

## 1. Wymagania funkcjonalne

Nazwa	Opis
Rejestracja	Klient ma możliwość utworzenia konta umożliwiające mu korzystanie z usług systemu.
Logowanie	Klient może zalogować się na swoje konto.
Konfiguracja	Klient ma możliwość konfiguracji swojego konta, w szczególności ustawienia i zmiany hasła.
Składanie zleceń	Klient może złożyć zlecenie uzupełniając odpowiednie pola w formularzu zgłoszeniowym.
Przeglądanie zleceń	Klient może przejrzeć swoje zlecenia (status, datę realizacji itp.).
Przydzielanie zleceń	Dyspozytor przydziela konkretnemu kierowcy zlecenia oczekujące na przydzielenie.
Usuwanie zleceń	Dyspozytor ma możliwość usunięcia nieprawidłowych zleceń.
Zarządzanie flotą kierowców	Administrator może dodawać, edytować i usuwać kierowców.
Przeglądanie przydzielonych zleceń	Dyspozytor może monitorować zlecenia przydzielone do kierowców.
Zmiana statusu zlecenia	Klient może zmienić status zlecenia na np. Dostarczone poprawnie
Wyświetlanie zleceń kierowcy	Kierowca może sprawdzić zlecenia, które zostały mu przydzielone.

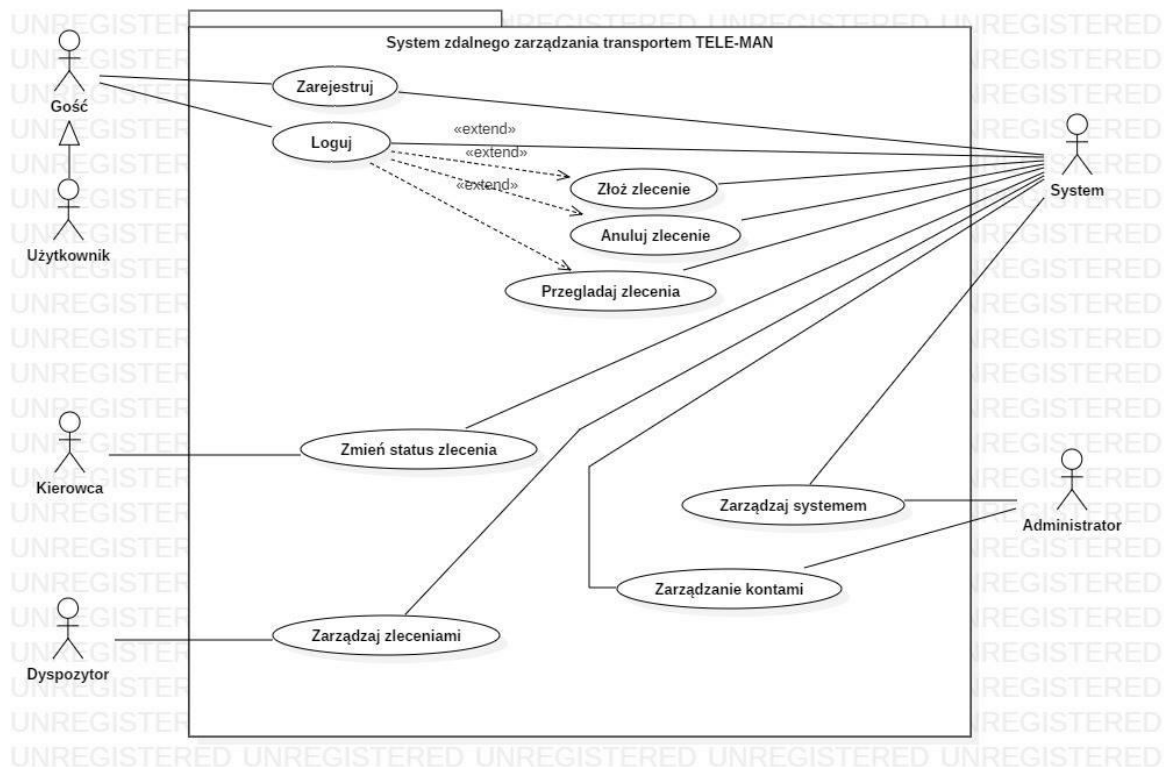
## 2. Wymagania niefunkcjonalne

- System będzie oferować wysoką wydajność przepływu danych, tzn. użytkownik nie będzie musiał czekać więcej niż pięć minut na powiadomienie o zmianie statusu
- Aplikacja będzie odświeżać listę zleceń w maksymalnie 10 sekund
- Do działania systemu wymagane jest połączenie z Internetem
- Aplikacja musi mieć interfejs przyjazny użytkownikowi, tzn. wszystkie potrzebne opcje będą wyprowadzone „na zewnątrz” interfejsu – wszystko w zasięgu ręki
- Aplikacja będzie zapewniać wysoki poziom bezpieczeństwa danych w systemie – dane logowania w bazie danych nie będą przechowywane w sposób jawny
- System będzie w jak najmniejszym stopniu podatny na awarie – komponenty będą korzystały z zaufanych i stabilnych rozwiązań, np. baza danych MySQL, FW .NET
- Strona internetowa musi pracować poprawnie na każdej z popularnych przeglądarek na urządzeniach typu desktop bądź na urządzeniach mobilnych
- Komunikacja aplikacji webowej, desktopowej oraz mobilnej będzie następować przez serwer bazodanowy
- System musi zapewniać dostępność ciągłą w systemie 24 godziny dziennie, 7 dni w tygodniu. Dopuszczalna jest chwilowa przerwa w pracy systemu.
- Aplikacja mobilna będzie używać minimum niezbędnego transferu danych do wymiany informacji z serwerem w celu oszczędzania danych pakietowych
- Aplikacja mobilna nie będzie zbyt obciążać zasobów sprzętowych urządzeń przenośnych.
- System będzie wymagał systemu Windows 7, 8.1 lub 10, Androida minimum 8.1 Oreo,

### 3. Przypadki użycia

Jako	Chcę	Żeby
Gość	Założyć konto	Móc się zalogować i mieć dostęp do panelu głównego aplikacji
Gość	Zalogować się	Korzystać w pełni z systemu i otrzymać uprawnienia do składania zleceń oraz ich przeglądania
Użytkownik	Wylogować się	Zakończyć korzystanie z systemu i zamknąć swoją sesję
Użytkownik	Złożyć zlecenie	Mogło zostać przydzielone do dalszej realizacji przez dyspozytora do konkretnego kierowcy
Użytkownik	Anulować zlecenie	Zatrzymać dalszą realizację niezatwierdzonego jeszcze zlecenia przez dyspozytora
Użytkownik	Przejrzeć historię zleceń	Uzyskać odpowiednie informacje o statusie zlecenia, jego parametrach oraz swoich danych osobowych
Kierowca	Przejrzeć przypisane zlecenia	Uzyskać odpowiednie informacje, aby móc analizować i realizować dane zlecenie
Kierowca	Zalogować się	Uzyskać dostęp do swojego panelu w celu realizacji zleceń

#### 4. Diagram przypadków użycia



Rysunek 1. – Diagram przypadków użycia

#### 5. Technologia

- Baza danych – MySQL
- Aplikacja webowa – PHP, SQL, Google Maps API, Bootstrap 4
- Aplikacja desktopowa – C# WPF .NET
- Aplikacja mobilna – Java/Kotlin

#### 6. Metodyka

Metodykę pracy, jaką przyjmujemy podczas realizacji projektu to XP – Extreme programming.

Cechy w/w metodyki to:

- Iteracyjność (krótkie kroki, bez planowania kolejnych iteracji).
- Brak ustalonej architektury – można zmieniać podczas trwania projektu
- Programowanie parami (koder i obserwator), zmiany co kilkadziesiąt minut – ułatwia jasność kodu.
- Stały kontakt z klientem (czasem zamiast specyfikacji).

#### 7. Podział pracy

- Lider grupy – Kamil Gawek
- Baza danych – Kamil Gawek
- Aplikacja webowa – Aleksandr Kowalczyk, Kamil Gawek
- Aplikacja desktopowa – Mateusz Janicki
- Aplikacja mobilna – Andrzej Baranowski

## 8. Harmonogram

- 15.04.2019 – Baza danych
- 13.05.2019 – Komunikacja, interfejs
- 27.05.2019 – Integracja
- 03.06.2019 – Dopracowanie projektu
- 10.06.2019 – Dokumentacja techniczna projektu
- 17.06.2019 – Gotowy projekt