Programowanie aplikacji internetowych - projekt  
Mateusz Śliwczyński, gr. 31E

# Dokumentacja techniczna

MobilePost  
aplikacja do obsługi firmy kurierskiej

Spis treści

[1. Specyfikacja wymagań 4](#_Toc400017921)

[1.1. Definicja produktu 4](#_Toc400017922)

[1.2. Technologia 4](#_Toc400017923)

[1.3. Wymagania funkcjonalne 4](#_Toc400017924)

[2. Projekt techniczny 5](#_Toc400017925)

[2.1. Projekt interfejsu użytkownika 5](#_Toc400017926)

[Tabela 1. Główne okno aplikacji 6](#_Toc400017927)

[Rysunek 2. Kolejny widok 6](#_Toc400017928)

[Tabela 2. Pasek menu – menu „Plik” 6](#_Toc400017929)

[Rysunek 3. Pasek menu – menu „Import/eksport” 7](#_Toc400017930)

[Tabela 3. Pasek menu – menu „Import/eksport” 7](#_Toc400017931)

[Rysunek 6. Okno „Import” 8](#_Toc400017932)

[Tabela 6. Okno „Import” 8](#_Toc400017933)

[Rysunek 8. Okno "Szukaj wg albumu" 8](#_Toc400017934)

[Tabela 8. Okno "Szukaj wg albumu" 8](#_Toc400017935)

[Rysunek 10. Okno "O aplikacji...” 9](#_Toc400017936)

[Tabela 10. Okno "O aplikacji...” 9](#_Toc400017937)

[Rysunek 13. Komunikat o błędzie 9](#_Toc400017938)

[Tabela 13. Komunikat o błędzie 9](#_Toc400017939)

[2.2. Opisy funkcjonalności 10](#_Toc400017940)

[Funkcjonalność otwierania plików do edycji 10](#_Toc400017941)

[Funkcjonalność wyświetlania tagów 10](#_Toc400017942)

[Funkcjonalność edycji tagów załadowanych plików 10](#_Toc400017943)

[Funkcjonalność edycji okładki załadowanych plików 10](#_Toc400017944)

[Funkcjonalność importowania tagów z pliku tekstowego 11](#_Toc400017945)

[2.3 Diagramy UML 11](#_Toc400017946)

[2.3.1. Diagram przypadków użycia 11](#_Toc400017947)

[2.3.2. Diagramy aktywności 12](#_Toc400017948)

[2.3.4. Diagramy sekwencji 13](#_Toc400017949)

[2.3.5. Diagram obiektów 14](#_Toc400017950)

[2.3.6. Diagram klas 14](#_Toc400017951)

[3. Scenariusze testowe 15](#_Toc400017952)

[4. Plan testów jednostkowych 16](#_Toc400017953)

# 1. Specyfikacja wymagań

## 1.1. Definicja produktu

Aplikacja przeznaczona jest dla firm kurierskich, które chcą podnieść jakość swoich usług, poprzez wprowadzenie zarządzania przesyłkami, pracownikami dostaw a także umożliwienie klientom internetowej obsługi ich zleceń.

## 1.2. Technologia

Aplikacja będzie napisana w języku PHP z wykorzystanie framework Symfony 3.0. Przy tworzeniu front endu aplikacji zostanie wykorzystany framework AngularJS.

## 1.3. Wymagania funkcjonalne

Jako nadawca mogę zgłosić przesyłkę przez stronę internetową.

Jako nadawca mogę ustalić termin wysłania przesyłki.

Jako nadawca mogę ustalić termin odebrania ode mnie przesyłki przez kuriera.

Jako nadawca mogę anulować wysłanie przesyłki.

Jako nadawca mogę dostawać powiadomienia o zmianie statusu mojej przesyłki.

Jako nadawca mogę przeglądać historię swoich przesyłek.

Jako nadawca mogę wyświetlać wszystkich swoich adresatów z historii.

Jako nadawca mogę śledzić swoje przesyłki.

Jako nadawca mogę zalogować się do aplikacji na swoje konto.

Jako odbiorca mogę śledzić adresowane do mnie przesyłki.

Jako odbiorca mogę ustalić termin dostarczenia do mnie przesyłki przez kuriera.

Jako odbiorca mogę przeglądać listę nadawców, którzy wysłali mi przesyłki.

Jako odbiorca mogę zalogować się do aplikacji na swoje konto.

Jako kurier mogę wyświetlić przypisane do mnie przesyłki.

Jako kurier mogę oznaczyć przesyłkę na liście jako odebraną.

Jako kurier mogę zalogować się do aplikacji na swoje konto.

Jako administrator mogę zarejestrować nowego kuriera.

Jako administrator mogę usunąć kuriera.

Jako administrator mogę wyświetlać i edytować zgłoszone przesyłki.

Jako administrator mogę sortować, filtrować i przeszukiwać listę zgłoszonych przesyłek.

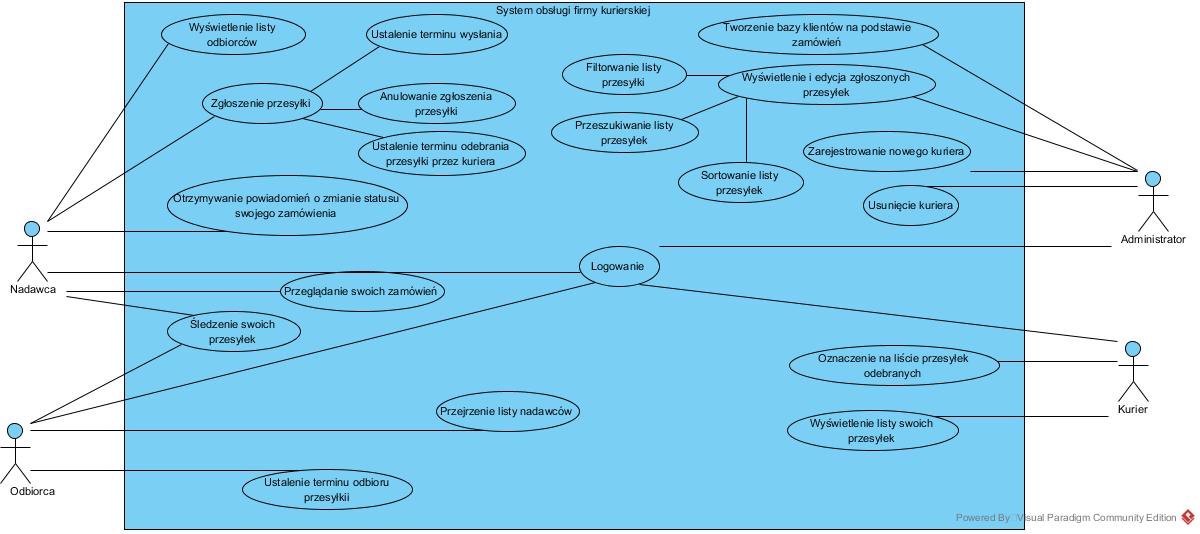
Jako administrator mogę tworzyć bazę danych klientów na podstawie złożonych zgłoszonych przesyłek.

# 2. Projekt techniczny

## 2.2. Opisy funkcjonalności

## 2.3 Diagramy UML

### 2.3.1. Diagram przypadków użycia



### 2.3.2 Diagramy aktywności

### 2.3.4. Diagramy sekwencji

### 2.3.5. Diagram obiektów

### 2.3.6. Diagram klas

# Scenariusze testowe

# Plan testów jednostkowych