

# Dokumentacja projektu zaliczeniowego

Przedmiot: Inżynieria oprogramowania

Temat:	Dziennik elektroniczny
Autorzy:	Aleksander Grobicki-Madej, Maciej Wolf
Grupa:	I1-222A
Kierunek:	informatyka
Rok akademicki:	2016/2017
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.....	6
4.4	Analiza SWOT organizacji .....	6
5	Specyfikacja wymagań .....	7
5.1	Charakterystyka ogólna.....	7
5.2	Wymagania funkcjonalne.....	7
5.3	Wymagania нефункционалне.....	8
6	Zarządzanie projektem .....	9
6.1	Zasoby ludzkie .....	9
6.2	Etapy/kamienie milowe projektu .....	9
6.3	Harmonogram prac.....	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 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 .....	13
9.6	Projekt interfejsu użytkownika.....	13
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**

<https://github.com/Grommuald/IOProjekt> - odnośnik do strony repozytorium z projektem i dokumentacją

### **3 Słownik pojęć**

brak

## 4 Wprowadzenie

### 4.1 Cel dokumentacji

Dokumentacja zawiera szczegółowe informacje na temat całego projektu. Stanowi opis wszelkich obiektów i czynności występujących w systemie, jak również jako materiał do wglądu przez Prowadzącego z zajęć projektowych z Inżynierii Oprogramowania.

### 4.2 Przeznaczenie dokumentacji

Dokumentacja kierowana jest zarówno do uczestników projektu, jak i do Prowadzącego przedmiot.

### 4.3 Opis organizacji

System realizowany jest dla szkoły - jednostki oświatowo-wychowawczej zajmującej się, kształceniem i wychowaniem - zgodnie z podstawą programową i ustawą o oświacie. Szkoła opiera się na relacji nauczyciel-uczeń, co pozwala na systematyczne przekazywanie wiedzy ze strony nauczyciela, jak również relację zwrotną ze strony ucznia, w postaci odbioru przekazywanego materiału. Do prowadzenia zajęć niezbędny jest specjalny dokument zwany dziennikiem. W nim przechowywane są informacje o postępach każdego z uczniów, tj.: oceny, obecności czy uwagi. Pomaga to w weryfikacji postępów w nauce każdego ucznia, zarówno przez rodziców, jak i nauczycieli. Każdy z uczniów przydzielony jest również do określonej klasy – pomaga to w organizacji zajęć, dzieląc wszystkich uczniów uczęszczających do szkoły na mniejsze grupy. Co więcej, istnieje również podział ze względu na zaawansowanie w opanowanej wiedzy i wiek – stąd wyróżniamy podział na roczniki.

### 4.4 Analiza SWOT organizacji

Silne strony: cyfryzacja w dziedzinie edukacji	Szanse: pojawianie się nowych technologii na rynku edukacji
Słabe strony: zastój w technologii, w pełni analogowe zarządzanie placówką	Zagrożenia: reforma edukacyjna

## **5 Specyfikacja wymagań**

### **5.1 Charakterystyka ogólna**

#### **5.1.1 Definicja produktu**

System służy do komunikacji między uczniami a nauczycielami oraz rodzicami a nauczycielami. System dokumentuje w formie elektronicznej zajęcia lekcyjne w szkole.

#### **5.1.2 Cel biznesowy**

- ułatwienie komunikacji między instytucją szkoły a rodzicami
- ułatwiony wgląd w dokumenty
- ułatwiony wgląd w postępy w edukacji uczniów
- ułatwienie wypełniania obowiązków przez nauczycieli

#### **5.1.3 Użytkownicy**

- **uczeń** - sprawdza oceny, komunikuje się z nauczycielem
- **nauczyciel** - wystawia oceny, komunikuje się z uczniem i rodzicem, usprawiedliwia nieobecności ucznia, sprawdza obecność, wpisuje uwagi
- **administrator systemu** - udostępnia plan lekcji, przydziela nauczycieli do klas, zarządza systemem
- **rodzic** - sprawdza oceny, komunikuje się z nauczycielem, usprawiedliwia nieobecności ucznia

#### **5.1.4 Korzyści z systemu**

##### **1. Korzyści**

- 1.1 - uczeń - wie, kiedy ma odwołane zajęcia, zna swoje oceny, zawsze zna swoją aktualną średnią ocen
- 1.2 - rodzic - może na bieżąco kontrolować oceny swojego dziecka
- 1.3 - nauczyciel - ma łatwiejszy kontakt z uczniami i rodzicami, łatwiej prowadzi statystykę uczniów
- 1.4 – administrator – jest mu łatwiej zarządzać planami zajęć

#### **5.1.5 Ograniczenia projektowe i wdrożeniowe**

- sale lekcyjne muszą być wyposażone we własne komputery
- kadra musi być przeszkolona w zakresie korzystania z komputera i systemu

### **5.2 Wymagania funkcjonalne**

#### **5.2.1 Lista wymagań**

- system ma umożliwiać komunikację między uczniem a nauczycielem oraz rodzicem a nauczycielem
- system powinien pokazywać wszystkie dotychczas zgromadzone przez ucznia oceny, zarówno w panelu ucznia, jak i panelu rodzica oraz nauczyciela

- system powinien umożliwiać nauczycielowi sprawdzanie obecności na podstawie aktualnie odbywających się zajęć w planie
- system powinien zapewnić nauczycielowi wystawienie dowolnej oceny z prowadzonego przez siebie przedmiotu dowolnemu uczniowi, który przypisany jest na jego zajęcia
- system powinien wyświetlać aktualny dla ucznia plan zajęć
- system powinien wyświetlać aktualny dla nauczyciela plan zajęć
- system powinien pokazywać obecności ucznia na zajęciach w panelu ucznia oraz panelu rodzica danego ucznia
- system powinien udostępniać administratorowi panel kompozycji planu zajęć dla dowolnej klasy
- system powinien zapewnić administratorowi możliwość wykreślenia z planu na konkretny dzień poszczególnych zajęć, z uwagi na możliwość niepojawienia się nauczyciela na konkretnych zajęciach
- system powinien udostępniać nauczycielowi panel do komunikacji z administratorem, aby móc poinformować go o niedyspozycji

## **5.2.2 Diagramy przypadków użycia**

## **5.2.3 Szczegółowy opis wymagań**

### **5.3 *Wymagania niefunkcjonalne***

1. Wydajność
  - system powinien zapewnić stały i stabilny dostęp do elektronicznego dziennika
2. Bezpieczeństwo
  - system powinien zapewnić tworzenie kopii zapasowych informacji w razie błędów związanych z niepoprawnym wprowadzeniem informacji przez osobę z niego korzystającą
3. Zabezpieczenia
  - system powinien zapewnić tworzenie kopii zapasowych informacji w razie nieprzewidzianej awarii związanej z działaniem czynników zewnętrznych
4. Inne cechy jakości
  - system powinien być dostępny z każdego urządzenia z dostępem do Internetu



## **6 Zarządzanie projektem**

### ***6.1 Zasoby ludzkie***

### ***6.2 Etapy/kamienie milowe projektu***

### ***6.3 Harmonogram prac***

## **7 Zarządzanie ryzykiem**

### **7.1 *Lista czynników ryzyka***

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

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

## **8 Zarządzanie jakością**

### ***8.1 Scenariusze testowe***

## 9 Projekt techniczny

### 9.1 Opis architektury systemu

Architektura będzie opierać się na wzorcu MVC (Model-View-Controller), co ułatwi oddzielenie warstwy front-end od warstwy back-end systemu.

### 9.2 Technologie implementacji systemu

- 1) Java – ponieważ jest to bardzo przenośne rozwiązanie. Poza tym oferuje szeroką bibliotekę potrzebnych narzędzi, tj. Java FX czy rozwiązania sieciowe. Jest również bardzo szeroko stosowane w projektach wykorzystujących programowanie zorientowane obiektowo.
- 2) Heroku – proste rozwiązanie dla małych projektów; oferujące wsparcie dla baz danych; działa na zasadzie chmury.

### 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

#### 9.3.2 Diagram(-y) czynności

#### 9.3.3 Diagramy sekwencji

co najmniej 5, w tym co najmniej 1 przypadek użycia zilustrowany kilkoma diagramami sekwencji

#### 9.3.4 Inne diagramy

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

### 9.4 Charakterystyka zastosowanych wzorców projektowych

**Model-View-Controller** – wzorzec służący do organizowania struktury aplikacji posiadających graficzne interfejsy użytkownika.

**Model** – jest pewną reprezentacją problemu bądź logiki aplikacji,

**View** – opisuje, jak wyświetlić pewną część modelu w ramach interfejsu użytkownika,

**Controller** – przyjmuje dane wejściowe użytkownika i reaguje na jego poczynania, zarządzając aktualizacje modelu oraz odświeżanie widoków.

## ***9.5 Projekt bazy danych***

### **9.5.1 Schemat**

### **9.5.2 Projekty szczegółowe tabel**

## ***9.6 Projekt interfejsu użytkownika***

### **9.6.1 Lista głównych elementów interfejsu**

### **9.6.2 Projekty szczegółowe poszczególnych elementów**

## ***9.7 Procedura wdrożenia***

## **10 Dokumentacja dla użytkownika**

## **11 Podsumowanie**

### ***11.1 Szczegółowe nakłady projektowe członków zespołu***

## **12 Inne informacje**