodać do słownika pojęć

**Klient/Gracz:** korzysta ze sklepu, swojej biblioteki z grami oraz wszystkich funkcji społeczności.

**Gość:** może przeglądać zawartość sklepu bez możliwości zakupu gier, .

**Game Dev:** wystawia swoje gry w sklepie. Po wystawieniu gry wprowadza do nich aktualizacje.

**Administrator systemu:** zarządza systemem w tym ustawieniami, bezpieczeństwem i aktualizacjami.

**Moderator forum:** monitoruje aktywność użytkowników forum w celu zapewnienia zgodności z zasadami społeczności.

**Tester:** Osoba zatwierdzająca nowo dodane gry po ich sprawdzeniu również po okresie próbnym decyduje o tym czy gra spełnia wymagania dot. jakości i czy nadaje się do dalszej sprzedaży w sklepie.

#### 5.1.5 Korzyści z systemu

dla poszczególnych grup użytkowników – każdy element z unikalnym numerem identyfikacyjnym

##### 1.Dla graczy:

- 1.1 Łatwy dostęp do szerokiej gamy gier online.
- 1.2 Możliwość zakupu i pobierania gier bez konieczności wychodzenia z domu.
- 1.3 Udogodnienia społecznościowe, takie jak czaty i dodawanie znajomych, umożliwiające interakcję z innymi graczami.
- 1.4 Możliwość oceniania i recenzowania gier, co pomaga w podejmowaniu decyzji zakupowych.
- 1.5 Śledzenie aktualizacji zapewniając dostęp do najnowszych funkcji i poprawek.
- 1.6 Możliwość skorzystania z promocji, kuponów rabatowych oraz subskrypcji, co umożliwia oszczędność pieniędzy na zakupach gier.
- 1.7 Korzystanie z funkcji listy życzeń, aby śledzić interesujące tytuły i otrzymywać powiadomienia o promocjach.

##### 2.Dla deweloperów gier:

- 2.1 Platforma do dystrybucji ich gier, umożliwiającą dotarcie do szerokiej publiczności.

- 2.2 Możliwość promowania swoich tytułów za pomocą opisów, trailerów i zrzutów ekranu.
- 2.3 Ułatwiony proces aktualizacji gier oraz wersjonowania, co zapewnia użytkownikom dostęp do najnowszych funkcji i poprawek.
- 2.4 Opcja udostępniania gier za pomocą subskrypcji, co może generować stałe przychody.
- 2.5 Możliwość monitorowania opinii i recenzji graczy, co pomaga w doskonaleniu gier oraz tworzeniu lepszych doświadczeń dla użytkowników.

### **3.Dla społeczności graczy:**

- 3.1 Możliwość uczestniczenia w dyskusjach na temat gier na forum.
- 3.2 Możliwość dodawania znajomych i prowadzenia czatów, co umożliwia budowanie społeczności.
- 3.3 Dostęp do ocen i recenzji innych graczy, co ułatwia wybór odpowiednich tytułów.
- 3.4 Możliwość udostępniania wrażeń z grania oraz rekomendowania ulubionych gier innym użytkownikom.

## **5.1.6 Ograniczenia projektowe i wdrożeniowe**

przepisy prawne, specyficzne technologie, narzędzia, b.d., protokoły komunikacyjne, aspekty zabezpieczeń, zgodność ze standardami, powiązania z innymi aplikacjami, platforma sprzętowa, system operacyjny, inne komponenty niezbędne do współpracy – wszystko wraz z uzasadnieniem!

Prawne:

- Prawa autorskie: Konieczność przestrzegania praw autorskich do treści umieszczanych na platformie, takich jak grafika, dźwięk czy kod źródłowy gier. Wymaga to zapewnienia odpowiednich licencji i zgód od twórców lub właścicieli praw autorskich.
- Ochrona własności intelektualnej: Zagwarantowanie ochrony dla twórców gier przed kradzieżą ich dzieł lub nielegalnym wykorzystaniem. Platforma musi zapewnić mechanizmy zapobiegające piractwu oraz umożliwiające zgłaszanie naruszeń praw autorskich.
- Ochrona danych osobowych: Konieczność przestrzegania przepisów dotyczących ochrony danych osobowych, takich jak Ogólne Rozporządzenie o Ochronie Danych (GDPR) w Europie. Platforma musi zapewnić odpowiednie procedury zarządzania danymi osobowymi użytkowników oraz ich zgodę na przetwarzanie danych.
- Przepisy dotyczące płatności elektronicznych: Jeśli platforma umożliwia dokonywanie transakcji finansowych, konieczne jest przestrzeganie przepisów dotyczących płatności elektronicznych, w tym zabezpieczeń płatności online oraz ochrony danych kart płatniczych.

- Prawa konsumenckie: Konieczność przestrzegania praw konsumenckich, takich jak prawo do odstąpienia od umowy czy obowiązek udzielania informacji o produktach i usługach.

Projektowe:

- Kompatybilność z różnymi platformami sprzętowymi i systemami operacyjnymi: Wymóg zapewnienia kompatybilności systemu z różnymi platformami sprzętowymi i systemami operacyjnymi, aby umożliwić jak najszerszej grupie użytkowników dostęp do systemu.
- Integracje z innymi aplikacjami i platformami: Konieczność zapewnienia integracji z innymi aplikacjami i platformami, takimi jak systemy płatności online czy usługi społecznościowe, aby zapewnić użytkownikom pełnię funkcjonalności i wygodę korzystania z systemu.

## **5.2 Wymagania funkcjonalne**

### **5.2.1 Lista wymagań**

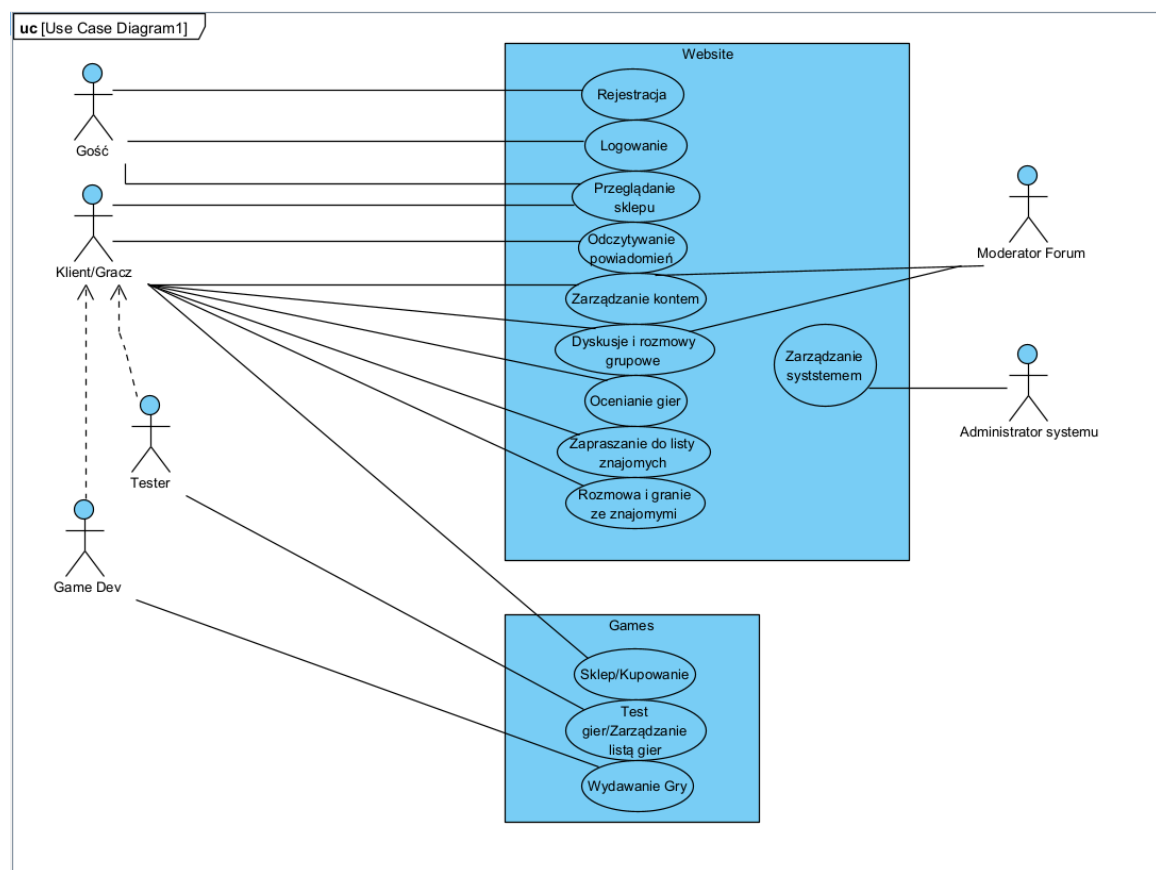
lista numerowana – czyli lista przypadków użycia lub bardziej ogólnie sformułowane wymagania, np. wymagania użytkownika

1. Możliwość przeglądania i wyszukiwania gier według kategorii, gatunków, ocen, popularności itp.
2. Funkcja tworzenia profilu użytkownika z możliwością dodawania zdjęć, opisów oraz ulubionych gier.
3. System oceniania i recenzowania gier przez użytkowników, z możliwością dodawania komentarzy i opinii.
4. Mechanizm powiadomień o nowych wydaniach gier, aktualizacjach oraz wydarzeniach związanych z platformą.
5. Integracja z zewnętrznymi platformami streamingowymi umożliwiającą udostępnianie treści z gier.
6. Funkcja dodawania gier do listy ulubionych oraz tworzenia listy życzeń.
7. System płatności umożliwiający zakup gier, mikropłatności w grach oraz wsparcie dla twórców poprzez donację.
8. Funkcja udostępniania użytkownikom możliwości tworzenia własnych gier i treści, w tym modyfikacji i skórek.
9. Możliwość interakcji społecznościowej, w tym czatu, forum dyskusyjnego, grup tematycznych itp.
10. Odpowiednie zabezpieczenia zapewniające prywatność danych użytkowników oraz ochronę przed atakami hakerskimi.
11. Dostępność platformy na różnych urządzeniach i systemach operacyjnych, w tym na komputerach stacjonarnych, laptopach, tabletach i smartfonach.
12. Szybkość i wydajność platformy zapewniająca płynne działanie nawet przy dużej liczbie użytkowników i danych.
13. Wsparcie techniczne dla użytkowników oraz twórców gier, zapewniające szybką pomoc w razie problemów lub pytań.

14. Stałe aktualizacje i rozwój platformy, uwzględniające opinie społeczności oraz nowe trendy w świecie gier wideo.
15. System rejestracji konta użytkownika umożliwiający zarządzanie danymi osobowymi oraz dostęp do funkcji platformy.

## 5.2.2 Diagramy przypadków użycia

Tutaj same diagramy – bez specyfikacji, ale każdy diagram z tytułem i na osobnej stronie



### 5.2.3 Szczegółowy opis wymagań

dla 5-7 wybranych najważniejszych przypadków użycia – przypadku zespołów 3-osobowych i większych, proporcjonalnie więcej każde na nowej stronie wg następujących punktów:

- Numer – jako ID
- Nazwa
- Uzasadnienie biznesowe – odwołanie (-a) do elementów wymienionych w 5.1.5. (id i treść elementu, do którego się odwołujemy)
- Użytkownicy
- Scenariusze, dla każdego z nich:
  - Nazwa scenariusza
  - Warunki początkowe
  - **Przebieg działań** – numerowana lista kroków, ze wskazaniem, kto realizuje dany krok
  - Efekty – warunki końcowe
  - Wymagania нефunkcjonalne – szczegółowe wobec poszczególnych wymagań funkcjonalnych
  - Częstotliwość - na skali 1-5 lub BN-BW
  - Istotność – inaczej: zależność krytyczna, znaczenie - na skali 1-5 lub BN-BW

#### ***Ważne!***

*Elementy od warunków początkowych do końca mogą być grupowane, tj. specyfikacja pojedynczego przypadku użycia może zawierać:*

- *pojedynczy przebieg działań (scenariusz główny) oraz ew. scenariusze alternatywne, albo*
- *wiele przebiegów głównych wraz z ew. scenariuszami alternatywnymi – wtedy każdy z przebiegów głównych powinien być opisany wg tych punktów (od warunków początkowych do końca).*

***Numer: 1***

***Nazwa: Ocena gry***

**Uzasadnienie biznesowe:**

Gracze mogą oceniać gry, co wpływa na decyzje zakupowe innych użytkowników. (Element 515.1)

**Użytkownicy:**

Gracze, społeczność platformy.

**Scenariusz:**

**Nazwa scenariusza:** Ocena gry

**Warunki początkowe:** Użytkownik jest zalogowany na platformie i ma dostęp do oceniania gier.

**Przebieg działań:**

- Użytkownik musi być zalogowany.
- Użytkownik wybiera grę, którą chce ocenić.
- Kliknięcie opcji "Oceń grę".
- Wybór oceny w skali od 1 do 5.
- Możliwość dodania komentarza, opisu.
- Potwierdzenie oceny.

**Efekty:**

**Warunki końcowe:** Ocena gry jest widoczna dla innych użytkowników, wpływa na ranking gier i opinie społeczności.

**Wymagania niefunkcjonalne:**

**Częstotliwość:** 5 (często wykorzystywany)

**Istotność:** 5 (istotny dla informacji zwrotnej dla innych użytkowników)

**Numer:** 2

**Nazwa:** Zakup gry

**Uzasadnienie biznesowe:**

Deweloperzy gier mogą generować przychody poprzez sprzedaż swoich tytułów na platformie. (Element 515.2)

**Użytkownicy:**

Gracze, deweloperzy gier.

**Scenariusz:**

**Nazwa scenariusza:** Zakup gry

**Warunki początkowe:** Użytkownik jest zalogowany na platformie i ma dostęp do sklepu.

**Przebieg działań:**

- Użytkownik musi być zalogowany.
- Przejsie do sklepu
- Użytkownik wybiera grę do zakupu.
- Kliknięcie przycisku "Kup teraz".
- Dodanie gry do koszyka
- Dalsze zakupy lub przejście do koszyka i zakończenie ich.
- Wybór formy płatności i potwierdzenie transakcji.
- Czekanie na odpowiedź od systemu czy pomyślnie zakończono transakcje.
- Pobranie lub odbiór klucza aktywacyjnego.

**Efekty:**

**Warunki końcowe:** Gra zostanie dodana do biblioteki użytkownika, a deweloper otrzymuje przychód z transakcji.

**Wymagania нефunkcjonalne:**

**Częstotliwość:** 4 (często wykorzystywany)

**Istotność:** 5 (istotny dla generowania przychodów dla deweloperów)

**Numer:** 3

**Nazwa:** Wydanie gry

**Uzasadnienie biznesowe:**

Deweloperzy gier mogą promować swoje tytuły i dotrzeć do nowej publiczności poprzez udostępnianie ich na platformie. (Element 515.3)

**Użytkownicy:**

Deweloperzy gier.

**Scenariusz:**

**Nazwa scenariusza:** Wydanie gry

**Warunki początkowe:** Deweloper jest zalogowany na platformie i ma gotową grę do opublikowania.

**Przebieg działań:**

- Deweloper przesyła grę na platformę.
- Testowanie gry.
- Decyzja o publikacji gry.



- Uzupełnienie danych o grze: opis, zrzuty ekranu, wymagania sprzętowe.
- Wybór modelu dystrybucji (płatna, darmowa, subskrypcja).
- Potwierdzenie publikacji.

**Efekty:**

**Warunki końcowe:** Gra jest dostępna do zakupu lub pobrania przez użytkowników.

**Wymagania niefunkcjonalne:**

**Częstotliwość:** 3 (czasem wykorzystywany)

**Istotność:** 5 (istotny dla zawartości platformy i jej atrakcyjności)

**Numer: 4**

**Nazwa: Logowanie się**

**Uzasadnienie biznesowe:**

Odwołanie do elementu 515.1: Gracze muszą być zalogowani, aby mieć dostęp do pełnej funkcjonalności platformy.

**Użytkownicy:**

Potencjalni gracze, społeczność platformy.

**Scenariusz:**

**Nazwa scenariusza:** Logowanie się

**Warunki początkowe:** Użytkownik ma konto na platformie lub musi je utworzyć.

**Przebieg działań:**

- Użytkownik otwiera stronę logowania.
- Wprowadzenie nazwy użytkownika, hasła.
- Kliknięcie przycisku "Zaloguj się".
- Sprawdzenie czy dane są poprawne.
- Potwierdzenie przez system.
- Zalogowanie użytkownika.

**Efekty:**

**Warunki końcowe:** Użytkownik uzyskuje dostęp do pełnej funkcjonalności platformy jako zalogowany użytkownik.

**Wymagania niefunkcjonalne:**

**Częstotliwość:** 5 (często wykorzystywany - użytkownicy muszą się logować za każdym razem, gdy korzystają z platformy)

**Istotność:** 5 (istotny dla zapewnienia pełnej funkcjonalności platformy)

**Numer: 5**

**Nazwa: Rejestracja**

**Uzasadnienie biznesowe:**

*Odwołanie do elementu 515.1: Aby korzystać z platformy, użytkownicy muszą mieć konto, dlatego istnieje konieczność rejestracji.*

**Użytkownicy:**

*Potencjalni gracze, społeczność platformy.*

**Scenariusz:**

**Nazwa scenariusza:** Rejestracja

**Warunki początkowe:** Użytkownik nie ma konta na platformie.

**Przebieg działań:**

- Użytkownik otwiera formularz rejestracji.
- Wprowadzenie danych osobowych: adres e-mail, nazwa użytkownika, hasło.
- System przyjmuje dane użytkownika i sprawdza czy użytkownik może utworzyć konto.
- Zaakceptowanie danych.
- Wysłanie linka aktywacyjnego na e-mail.
- Potwierdzenie rejestracji.
- Utworzenie konta użytkownika.

**Efekty:**

**Warunki końcowe:** Użytkownik uzyskuje dostęp do pełnej funkcjonalności platformy po zarejestrowaniu się.

**Wymagania niefunkcjonalne:**

**Częstotliwość:** 4 (rejestracja jest potrzebna tylko raz, ale w skali całej społeczności może być często wykorzystywana)

**Istotność:** 5 (istotny dla zapewnienia dostępu do pełnej funkcjonalności platformy)

### **5.3 Wymagania niefunkcjonalne**

W odniesieniu do całego systemu, modułów lub innych składowych systemu

1. Wydajność – w odniesieniu do konkretnych sytuacji – funkcji systemu
2. Bezpieczeństwo – utrata, zniszczenie danych, zniszczenie innego systemu przez nasz – wraz z działaniami zapobiegawczymi i ograniczającymi skutki
3. Zabezpieczenia

4. Inne cechy jakości – najlepiej ilościowo, żeby można było zweryfikować (zmierzyć)  
– adaptowalność, dostępność, poprawność, elastyczność, łatwość konserwacji, przenośność, awaryjność, testowalność, użyteczność

**Wydajność:**

- System musi obsługiwać jednocześnie dużą liczbę użytkowników podczas wysokich obciążeń, zapewniając stabilną i szybką odpowiedź. Na start możemy przyjąć że będzie to do 1 miliona użytkowników.
- Czas ładowania stron i interakcji z interfejsem użytkownika powinien być minimalny, nie przekraczając 2 sekund.

**Bezpieczeństwo:**

- Zapewnienie bezpiecznej autoryzacji i uwierzytelniania, z wykorzystaniem silnych algorytmów szyfrowania.
- Regularne tworzenie kopii zapasowych danych (co tydzień) w celu minimalizacji ryzyka utraty lub zniszczenia danych.

**Zabezpieczenia:**

- Wdrożenie systemu monitorowania zdarzeń, które umożliwiają wykrywanie i reagowanie na potencjalne zagrożenia.
- Ustawienie ścisłych uprawnień dostępu do danych, zapewniających, że tylko upoważnione osoby mają dostęp do określonych zasobów.

**Inne cechy jakości:**

- Adaptowalność: System powinien być łatwo modyfikowalny i dostosowywalny do zmieniających się potrzeb i wymagań biznesowych.
- Dostępność: Zapewnienie, że system jest dostępny dla użytkowników przez całą dobę, minimalizując przestoje i przerwy w działaniu.
- Poprawność: Zapewnienie, że system działa zgodnie z oczekiwaniami i spełnia określone kryteria jakościowe.
- Łatwość konserwacji: System powinien być łatwy w utrzymaniu i naprawianiu, z minimalnymi kosztami i wysiłkiem.
- Kompatybilność: System powinien być kompatybilny z różnymi platformami sprzętowymi i systemami operacyjnymi.
- Użyteczność: System powinien być intuicyjny i łatwy w użyciu, zapewniając pozytywne doświadczenia użytkowników.

## 6. Zarządzanie projektem

### 6.1 Zasoby ludzkie

(rzeczywiste lub hipotetyczne) – przy realizacji projektu

Należy założyć, że projekt byłby realizowany w całości jako projekt komercyjny a nie tylko częściowo w ramach zajęć na uczelni

- Manager
- Programiści (Java, JavaScript)
- Analitycy biznesowi
- Projektanci interfejsu użytkownika
- Testerzy oprogramowania

### 6.2 Harmonogram prac

Etapy mogą się składać z zadań.

Wskazać czasy trwania poszczególnych etapów i zadań – wykres Gantta.

obejmuje również harmonogram wdrożenia projektu – np. szkolenie, rozruch, konfiguracja, serwis – może obejmować różne wydania (tj. o różnej funkcjonalności – personal, professional, enterprise) i wersje (1.0, 1.5, itd.)

Etapy	Nazwa	Czas trwania	Przykładowe czynności
1	Analiza wymagań	5 tygodni	-Zbieranie wymagań od klienta -Analiza istniejących rozwiązań na rynku -Tworzenie dokumentacji wymagań
2	Projektowanie systemu	6 tygodni	-Tworzenie architektury systemu -Projektowanie interfejsu użytkownika -Tworzenie dokumentacji projektowej

3	Implementacja	10 tygodni	-Programowanie back-endu (Java) -Programowanie front-endu (JavaScript/React.js) -Integracja komponentów systemu
4	Testowanie	4 tygodnie	-Testy jednostkowe -Testy integracyjne -Testy akceptacyjne
5	Wdrożenie	2 tygodnie	-Rozpoczęcie wsparcia technicznego

### ***6.3 Etapy/kamienie milowe projektu***

dla głównych etapów projektu

- analiza wymagań,
- rozplanowanie projektu,
- zaprojektowanie interfejsu użytkownika,
- utworzenie dokumentacji,
- wypuszczenie wersji alfa - testowanie projektu przez małą grupę wybranych przez projektanta użytkowników,
- wypuszczenie wersji beta - testowanie projektu przez dużą grupę użytkowników,
- wypuszczenie finalnej wersji projektu.

W grupach prowadzonych przez ŁR ten cały ten rozdział jest opcjonalny – dla chętnych. Nie jest omawiany na wykładzie!

Studenci powinni skonsultować szczegółowe wymagania w tym zakresie z nauczycielem prowadzącym zajęcia w danej grupie.

## **7. Zarządzanie ryzykiem**

### **7.1 *Lista czynników ryzyka***

Wypełniona lista kontrolna

### **7.2 *Ocena ryzyka***

prawdopodobieństwo i wpływ

### **7.3 *Plan reakcji na ryzyko***

Działania w odniesieniu do poszczególnych ryzyk.

Mogą być wg różnych strategii, tj. kilka strategii dla pojedynczego czynnika ryzyka

Rozdział obowiązkowy w zespołach co najmniej 3-osobowych, w mniejszych – do uzgodnienia z prowadzącym zajęcia.

## 8. Zarządzanie jakością

### 8.1 Scenariusze i przypadki testowe

głównie testowanie funkcjonalności, ale może być też testowanie wymagań niefunkcjonalnych/zgodności; każdy scenariusz od nowej strony, musi zawierać co najmniej następujące informacje (sugerowany układ tabelaryczny, np. wg szablonu podanego w osobnym pliku lub na wykładzie):

- numer – jako ID
- nazwa scenariusza – co test w nim testowane (max kilka wyrazów)
- kategoria – poziom/kategoria testów
- opis – dodatkowe opcjonalne informacje, które nie zmieściły się w nazwie
- tester - konkretna osoba lub klient/pracownik,
- termin – kiedy testowanie ma być przeprowadzane,
- narzędzia wspomagające – jeśli jakieś są używane przy danym scenariuszu
- przebieg działań – tabela z trzema kolumnami: lp. oraz opisującymi działania testera i systemu
- założenia, środowisko, warunki wstępne, dane wejściowe – przygotowanie przed uruchomieniem testów
- zestaw danych testowych – najlepiej w formie tabelarycznej – jakie konkretnie dane mają być użyte przez testera i zwrócone przez system w poszczególnych krokach przebiegu działań
- *przebieg lub zestaw danych testowych musi zawierać jawną informację o warunku zaliczenia testu*

#### Rejestracja

lp	Test	System
1	Otworzyć okno rejestracji	
2		Wyświetla Okno
3	Wprowadź imię <imie>	
4		Weryfikacja poprawności <komunikat>

5	Wprowadź nazwisko <nazwisko>	
6		Weryfikacja poprawności <komunikat>
7	Wprowadź email <email>	
8		Weryfikacja poprawności <komunikat>
9	Wprowadź login <login>	
10		Weryfikacja poprawności <komunikat>
11	Wprowadź hasło <hasło>	
12		Weryfikacja poprawności <komunikat>
13	Zatwierdź rejestrację	
14		Informacja o utworzeniu konta <komunikat>

Zestaw 1 - wariant poprawny

3 : Filip

4 : Wprowadzono poprawne imię

5 : Kaźmierczak

6 : Wprowadzono poprawne nazwisko

7 : [Email@wp.pl](mailto:Email@wp.pl)

8 : Wprowadzono poprawne email

9 : MateuszDziobak

10 : Wprowadzono poprawny login

11 : Mateusz123

12 : Wprowadzono poprawne hasło

14 : Konto zostało utworzone pomyślnie

Zestaw 2 - wariant imię przekraczające 50 znaków

3 : (imię przekraczające 50 znaków)

4 : Długość imienia nie może przekraczać 50 znaków



Zestaw 3 - wariant nazwisko przekraczające 50 znaków

3 : Michał

4 : Wprowadzono poprawne imię

5 : (nazwisko przekraczające 50 znaków)

6 : Długość wprowadzonego nazwiska nie może przekraczać 50 znaków

Zestaw 4 - wariant błędny email

3 : Michał

4 : Wprowadzono poprawnie imię

5 : Świeczak

6 : Wprowadzono poprawnie nazwisko

7 : emailgmail.com

8 : Błędny email - brak “@”

Zestaw 5 - wariant błędny login

3 : Michał

4 : Wprowadzono poprawnie imię

5 : Świeczak

6 : Wprowadzono poprawnie nazwisko

7 : [Email@gmail.com](mailto:Email@gmail.com)

8 : Wprowadzono poprawnie email

9 : (login przekraczający 50 znaków)

10 : Długość wprowadzonego loginu nie może przekraczać 50 znaków

Zestaw 6 - wariant błędny, niepoprawne hasło

3 : Michał

4 : Wprowadzono poprawnie imię

5 : Świeczak

6 : Wprowadzono poprawnie nazwisko

7 : [Email@gmail.com](mailto:Email@gmail.com)

8 : Wprowadzono poprawnie email

9 : EjczuEbezoga

10 : Wprowadzono poprawny login

11 : (hasło przekraczające 50 znaków)

12 : Długość wprowadzonego hasła nie może przekraczać 50 znaków

Zestaw 7 - wariant błędny powtórzony login

3 : Michał

4 : Wprowadzono poprawnie imię

5 : Świczak

6 : Wprowadzono poprawnie nazwisko

7 : [Email@gmail.com](mailto:Email@gmail.com)

8 : Wprowadzono poprawnie email

9 : (wpisany login już istnieje w bazie danych)

10 : Taki login już istnieje

Zestaw 8 - wariant powtórzenia emaila

3 : Michał

4 : Wprowadzono poprawnie imię

5 : Świczak

6 : Wprowadzono poprawnie nazwisko

7 : PowtorzonyEmail[Email@gmail.com](mailto:Email@gmail.com)

8 : Taki email już istnieje

Zestaw 9 - wariant błędny za długi email

3 : Michał

4 : Wprowadzono poprawnie imię

5 : Świczak

6 : Wprowadzono poprawnie nazwisko

7 : (email o długości 51 znaków)

8 : Długość wprowadzanego emaila nie może przekraczać 50 znaków

Zestaw 10 - wariant imię zawierające specjalne znaki

3 : \$k\$oS\$m#ita

4 : imię zawiera znaki specjalne

Dodanie gry do koszyka

lp	Test	System
----	------	--------



4: Zbyt długa nazwa

Zestaw 5 - Dodanie zbyt dużej ilości gier do koszyka, limit 50

7 : Dodanie 51 różnych gier

Zestaw 6 - Dodanie gry, którą masz już w bibliotece

3: Wiedźmin 3

4: Gra zostało znaleziona

5: Wybranie gry z listy

6: Przejście do strony gry

7: Dodanie gry do koszyka

8: Komunikat o błędzie, bo grę już posiadasz w bibliotece

6:

Zestaw 7 - Gracz nie jest zalogowany

3: Wiedźmin 3

4: Gra zostało znaleziona

5: Wybranie gry z listy

6: Przejście do strony gry

7: Dodanie gry do koszyka

8: Komunikat Musisz zalogować się aby dodać grę do koszyka

Zestaw 8 - Wyszukanie gry bez nazwy

3 : ""

4 : Proszę podać nazwę gry do wyszukania

Zestaw 9 : Próba dodania gry, na którą masz nieodpowiedni wiek

3: Wiedźmin 3

4: Gra zostało znaleziona

5: Wybranie gry z listy

6: Komunikat o nieodpowiednim wieku

Zestaw 10 : Gra jest darmowa i nie można jej dodać do koszyka

3: Wiedźmin 3

4: Gra zostało znaleziona

5: Wybranie gry z listy

6: Przejście do strony gry

7: Dodanie gry do koszyka

8: Komunikat Gra jest darmowa i nie można jej dodać do koszyka

## 9. Projekt techniczny

### 9.1 Opis architektury systemu

z ew. rysunkami pomocniczymi

System Maets został podzielony na trzy główne części: serwer aplikacyjny, baza danych oraz interfejs użytkownika.

Serwer aplikacyjny:

- Odpowiada za przetwarzanie żądań użytkowników i zarządzanie logiką biznesową.
- Implementuje wszystkie funkcje aplikacyjne, takie jak zarządzanie kontami użytkowników, przetwarzanie płatności, zarządzanie zawartością gier, etc.

Baza danych:

- Przechowuje wszystkie dane systemu, takie jak profile użytkowników, informacje o grach, historię transakcji, etc.
- Może obejmować różne typy baz danych, w zależności od potrzeb i wymagań systemu, np. relacyjne bazy danych, bazy danych NoSQL, itp.
- Odpowiada za zapewnienie integralności danych oraz wydajnego dostępu do informacji.

Interfejs użytkownika:

- Zapewnia użytkownikom interakcję z systemem poprzez intuicyjny interfejs graficzny.
- Może obejmować aplikacje webowe, aplikacje mobilne oraz aplikacje desktopowe, dostosowane do różnych platform i urządzeń.
- Interfejs użytkownika wykorzystuje API serwera aplikacyjnego do komunikacji z backendem systemu.

### 9.2 Technologie implementacji systemu

tabela z listą wykorzystanych technologii, każda z uzasadnieniem

Technologia	Uzasadnienie
Java Gui	Implementacja graficzna programu
SQL	Zastosowany wraz z mySQL do stworzenia i zarządzania bazą danych
Java	Potrzebny do napisania aplikacji na

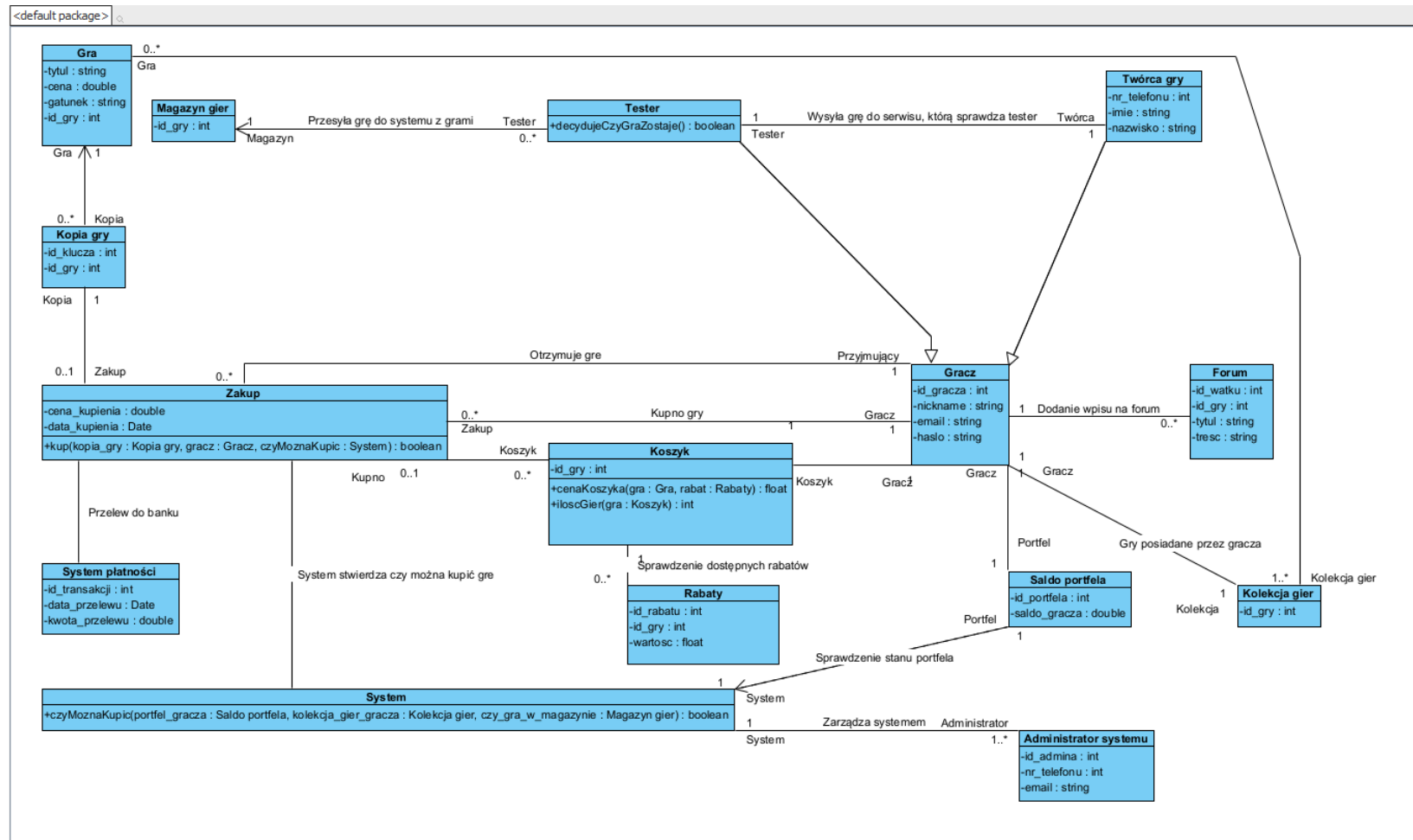
	urządzenia stacjonarne oraz mobilne
Komputer stacjonarny/laptop	Komputer stacjonarny lub laptop będzie głównym narzędziem programistycznym do pisania, testowania i debugowania kodu. Dzięki wydajności i mobilności laptopa, programiści będą mogli pracować zarówno w biurze, jak i w terenie.
Smartfony/tablety	Smartfony i tablety stanowią platformy docelowe dla interfejsu użytkownika. Testowanie aplikacji na różnych urządzeniach mobilnych pozwoli zapewnić kompatybilność i responsywność interfejsu użytkownika dla różnych rozmiarów ekranów i urządzeń.
Komputery i konsole do gier	Komputery i konsole do gier stanowią platformy docelowe dla gier w ramach systemu Maets. Testowanie gier na różnych platformach pozwoli zapewnić optymalne doświadczenie użytkownika na każdym urządzeniu.

### 9.3 Diagramy UML

każdy diagram ma mieć tytuł oraz ma być na osobnej stronie  
diagramy przypadków użycia umieszczone w punkcie 5.2.2, a nie tutaj.

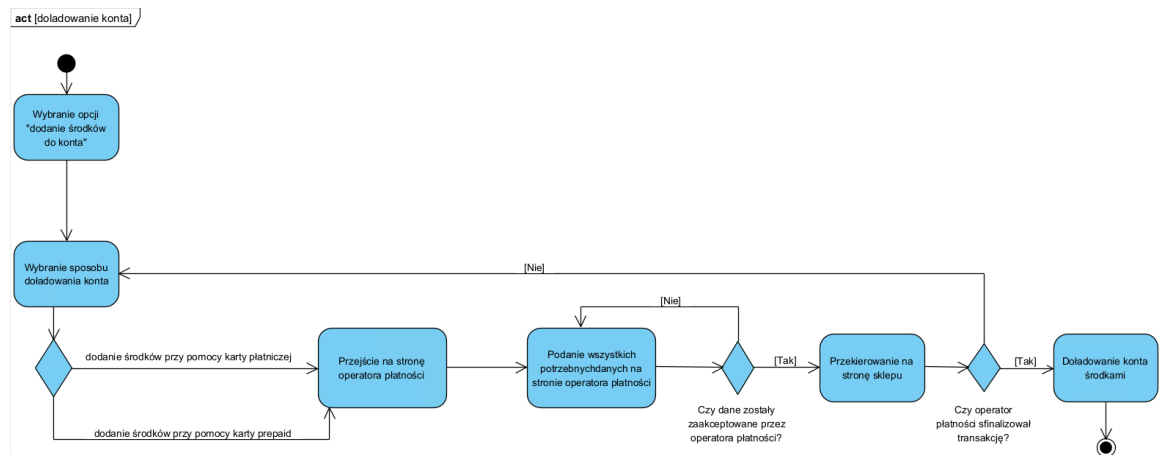
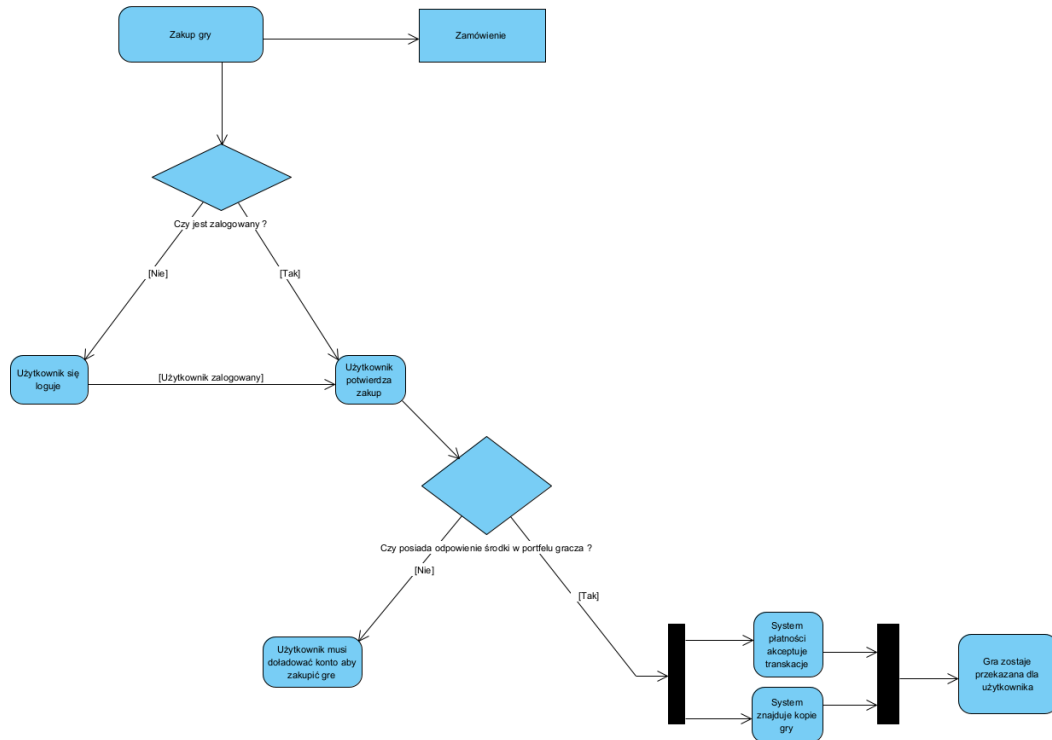
### 9.3.1 Diagram(-y) klas

1 lub więcej



### 9.3.2 Diagram(-y) czynności

Co najmniej 1 dla zespołów 2-osobowych, więcej dla liczniejszych  
złożony diagram jakies czynnosci (4 diagramy) activity diagram

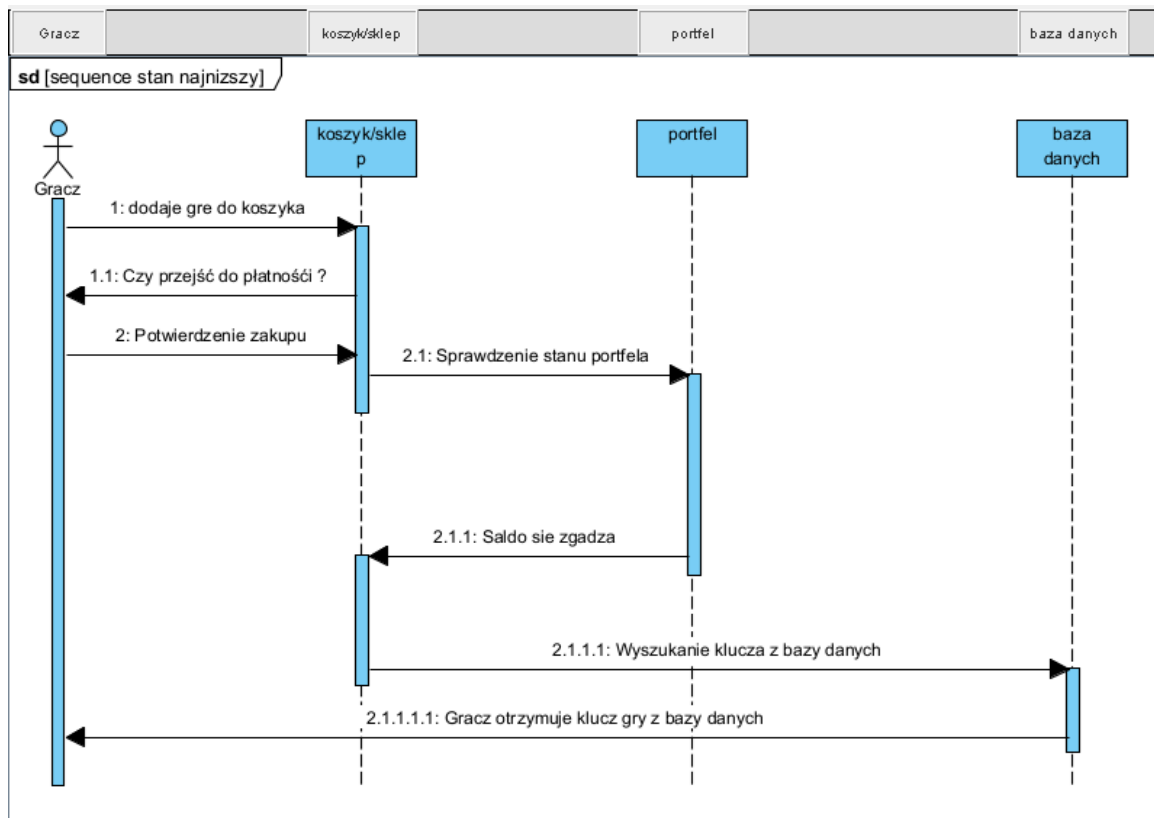
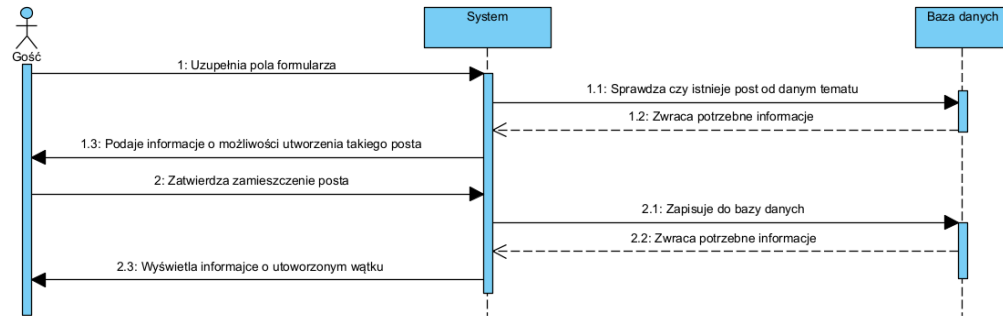


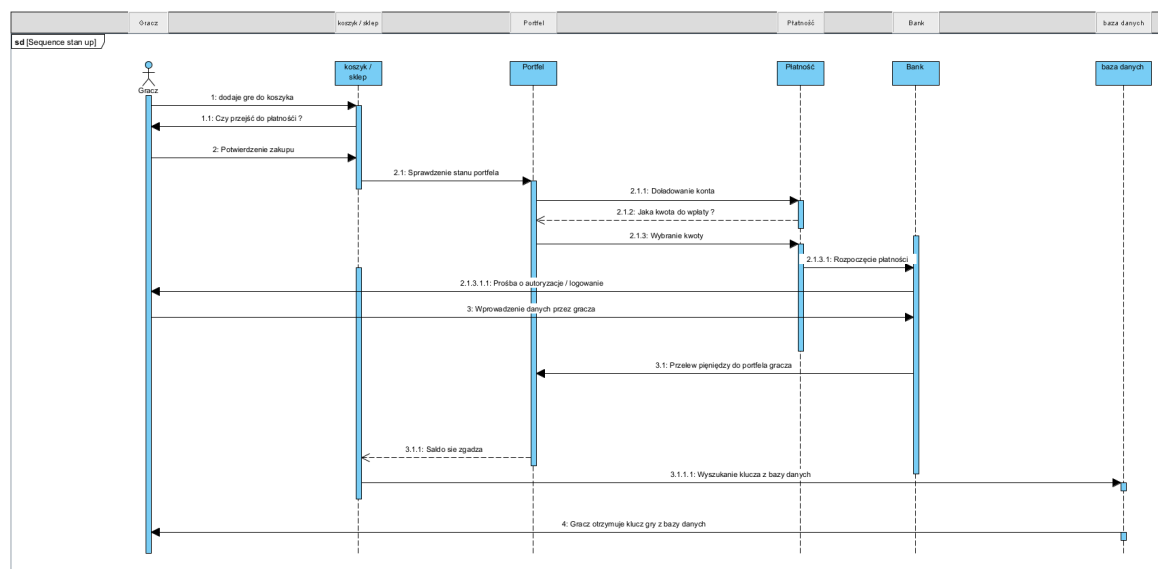
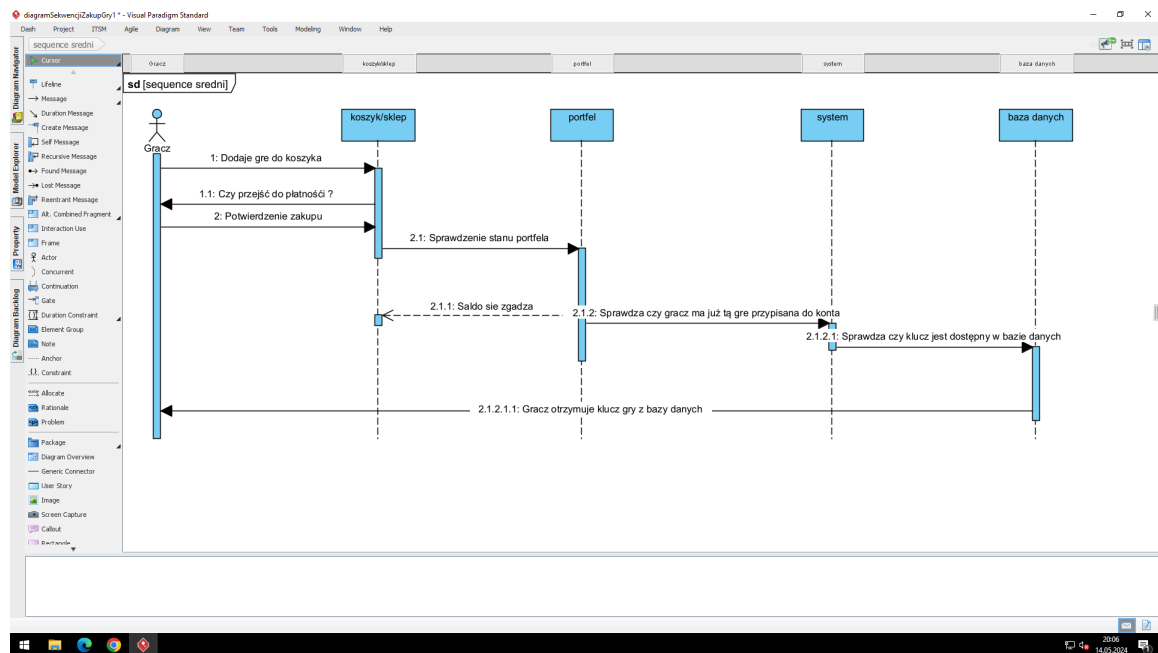


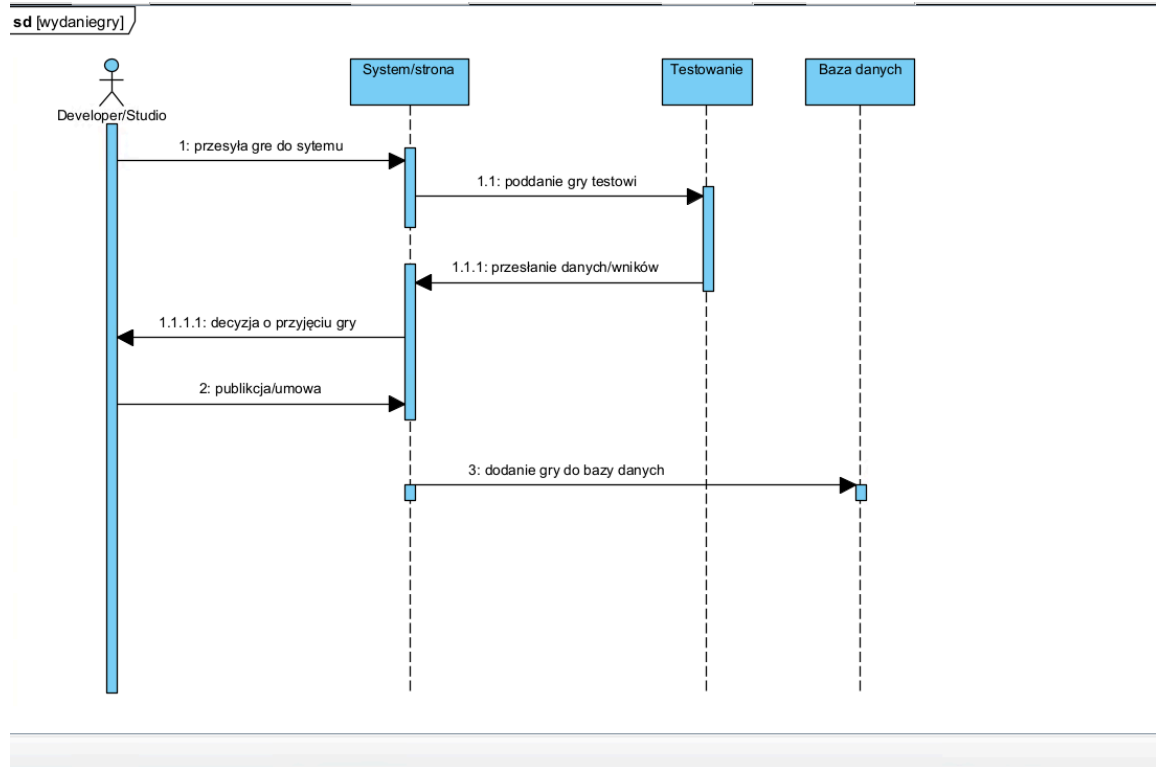
### 9.3.3 Diagramy sekwencji

co najmniej 5, w tym co najmniej 1 przypadek użycia zilustrowany kilkoma diagramami (dla zespołów 2-osobowych, dla liczniejszych więcej)

**sd** [Umieszczanie posta na forum]

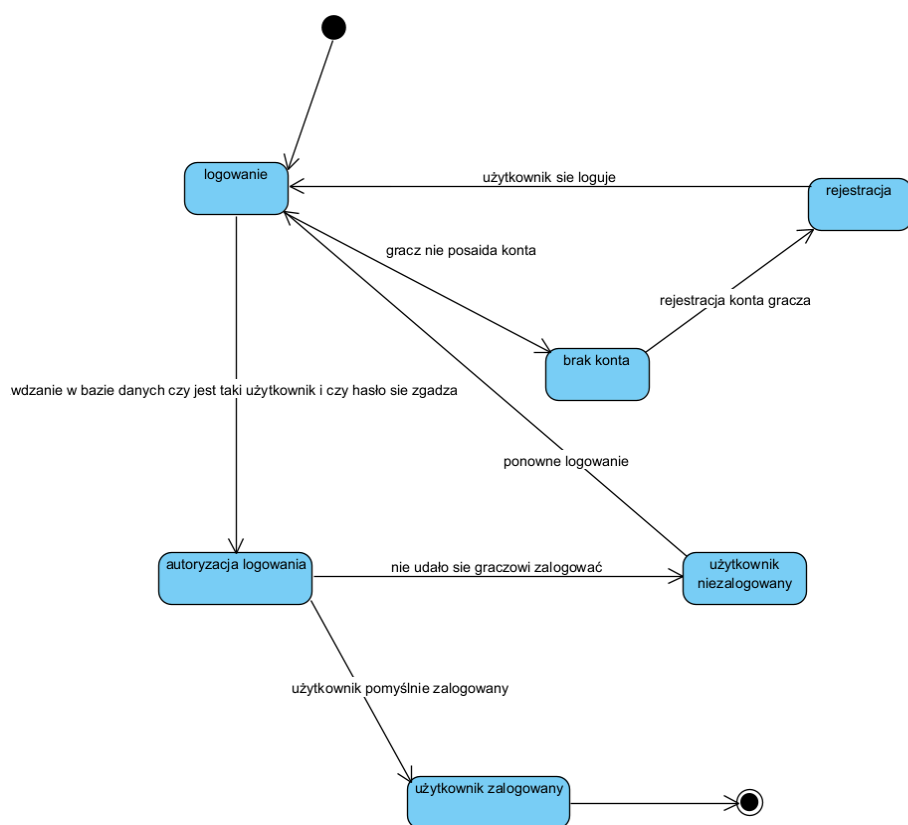
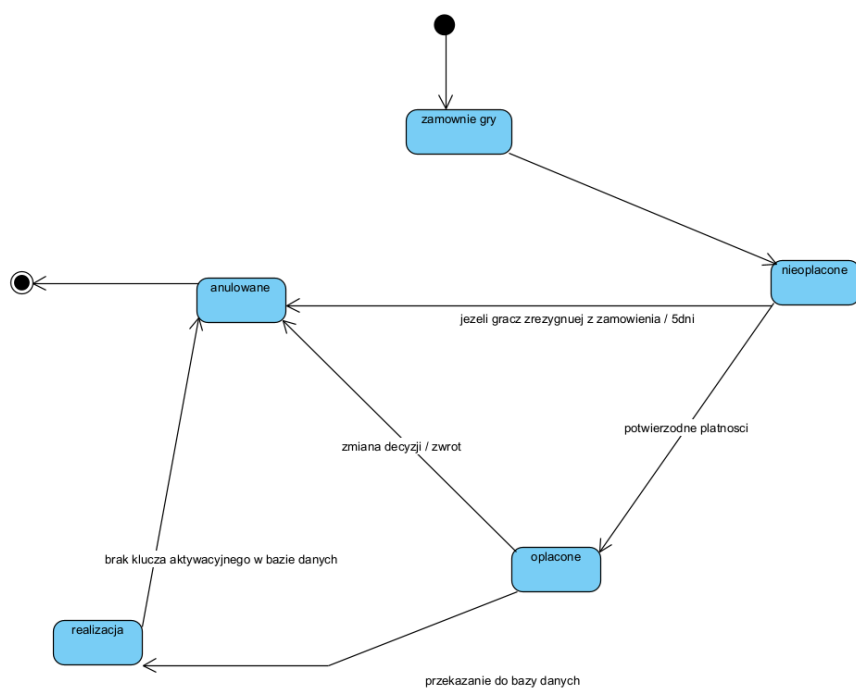






### 9.3.4 Inne diagramy

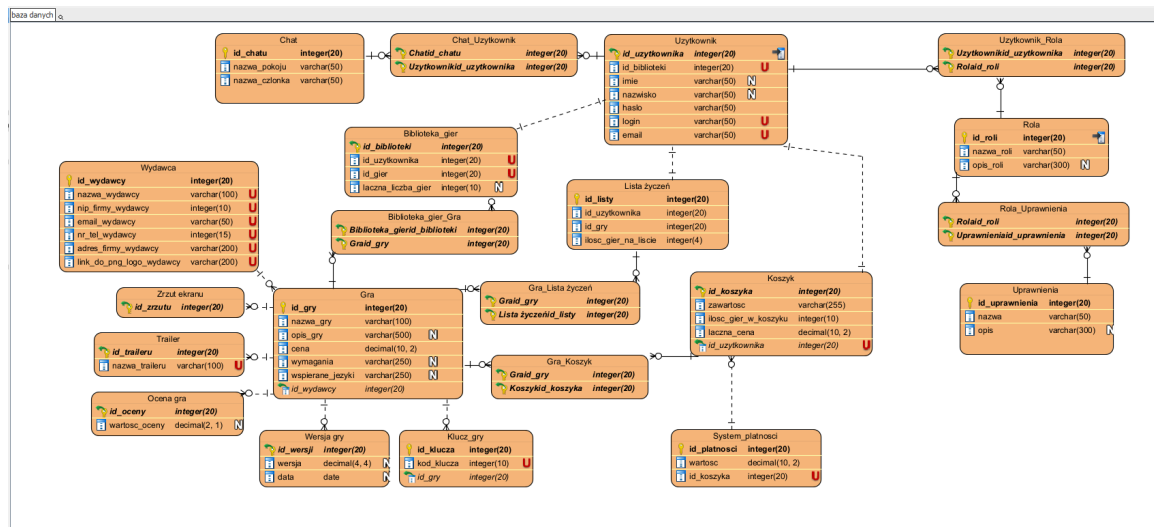
co najmniej trzy – komponentów, rozmieszczenia, maszyny stanowej itp.



## 9.4 Charakterystyka zastosowanych wzorców projektowych

informacja opisowa wspomagana diagramami (odsylaczami do diagramów UML); jeśli wykorzystano wzorce projektowe, to należy wykazać dwa z nich  
uwaga – wzorce projektowe nie są omawiane na wykładach!

## 9.5 Projekt bazy danych



### 9.5.1 Schemat

w trzeciej formie normalnej; jeśli w innej to umieć uzasadnić wybór

### 9.5.2 Projekty szczegółowe tabel

w zależności, czy następujące elementy są widoczne na schemacie b.d.: nazwa tabeli, nazwy pól, typ danych, wartości NULL, klucz główny, klucz obcy –

- jeśli TAK: i nie ma potrzeby pokazania dodatkowych elementów b.d., to ten punkt może być pusty,

- jeśli NIE: to podać te elementy, których nie widać na schemacie.

dodatkowymi elementami mogą być np. triggerzy, procedury, funkcje, indeksy, użytkownicy, role.

## 9.6 Projekt interfejsu użytkownika

co najmniej dla głównej funkcjonalności programu – w razie wątpliwości, uzgodnić z prowadzącym zajęcia

<https://github.com/TraperRoku/IO---app->

## 9.6.1 Lista głównych elementów interfejsu

okien, stron, aktywności (Android)

1) Okno startowe:

\*buttons:

- rejestracja
- logowanie

\*pola do wypełnienia: (JTextField)

- e-mail
- hasło

2) Okno rejestracji:

\*pola do wypełnienia:

- e-mail
- hasło
- telefon
- nickname

\*buttons:

- register
- log-in

3) Okno po zalogowaniu:

\*buttons:

- sing out
- wyszukaj
- dodaj kwotę do portfela
- dodaj do koszyka
- przejdź do koszyka

\*pola do wypełnienia:

- kwota (którą chcesz dodać)
- nazwa gry do wyszukania

\*inne element

- po wyszukaniu gry jej nazwa i grafika przedstawiająca ją

4) Okno koszyka:

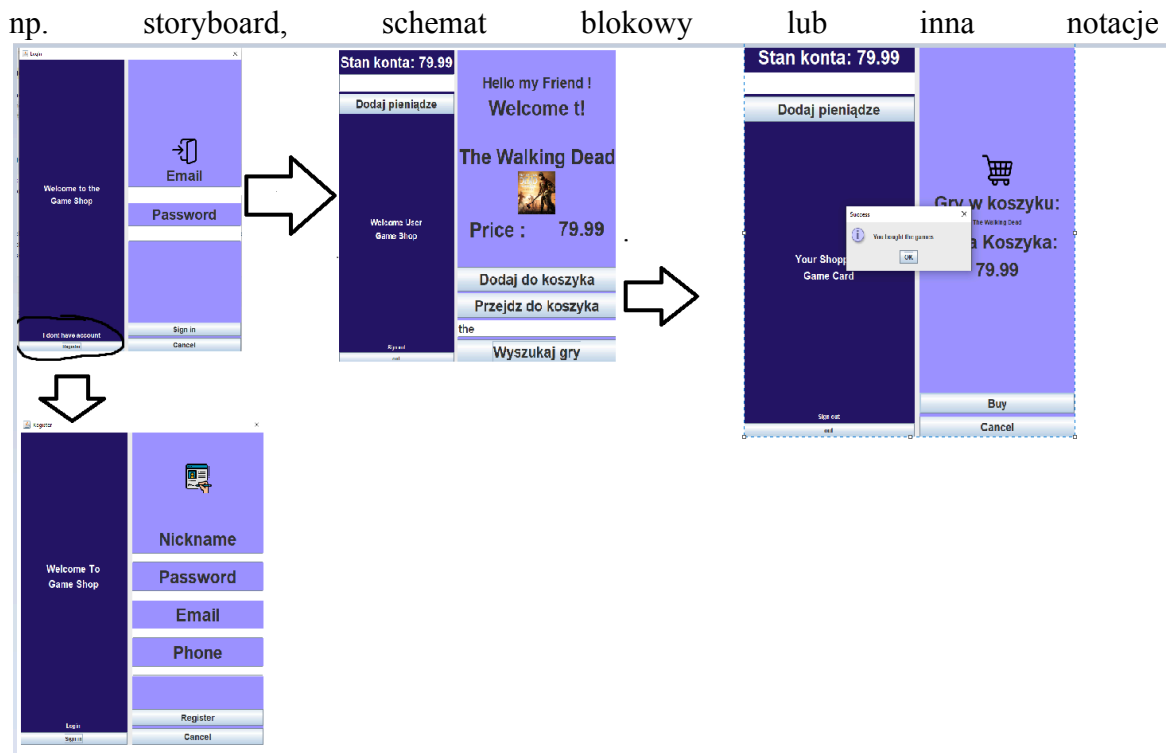
\*pola do wypełnienia:

- kwota (którą chcesz dodać)

\*buttons:

- dodaj kwotę do portfela
- buy
- cancel
- sing out

## 9.6.2 Przejścia między głównymi elementami




## 9.6.3 Projekty szczegółowe poszczególnych elementów

dla 5-7 głównych elementów (w zespołach 2-osobowych)

każdy element od nowej strony z następującą minimalną zawartością:


- numer – ID elementu
- nazwa – np. formularz danych produktu
- projekt graficzny – wystarczy schemat w narzędziu graficznym lub zrzut ekranu – z przykładowymi danymi (nie pusty!!!)
- opcjonalnie:
- opis – dodatkowe opcjonalne informacje o przeznaczeniu, obsłudze – jeśli nazwa nie będzie wystarczająco czytelna
- wykorzystane dane – jakie dane z bazy danych są wykorzystywane
- opis działania – tabela pokazująca m.in. co się dzieje po kliknięciu przycisku, wybraniu opcji z menu itp.

 Login

Welcome to the  
Game Shop

I dont have account

Register



Email

Password

Sign in

Cancel



**Stan konta: 79.99**

**Dodaj pieniądze**

Welcome User  
Game Shop

Sign out

out

Hello my Friend !

**Welcome t!**

**The Walking Dead**



**Price : 79.99**

**Dodaj do koszyka**

**Przejdź do koszyka**

the

**Wyszukaj gry**

**Stan konta: 79.99**

**Dodaj pieniądze**

Your Shopping  
Game Card

Sign out  
out



**Gry w koszyku:**

The Walking Dead

**a Koszyka:**  
**79.99**

**Buy**

**Cancel**


Success

×



You bought the games


OK

 Register ×

Welcome To  
Game Shop

Login

Sign in



Nickname

Password

Email

Phone

Register

Cancel

### 9.7 *Procedura wdrożenia*

jeśli informacje w harmonogramie nie są wystarczające (a zapewne nie są)

1. Pobranie aplikacji
2. Wstawienie aplikacji na serwer
3. Podłączenie z bazą danych
4. Uruchomienie serwera dla wszystkich użytkowników
5. Opublikowanie aplikacji

## **10. Dokumentacja dla użytkownika**

Opcjonalnie – dla chętnych

Na podstawie projektu docelowej aplikacji, a nie zaimplementowanego prototypu architektury

4-6 stron z obrazkami (np. zrzuty ekranowe, polecenia do wpisania na konsoli, itp.)

- pisana językiem odpowiednim do grupy odbiorców – czyli najczęściej nie do informatyków
- może to być przebieg krok po kroku obsługi jednej głównej funkcji systemu, kilku mniejszych, instrukcja instalacji lub innej pomocniczej czynności.

## 11. Podsumowanie

### *11.1 Szczegółowe nakłady projektowe członków zespołu*

tabela (kolumny to osoby, wiersze to działania) pokazująca, kto ile czasu poświęcił na projekt oraz procentowy udział każdej osoby w danym zadaniu oraz wiersz podsumowania – procentowy udział każdej osoby w skali całego projektu

Michał 25%

Cezary 25 %

Filip 25%

Kamil 25%

## **12. Inne informacje**

przydatne informacje, które nie zostały ujęte we wcześniejszych punktach