

System Wypożyczalni Hulajnóg

Zaawansowana inżynieria oprogramowania

101650, 101651, 101649, 101641

1.1 oraz 1.2 Gromadzenie i analiza wymagań do systemu. Metoda historyjek użytkownika.....	2
1. Kto będzie używał systemu?.....	2
2. Jakie są role użytkowników?.....	2
3. W jakim celu będzie używany system?.....	2
4. Jakie procesy w instytucji powinien wspierać system?.....	2
5. Jak przebiega realizacja procesów, które powinien wspierać system?.....	3
6. Jaka jest skala działalności instytucji i jej potrzeb pod kątem przetwarzania danych....	3
7. Czy praca systemu wymaga składowania danych? Jeśli tak, to jakich?.....	3
8. Jak szczegółowe dane powinny być przechowywane?.....	3
9. Czy instytucja będzie chciała uruchomić system lokalnie, czy w chmurze?.....	3
10. Czy instytucja ma jakieś wymagania dotyczące technologii, w których nastąpi implementacja systemu?.....	4
11. Czy system powinien być przyjazny dla osób niepełnosprawnych?.....	5
12. Jaki jest czas realizacji systemu?.....	5
13. Jak sobie poradzić z potencjalnymi żądaniami wprowadzenia modyfikacji wymagań do systemu na etapie implementacji systemu?.....	5
1.3 Zidentyfikowanie aktorów.....	5
1.4 Historyjki użytkownika (przykłady, pokrywają funkcjonalność).....	6
1.5 Lista wymagań funkcjonalnych.....	7
1.6 Lista wymagań niefunkcjonalnych.....	7
1.7 Zidentyfikowanie istotnych obiektów.....	8
1.8 Dokument ze specyfikacją systemu – proponowana struktura i wypełnienie.....	8
2.1 Weryfikacja listy aktorów.....	9
2.2 Identyfikacja przypadków użycia (z podziałem na aktorów).....	10
Aktor: Użytkownik (Klient) – niezalogowany.....	10
Aktor: Użytkownik (Klient) – zalogowany.....	10
Aktor: Administrator systemu.....	11
Aktor: Serwisant.....	11
Aktorzy Zewnętrzni (Systemy).....	12
Aktor: System płatności (zewnętrzny).....	12
Aktor: Usługa map/GPS (zewnętrzna).....	12
Aktor: Usługa powiadomień (zewnętrzna).....	12
2.3. Specyfikacja scenariuszy przypadków użycia.....	13
S1. Zapoznawanie się z cennikiem i regulaminem.....	13
S2. Rejestracja nowego konta.....	15
S3. Logowanie do systemu.....	17
S4. Resetowanie hasła.....	19
S5. Rozpoczynanie przejazdu.....	21

S6. Ukończenie przejazdu.....	23
S7. Zarządzanie metodami płatności.....	25
S8. Przeglądanie historii przejazdów.....	27
S9. Doładowanie portfela.....	28
S10. Zgłaszczenie usterki hulajnogi.....	29
S11. Zarządzanie profilem.....	30
S12. Lokalizowanie hulajnogi dźwiękiem.....	32
S13. Zarządzanie kontami użytkowników.....	33
S14. Zarządzanie flotą hulajnóg.....	34
S15. Zarządzanie taryfami i cennikiem.....	35
S16. Zarządzanie strefami.....	36
S17. Generowanie raportów i statystyk.....	38
S18. Zarządzanie zgłoszeniami serwisowymi.....	39
S19. Przeglądanie listy zadań serwisowych.....	41
S20. Aktualizacja statusu hulajnogi.....	43
Diagram 1: S2. Rejestracja nowego konta.....	45
Diagram 2: S3. Logowanie do systemu.....	46
Diagram 3: S9. Doładowanie portfela.....	47
Diagram 4: Rozpoczynanie przejazdu.....	47
Diagram 5: Ukończenie przejazdu.....	48
Diagram 6: Zgłaszczenie usterki hulajnogi.....	50
Diagram 10: S18. Zarządzanie zgłoszeniami serwisowymi.....	52
Diagram 11: S19. Przeglądanie listy zadań serwisowych.....	53
Diagram 12: S20 – Aktualizacja statusu hulajnogi.....	54

1. Gromadzenie i analiza wymagań do systemu. Metoda historyjek użytkownika.

1. Kto będzie używał systemu?

- **Administrator systemu**
- **Klienci indywidualni** (użytkownicy hulajnóg)
- **Serwisanci** (osoby odpowiedzialne za konserwację hulajnóg)

2. Jakie są role użytkowników?

- **Użytkownik (klient)** – rejmuje konto, lokalizuje hulajnogę, wypożycza ją i dokonuje płatności.
- **Administrator** – zarządza kontami użytkowników, monitoruje stan floty hulajnóg, analizuje dane.
- **Serwisant** – przegląda zgłoszenia o awariach, serwisuje hulajnogi, aktualizuje ich status.

3. W jakim celu będzie używany system?

Celem systemu jest umożliwienie użytkownikom wypożyczanie hulajnóg elektrycznych na określony czas, z automatycznym naliczaniem opłat i monitorowaniem lokalizacji pojazdów w czasie rzeczywistym.

4. Jakie procesy w instytucji powinien wspierać system?

- Rejestrację i logowanie użytkowników
- Wypożyczanie i zwracanie hulajnóg
- Naliczanie opłat za przejazdy
- Śledzenie lokalizacji hulajnóg (GPS)
- Zgłaszcenie awarii hulajnóg
- Zarządzanie flotą hulajnóg (dodawanie, usuwanie, konserwacja)

5. Jak przebiega realizacja procesów, które powinien wspierać system?

1. Użytkownik loguje się do aplikacji
2. System pokazuje dostępne hulajnogi na mapie
3. Użytkownik skanuje kod QR hulajnogi i rozpoczyna przejazd
4. System nalicza opłatę w zależności od czasu i dystansu
5. Po zakończeniu jazdy użytkownik blokuje hulajnogę i dokonuje płatności
6. Serwisant widzi zgłoszenia o awariach i aktualizuje status pojazdów
7. Administrator nadzoruje cały system – statystyki, zgłoszenia, finanse

6. Jaka jest skala działalności instytucji i jej potrzeb pod kątem przetwarzania danych

System jest przewidziany dla średniej skali działalności - np. miasta wojewódzkiego. Dane obejmują setki hulajnóg i tysiące użytkowników, więc potrzebne jest skalowalne rozwiązanie (baza danych i serwery obsługujące wiele zapytań jednocześnie).

7. Czy praca systemu wymaga składowania danych? Jeśli tak, to jakich?

Tak. System ma przechowywać:

- Dane użytkowników (imię, nazwisko, e-mail, numer telefonu, historia przejazdów)
- Dane hulajnóg (numer seryjny, lokalizacja GPS, poziom baterii, status techniczny)
- Dane płatności (transakcje, kwoty, czas trwania przejazdu)
- Dane serwisowe (zgłoszenia, naprawy)

8. Jak szczegółowe dane powinny być przechowywane?

- Dane lokalizacji z dokładnością do kilku metrów, aktualizowane co kilka sekund
- Dane o przejazdach (czas startu, zakończenia, trasa, koszt)
- Historia kont użytkowników i ich aktywności

9. Czy instytucja będzie chciała uruchomić system lokalnie, czy w chmurze?

System powinien być uruchomiony **w chmurze**, aby:

- umożliwić dostęp z aplikacji mobilnych
- zapewnić skalowalność
- ułatwić integrację z systemami płatności i map

10. Czy instytucja ma jakieś wymagania dotyczące technologii, w których nastąpi implementacja systemu?

Klient (aplikacja mobilna): React Native (Expo)

Backend i Mikroserwisy: Node.js / Go

Baza danych: PostgreSQL, Redis

Usługi w chmurze: AWS / CloudFlare (np. do przechowywania danych GPS)

Usługi lokalizacyjne real time: Apache Kafka

11. Czy system powinien być przyjazny dla osób niepełnosprawnych?

Nie. Ze względu na charakter projektu, nie przewiduje się wdrażania specjalnych funkcji dla osób niepełnosprawnych.

12. Jaki jest czas realizacji systemu?

- **3 miesiące – wersja MVP:**
 - Rejestracja i logowanie użytkowników
 - Wyświetlanie dostępnych hulajnóg na mapie
 - Rozpoczęcie i zakończenie przejazdu
 - Podstawowy system płatności
- **5 miesiące – wersja rozszerzona:**
 - Zgłaszczenie usterek
 - Panel administracyjny (zarządzanie flotą hulajnóg)
 - Monitoring GPS i raporty
- **8 miesięcy – pełna wersja systemu:**
 - System powiadomień (e-mail i powiadomienia w aplikacji)
 - Panel serwisanta z historią napraw
 - Integracja z zewnętrznymi usługami map i płatności
 - Testy wydajności i optymalizacja

13. Jak sobie poradzić z potencjalnymi żądaniami wprowadzenia modyfikacji wymagań do systemu na etapie implementacji systemu?

- Wprowadzić system zarządzania zmianami oraz kontroli wersji - każda zmiana wymaga zatwierdzenia przez project leader'a (Github i Linear)
- Cele projektu będą definiowane i realizowane w oparciu o krótkoterminowe etapy (sprinty) oraz przypisane do nich zadania (Linear)
- Stosować model kaskadowy (waterfall) z kontrolą zmian, system rozwija się etapami: analiza → projekt → implementacja → testy → wdrożenie

1.3 Zidentyfikowanie aktorów

- Klient indywidualny (użytkownik hulajnóg).
- Administrator systemu.
- Serwisant.
- Zewnętrzne systemy: operator płatności, usługi map/GPS, system powiadomień e-mail/push.
- Hierarchia: Użytkownik niezalogowany → Użytkownik zalogowany; Pracownik → Serwisant → Administrator.

1.4 Historyjki użytkownika (przykłady, pokrywają funkcjonalność)

Klient:

1. Jako klient chce założyć konto, aby korzystać z aplikacji.
2. Jako klient chce się logować, aby mieć dostęp do swoich danych.
3. Jako klient chce dodać metodę płatności, aby opłacić przejazdy.
4. Jako klient chce widzieć na mapie dostępne hulajnogi, aby wybrać najbliższą.
5. Jako klient chce skanować kod QR, aby odblokować hulajnogę.
6. Jako klient chce rozpocząć i zakończyć przejazd w aplikacji, aby system naliczył opłatę.
7. Jako klient chce podglądu kosztu przejazdu w czasie rzeczywistym, aby kontrolować wydatki.
8. Jako klient chce zapłacić po zakończeniu jazdy, aby sfinalizować wypożyczenie.
9. Jako klient chce historii przejazdów i płatności, aby mieć rozliczenia.
10. Jako klient chce zgłosić usterkę hulajnogi, aby serwis ją naprawił.
11. Jako klient chce powiadomień o stanie konta i płatności, aby być na bieżąco.

Serwisant:

1. Jako serwisant chcę listy zgłoszeń z priorytetami, aby planować naprawy.
2. Jako serwisant chcę widzieć stan techniczny i poziom baterii, aby decydować o serwisie.
3. Jako serwisant chcę aktualizować status hulajnogi (sprawna/serwis), aby użytkownicy widzieli realną dostępność.
4. Jako serwisant chcę rejestrować wykonane naprawy, aby utrzymać historię serwisową.

Administrator

5. Jako administrator chcę zarządzać kontami użytkowników i serwisantów, aby
6. kontrolować dostęp.
7. Jako administrator chcę dodawać/usuwać hulajnogi i ich parametry, aby utrzymywać flotę.
8. Jako administrator chcę konfigurować taryfy i promocje, aby sterować cenami.
9. Jako administrator chcę raportów z GPS, przychodów i wykorzystania floty, aby analizować wyniki.
10. Jako administrator chcę definiować strefy działania i parkowania, aby egzekwować zasady.

1.5 Lista wymagań funkcjonalnych

- Rejestracja i logowanie użytkowników.
- Zarządzanie profilem i metodami płatności.
- Mapy z dostępnością hulajnóg w czasie rzeczywistym.
- Skanowanie QR i zdalne blokowanie/odblokowanie.
- Rozpoczęcie/zakończenie przejazdu.
- Naliczanie opłat wg czasu/dystansu/taryfy, podgląd kosztu.
- Płatność bezgotówkowa i wystawianie potwierdzeń.
- Historia przejazdów i transakcji.
- Zgłaszanie usterek z opisem i zdjęciem.
- Panel serwisowy: lista zgłoszeń, statusy, historia napraw.
- Panel administracyjny: zarządzanie flotą, użytkownikami, taryfami, strefami.
- Monitoring GPS, poziom baterii, status techniczny.
- Powiadomienia e-mail/push (płatności, blokady, serwis).
- Integracje: operator płatności, mapy/GPS, system powiadomień.
- Raporty i eksport danych (CSV/PDF).

1.6 Lista wymagań niefunkcjonalnych

- Wydajność: aktualizacja lokalizacji co 5–10 s.
- Dostępność: 99,5% miesięcznie dla usług kluczowych.
- Skalowalność: setki hulajnóg, tysiące użytkowników równocześnie.
- Bezpieczeństwo: OAuth2.ae
- Prywatność/RODO: minimalizacja danych, zgody, anonimizacja raportów, prawo do usunięcia.
- Niezawodność: odporność na awarie stref w chmurze, kopie zapasowe.
- Użyteczność: prosty UI mobilny, dostępność WCAG w aplikacji administracyjnej.
- Zgodność: iOS/Android.
- Obsługa błędów i logowanie: centralny logging, śledzenie zdarzeń audytowych.
- Lokalizacja: PL jako podstawowy język, przygotowanie do wielojęzyczności.

- Utrzymanie: CI/CD, testy jednostkowe i e2e, obserwowalność (metryki, alerty).

1.7 Zidentyfikowanie istotnych obiektów

- Użytkownik: id, imię, nazwisko, e-mail, telefon, status konta, data rejestracji.
- MetodaPłatności: id, użytkownikId, typ, token, status.
- Hulajnoga: id, numerSeryjny, model, status, poziomBaterii, ostatniaLokalizacja, dataAktualizacji.
- Lokalizacja: hulajnogald, lat, lon, czasPomiaru, dokładność.
- Przejazd: id, użytkownikId, hulajnogald, startCzas, startLatLon, stopCzas, stopLatLon, dystans, koszt, taryfald, status.
- Taryfa: id, nazwa, opłataStartowa, stawkaCzas, stawkaDystans, zasadyPromocji.
- Płatność: id, przejazdId, użytkownikId, kwota, waluta, status, czas, identyfikatorOperacji.
- ZgłoszenieUsterki: id, hulajnogald, zgłaszającyId, opis, zdjęcieUrl, priorytet, status, czasUtworzenia, czasZamknięcia.
- Naprawa: id, zgłoszeniId, serwisantId, czynności, części, koszt, czas.
- UżytkownikRola: użytkownikId, rola (Klient/Serwisant/Administrator).
- Powiadomienie: id, użytkownikId, typ, treść, kanał, status, czasWysłania.

Opis przetwarzania: obiekty wspierają rejestrację, wypożyczanie, naliczanie opłat, serwis i raportowanie. Atrybuty umożliwiają pełny audyt i integrację.

1.8 Dokument ze specyfikacją systemu – proponowana struktura i wypełnienie

1. Tytuł projektu: Wypożyczalnia hulajnóg – system mobilny i panel administracyjny.
2. Dane grupy projektowej: zgodnie z wytycznymi prowadzącego.
3. Podział czynności: role i zakresy prac członków zespołu.
4. Krótka charakterystyka systemu: cel, adresaci, kluczowa funkcjonalność, innowacje (taryfy dynamiczne, strefy parkowania).
5. Specyfikacja aktorów: Klient, Serwisant, Administrator, system płatności, mapy/GPS, powiadomienia.
6. Historyki użytkownika: pozycje 1–20 z pkt 1.4 z kryteriami akceptacji.
7. Wymagania funkcjonalne: lista z pkt 1.5.
8. Wymagania niefunkcjonalne: lista z pkt 1.6.
9. Obiekty istotne i atrybuty: model z pkt 1.7, diagram ER/UML.
10. Plan realizacji: MVP 3 mies., rozszerzenie 5 mies., pełna wersja 8 mies.
11. Architektura i technologie: React Native, Node.js/Go, PostgreSQL, Redis, chmura AWS/Cloudflare, Kafka dla strumieni lokalizacji.
12. Integracje i bezpieczeństwo: operator płatności, mapy, RBAC, RODO, audyt.
13. Testy i wdrożenie: CI/CD, testy jednostkowe/e2e, monitoring.
14. Zarządzanie zmianą: kontrola wersji i akceptacja zmian (GitHub/Linear), sprinty.

2.1 Weryfikacja listy aktorów

Użytkownik (Klient) – niezalogowany

- Osoba niezalogowana, ma dostęp jedynie do opcji zalogowania się, lub rejestracji nowego konta.

Użytkownik (Klient) – zalogowany

- Zarejestrowany klient korzystający z aplikacji mobilnej: skanuje QR, rozpoczyna i kończy przejazdy, dodaje metodę płatności, przegląda historię przejazdów oraz zgłasza usterki.

Administrator systemu

- Osoba administrująca systemem: zarządza kontami użytkowników i serwisantów, konfiguracją taryf, strefami parkowania, przegląda raporty i statystyki oraz wykonuje operacje utrzymaniowe na bazie danych i konfiguracji.

Serwisant

- Pracownik techniczny odpowiedzialny za obsługę zgłoszeń serwisowych, naprawę hulajnóg, aktualizację statusów urządzeń i rejestrowanie wykonanych prac.

System płatności (zewnętrzny)

- Operator płatności (np. Stripe, PayU) przetwarzający transakcje finansowe i zwracający statusy płatności.

Usługa map/GPS (zewnętrzna)

- Dostawca map i usług lokalizacyjnych (np. Mapbox, Google Maps) zapewniający wyświetlanie mapy, geokodowanie i routing oraz (opcjonalnie) strumień lokalizacji.

Usługa powiadomień (zewnętrzna)

- System wysyłający powiadomienia e-mail i push (np. Firebase Cloud Messaging, SendGrid) – powiadomienia o płatnościach, blokadach, przypomnieniach.

2.2 Identyfikacja przypadków użycia (z podziałem na aktorów)

Aktor: Użytkownik (Klient) – niezalogowany

Osoba, która uruchomiła aplikację, ale nie jest zalogowana na swoje konto.

- **PU-01: Zapoznawanie się z cennikiem i regulaminem**
 - **Opis:** Pozwala na dostęp do ekranów informacyjnych zawierających aktualne taryfy za wynajem, opłaty dodatkowe oraz pełny regulamin korzystania z usługi.
- **PU-02: Rejestracja nowego konta**
 - **Opis:** Użytkownik może utworzyć nowe konto, podając wymagane dane (np. e-mail, hasło, numer telefonu) i akceptując warunki korzystania z serwisu.
- **PU-03: Logowanie do systemu**
 - **Opis:** Umożliwia zalogowanie się na istniejące konto za pomocą podanych wcześniej danych uwierzytelniających w celu uzyskania dostępu do pełnej funkcjonalności aplikacji.
- **PU-04: Resetowanie hasła**
 - **Opis:** W przypadku zapomnienia hasła, użytkownik może zainicjować proces jego odzyskiwania, zazwyczaj poprzez link wysłany na powiązany z kontem adres e-mail.

Aktor: Użytkownik (Klient) – zalogowany

Zarejestrowany użytkownik, który jest uwierzytelniony w aplikacji mobilnej.

- **PU-05: Rozpoczynanie przejazdu**
 - **Opis:** Umożliwia odblokowanie hulajnogi poprzez zeskanowanie jej kodu QR, co rozpoczyna naliczanie opłat za przejazd zgodnie z obowiązującą taryfą.
- **PU-06: Ukończenie przejazdu**
 - **Opis:** Pozwala na zakończenie wynajmu, co wiąże się z zablokowaniem hulajnogi i automatycznym obciążeniem podpiętej metody płatności za odbytą podróż.
- **PU-07: Zarządzanie metodami płatności**
 - **Opis:** Użytkownik może dodawać, usuwać lub zmieniać domyślną kartę płatniczą, która będzie używana do regulowania należności za przejazdy.
- **PU-08: Przeglądanie historii przejazdów**
 - **Opis:** Daje dostęp do listy wszystkich historycznych przejazdów wraz ze szczegółami, takimi jak trasa na mapie, czas trwania, pokonany dystans i poniesione koszty.
- **PU-09: Doładowanie portfela**
 - **Opis:** Umożliwia zasilenie wewnętrznego portfela w aplikacji predefiniowaną lub dowolną kwotą, która może być później wykorzystana na opłacenie przejazdów.
- **PU-10: Zgłaszcanie usterki hulajnogi**

- **Opis:** Pozwala na zgłoszenie problemu technicznego z hulajnogą (np. uszkodzone hamulce, niska bateria), co tworzy zgłoszenie serwisowe w systemie.
- **PU-11: Zarządzanie profilem**
 - **Opis:** Umożliwia edycję danych osobowych, zmianę hasła oraz zarządzanie ustawieniami konta, takimi jak preferencje dotyczące powiadomień.
- **PU-12: Lokalizowanie hulajnogi dźwiękiem**
 - **Opis:** Użytkownik może zdalnie uruchomić sygnał dźwiękowy na wybranej hulajnodze, aby łatwiej ją zlokalizować w pobliżu.

Aktor: Administrator systemu

Użytkownik z najwyższymi uprawnieniami, zarządzający całą platformą.

- **PU-13: Zarządzanie kontami użytkowników**
 - **Opis:** Administrator może przeglądać, blokować, usuwać i edytować konta klientów oraz serwisantów w systemie.
- **PU-14: Zarządzanie flotą hulajnóg**
 - **Opis:** Umożliwia dodawanie nowych hulajnóg do systemu, edytowanie ich danych oraz zdalne zmienianie ich statusu (np. aktywna, wyłączona z użytku, w serwisie).
- **PU-15: Zarządzanie taryfami i cennikiem**
 - **Opis:** Pozwala na konfigurowanie opłat za odblokowanie, stawek za minutę jazdy oraz cen pakietów i subskrypcji dostępnych dla użytkowników.
- **PU-16: Zarządzanie strefami**
 - **Opis:** Administrator definiuje na mapie strefy operacyjne, takie jak strefy parkowania, strefy zakazu parkowania oraz strefy ograniczonej prędkości.
- **PU-17: Generowanie raportów i statystyk**
 - **Opis:** Umożliwia tworzenie i przeglądanie zbiorczych raportów dotyczących m.in. liczby przejazdów, przychodów, obłożenia hulajnogą czy najpopularniejszych tras.
- **PU-18: Zarządzanie zgłoszeniami serwisowymi**
 - **Opis:** Dajegląd we wszystkie zgłoszenia usterek od użytkowników i umożliwia przypisywanie ich do konkretnych serwisantów.

Aktor: Serwisant

Pracownik techniczny odpowiedzialny za utrzymanie floty hulajnóg.

- **PU-19: Przeglądanie listy zadań serwisowych**
 - **Opis:** Serwisant ma dostęp do listy przypisanych mu zadań, takich jak naprawa zgłoszonej usterki, wymiana baterii, zamawianie części zepsutej hulajnogi.
- **PU-20: Aktualizacja statusu hulajnogi**
 - **Opis:** Po wykonaniu zadania serwisant zmienia status hulajnogi w systemie (np. z "W naprawie" na "Dostępna"), aby ponownie udostępnić ją użytkownikom.

- **PU-21: Rejestrowanie wykonanych czynności**
 - **Opis:** Umożliwia serwisantowi dodanie notatki lub oznaczenie wykonanych prac w ramach zamykanego zgłoszenia serwisowego.

Aktorzy Zewnętrzni (Systemy)

Systemy, z którymi nasza aplikacja się komunikuje. Przypadki użycia opisują interakcję z perspektywy naszego systemu.

Aktor: System płatności (zewnętrzny)

- **PU-22: Przetwarzanie płatności za przejazd**
 - **Opis:** System wysyła żądanie obciążenia karty klienta na określoną kwotę po zakończeniu przejazdu i otrzymuje potwierdzenie lub odrzucenie transakcji.
- **PU-23: Autoryzacja i tokenizacja karty płatniczej**
 - **Opis:** Podczas dodawania nowej metody płatności system bezpiecznie przesyła dane karty do operatora, który zwraca unikalny token do późniejszych obciążzeń.

Aktor: Usługa map/GPS (zewnętrzna)

- **PU-24: Dostarczanie danych mapowych**
 - **Opis:** System pobiera od dostawcy kafelki mapy oraz inne dane geograficzne, aby wyświetlić interaktywną mapę w aplikacji mobilnej.
- **PU-25: Weryfikacja położenia w strefie**
 - **Opis:** System wysyła współrzędne geograficzne hulajnogi do usługi mapowej w celu sprawdzenia, czy znajduje się ona wewnątrz zdefiniowanej strefy (np. dozwolonego parkowania).

Aktor: Usługa powiadomień (zewnętrzna)

- **PU-26: Wysyłanie powiadomień**
 - **Opis:** System zleca zewnętrznej usłudze wysłanie powiadomienia (push, e-mail lub SMS) do użytkownika, np. z podsumowaniem przejazdu lub informacją o niskim stanie środków.

2. Modelowanie wymagań - scenariusze przypadków użycia

S1. Zapoznawanie się z cennikiem i regulaminem

S1.1. Opis

Scenariusz opisuje możliwość przeglądania przez użytkownika niezalogowanego informacji o aktualnych taryfach, kosztach wypożyczeń i obowiązującym regulaminie. Funkcjonalność ta ma charakter informacyjny i nie wymaga logowania.

S1.2. Aktorzy

Użytkownik (niezalogowany), System aplikacji mobilnej.

S1.3. Warunki początkowe

- Aplikacja została uruchomiona na urządzeniu mobilnym.
- Użytkownik nie jest zalogowany.
- Połączenie z Internetem jest aktywne.

S1.4. Warunki końcowe

- Użytkownik zapoznał się z treścią cennika i/lub regulaminu.
- System zakończył prezentację informacji bez błędów.

S1.5. Przebieg główny

1. Użytkownik wybiera z ekranu startowego opcję „Cennik” lub „Regulamin”.
2. System pobiera aktualne dane z serwera (cennik, regulamin).
3. System wyświetla treść na ekranie w postaci przewijalnej listy lub dokumentu.
4. Użytkownik przegląda dokument, może powiększać i przewijać treść.
5. Użytkownik może powrócić do ekranu głównego.

S1.6. Przebiegi alternatywne

PA1. Użytkownik wybiera „Cennik”, ale chce też zobaczyć „Regulamin”.
→ System umożliwia przełączenie zakładek bez powrotu do menu głównego.

S1.7. Sytuacje wyjątkowe

SW1. Brak połączenia z serwerem – system wyświetla komunikat „Nie udało się pobrać danych. Sprawdź połączenie z Internetem.”

SW2. Błąd ładowania danych – system prezentuje ostatnio zapisaną wersję offline (jeśli dostępna).

S1.8. Wymagania niefunkcjonalne

- Treść powinna być czytelna na ekranach o różnych rozdzielczościach.
- Strona powinna ładować się w czasie <3 sekundy przy stabilnym łączu.

- Dane cennika muszą być zgodne z wersją obowiązującą w panelu administracyjnym.

S1.9. Uwagi i pytania otwarte

- Czy aplikacja ma udostępniać plik PDF regulaminu do pobrania?
- Czy wymagana jest wersja językowa EN?

S2. Rejestracja nowego konta

S2.1. Opis

Scenariusz opisuje proces tworzenia nowego konta użytkownika, obejmujący wprowadzenie danych osobowych, ustawienie hasła oraz akceptację regulaminu.

S2.2. Aktorzy

Użytkownik (niezalogowany), **System aplikacji mobilnej, System e-mail (zewnętrzny)**.

S2.3. Warunki początkowe

- Aplikacja jest uruchomiona.
- Użytkownik nie jest zalogowany.
- System ma dostęp do Internetu.

S2.4. Warunki końcowe

- Konto użytkownika zostało utworzone.
- Na adres e-mail użytkownika wysłano wiadomość potwierdzającą rejestrację.

S2.5. Przebieg główny

1. Użytkownik wybiera opcję „Zarejestruj się”.
2. System wyświetla formularz rejestracyjny.
3. Użytkownik wprowadza dane: imię, nazwisko, adres e-mail, hasło, numer telefonu.
4. Użytkownik akceptuje regulamin i politykę prywatności.
5. System waliduje dane wejściowe.
6. System tworzy nowe konto użytkownika.
7. System wysyła wiadomość e-mail z linkiem aktywacyjnym.
8. Użytkownik kliką link, aktywujący konto.
9. System potwierdza zakończenie rejestracji i umożliwia logowanie.

S2.6. Przebiegi alternatywne

PA1. Użytkownik pomija krok 4 (brak akceptacji regulaminu).

→ System blokuje rejestrację i wyświetla komunikat o konieczności akceptacji warunków.

S2.7. Sytuacje wyjątkowe

SW1. Adres e-mail już istnieje – system informuje: „Konto z tym adresem e-mail już istnieje.”

SW2. Brak odpowiedzi z serwera – system wyświetla komunikat o błędzie i proponuje ponowienie próby.

SW3. Niepoprawny e-mail – system nie wysyła wiadomości aktywacyjnej i prosi o poprawny adres.

S2.8. Wymagania niefunkcjonalne

- Dane muszą być szyfrowane (HTTPS, bcrypt dla haseł).

- Ekran rejestracji powinien być intuicyjny i dostępny.
- Czas rejestracji nie powinien przekraczać 1 minuty.

S2.9. Uwagi i pytania otwarte

- Czy wymagana jest weryfikacja SMS?
- Czy numer telefonu jest obowiązkowy w MVP?

S3. Logowanie do systemu

S3.1. Opis

Scenariusz opisuje proces logowania użytkownika na istniejące konto za pomocą e-maila i hasła, umożliwiający dostęp do funkcji dla użytkowników zalogowanych.

S3.2. Aktorzy

Użytkownik (niezalogowany), System aplikacji mobilnej, Serwer autoryzacji.

S3.3. Warunki początkowe

- Konto użytkownika istnieje i jest aktywne.
- Użytkownik nie jest zalogowany.
- Połączenie sieciowe jest dostępne.

S3.4. Warunki końcowe

- Użytkownik został poprawnie uwierzytelniony.
- System zapisuje token sesji i przekierowuje użytkownika do ekranu głównego.

S3.5. Przebieg główny

1. Użytkownik wybiera „Zaloguj się”.
2. System wyświetla formularz logowania.
3. Użytkownik wprowadza adres e-mail i hasło.
4. System wysyła dane do serwera autoryzacji.
5. Serwer weryfikuje dane i zwraca token autoryzacyjny.
6. System zapisuje token i loguje użytkownika
7. Użytkownik zostaje przekierowany do ekranu mapy dostępnych hulajnóg.

S3.6. Przebiegi alternatywne

PA1. Użytkownik wybiera logowanie przez Google/Apple.

→ System przekierowuje do zewnętrznego dostawcy autoryzacji (OAuth2).

→ Po sukcesie przechodzi do kroku 7.

S3.7. Sytuacje wyjątkowe

SW1. Nieprawidłowe dane logowania – komunikat: „Niepoprawny e-mail lub hasło.”

SW2. Konto заблокowane – system informuje o blokadzie i podaje kontakt do wsparcia.

SW3. Brak Internetu – komunikat: „Brak połączenia z serwerem.”

S3.8. Wymagania niefunkcjonalne

- Proces logowania ≤ 2 sekundy.
- Dane przesyłane z użyciem protokołu HTTPS.
- Token sesyjny wygasza po 24 godzinach nieaktywności.

S3.9. Uwagi i pytania otwarte

- Czy przewidziano funkcję „zapamiętaj mnie”?
- Czy aplikacja ma automatycznie odnawiać token po jego wygaśnięciu?

S4. Resetowanie hasła

S4.1. Opis

Scenariusz opisuje procedurę odzyskiwania dostępu do konta w przypadku zapomnienia hasła poprzez wysłanie linku resetującego na adres e-mail użytkownika.

S4.2. Aktorzy

Użytkownik (niezalogowany), System aplikacji, System e-mail (zewnętrzny).

S4.3. Warunki początkowe

- Konto istnieje w systemie.
- Użytkownik nie jest zalogowany.
- System ma połączenie z Internetem.

S4.4. Warunki końcowe

- Hasło użytkownika zostało zresetowane.
- Użytkownik może zalogować się przy użyciu nowego hasła.

S4.5. Przebieg główny

1. Użytkownik wybiera opcję „Nie pamiętam hasła”.
2. System prosi o podanie adresu e-mail.
3. Użytkownik wprowadza adres e-mail i potwierdza.
4. System wysyła wiadomość z linkiem resetującym.
5. Użytkownik otwiera wiadomość i kliką link.
6. System wyświetla ekran zmiany hasła.
7. Użytkownik wprowadza nowe hasło i potwierdza.
8. System zapisuje nowe hasło i informuje o powodzeniu operacji.
9. Użytkownik może zalogować się nowymi danymi.

S4.6. Przebiegi alternatywne

PA1. Użytkownik nie pamięta adresu e-mail.

→ System sugeruje kontakt z pomocą techniczną.

S4.7. Sytuacje wyjątkowe

SW1. Adres e-mail nie istnieje w systemie – komunikat: „Nie znaleziono konta.”

SW2. Link wygasł – system prosi o ponowne zainicjowanie resetu.

SW3. Błąd serwera – komunikat o niepowodzeniu i próbie ponowienia.

S4.8. Wymagania niefunkcjonalne

- Link resetujący ważny przez 30 minut.
- Wiadomość e-mail powinna dotrzeć w ciągu 1 minuty.
- Formularz musi mieć walidację siły hasła.

S4.9. Uwagi i pytania otwarte

- Czy wymagane jest potwierdzenie SMS dla resetu?
- Czy można użyć linku resetującego wielokrotnie?

S5. Rozpoczynanie przejazdu

S5.1. Opis

Scenariusz opisuje rozpoczęcie przejazdu hulajnogą przez użytkownika zalogowanego. Użytkownik może zeskanować kod QR pojazdu lub wybrać go z mapy. Po pomyślnym odblokowaniu hulajnogi system rozpoczyna naliczanie opłat zgodnie z aktualną taryfą.

S5.2. Aktorzy

Użytkownik (zalogowany), Urządzenie hulajnoga (IoT), System płatności (zewnętrzny).

S5.3. Warunki początkowe

- Użytkownik jest zalogowany.
- Hulajnoga jest dostępna i widoczna na mapie.
- Aplikacja ma dostęp do Internetu, aparatu i lokalizacji GPS.
- Użytkownik posiada wystarczające środki lub aktywną metodę płatności.

S5.4. Warunki końcowe

- Hulajnoga zostaje odblokowana.
- System rozpoczyna naliczanie opłat w czasie rzeczywistym.
- Status hulajnogi zmienia się na „W trakcie jazdy”.

S5.5. Przebieg główny

1. Użytkownik otwiera aplikację i wybiera hulajnogę z mapy lub skanuje jej kod QR.
2. System weryfikuje dostępność hulajnogi.
3. System sprawdza, czy użytkownik ma środki lub ważną metodę płatności.
4. System wysyła do hulajnogi komendę „odblokuj”.
5. Hulajnoga potwierdza odblokowanie.
6. System rozpoczyna naliczanie opłat zgodnie z taryfą.
7. Użytkownik może rozpocząć jazdę.

S5.6. Przebiegi alternatywne

PA1. Użytkownik nie ma wystarczających środków w portfelu.

→ System wyświetla komunikat i proponuje doładowanie.

→ Po doładowaniu powrót do kroku 3.

PA2. Hulajnoga nie reaguje na polecenie.

→ System ponawia próbę 3 razy, po czym proponuje kontakt z pomocą techniczną.

S5.7. Sytuacje wyjątkowe

SW1. Brak połączenia z serwerem – system informuje o błędzie i blokuje proces.

SW2. Hulajnoga zgłasza błąd techniczny (np. niski poziom baterii) – system anuluje start przejazdu.

S5.8. Wymagania niefunkcjonalne

- Czas od skanowania do odblokowania \leq 5 sek.
- Komunikacja z hulajnogą szyfrowana (TLS).
- System powinien działać także przy słabym sygnale LTE.

S5.9. Uwagi i pytania otwarте

- Czy można rozpocząć przejazd przy saldzie ujemnym, jeśli użytkownik ma podpiętą kartę?
- Czy aplikacja powinna automatycznie blokować hulajnogę przy utracie sygnału GPS?

S6. Ukończenie przejazdu

S6.1. Opis

Scenariusz opisuje proces zakończenia przejazdu – użytkownik kończy jazdę, blokuje hulajnogę, a system dokonuje automatycznego rozliczenia.

S6.2. Aktorzy

Użytkownik (zalogowany), System aplikacji, Hulajnoga (IoT), System płatności (zewnętrzny).

S6.3. Warunki początkowe

- Użytkownik ma aktywny przejazd.
- Hulajnoga znajduje się w strefie dozwolonego parkowania.
- Połączenie z Internetem jest aktywne.

S6.4. Warunki końcowe

- Przejazd zostaje zakończony.
- Hulajnoga jest zablokowana.
- Opłata została naliczona i potwierdzona.
- Dane przejazdu są zapisane w historii użytkownika.

S6.5. Przebieg główny

1. Użytkownik zatrzymuje hulajnogę i wybiera „Zakończ przejazd”.
2. System pobiera aktualne współrzędne GPS.
3. System weryfikuje, czy hulajnoga znajduje się w strefie dozwolonej.
4. System wysyła do hulajnogi komendę „zablokuj”.
5. Hulajnoga potwierdza blokadę.
6. System oblicza koszt przejazdu i przesyła żądanie do systemu płatności.
7. System otrzymuje potwierdzenie płatności.
8. Użytkownik widzi podsumowanie przejazdu (czas, dystans, koszt).
9. System zapisuje dane w historii przejazdów.

S6.6. Przebiegi alternatywne

PA1. Użytkownik próbuje zakończyć przejazd poza strefą.

→ System proponuje zaparkowanie w dozwolonej strefie lub nalicza opłatę dodatkową.

S6.7. Sytuacje wyjątkowe

SW1. Brak łączności z hulajnogą – system proponuje zakończenie awaryjne (np. zgłoszenie do obsługi).

SW2. Płatność nieudana – system wysyła powiadomienie o problemie i oznacza przejazd jako „oczekujący na płatność”.

S6.8. Wymagania niefunkcjonalne

- Czas zakończenia przejazdu \leq 10 sekund.
- Raport przejazdu generowany natychmiast po płatności.
- Dane zapisane w systemie przez minimum 2 lata.

S6.9. Uwagi i pytania otwarte

- Czy dopuszczalne jest zakończenie przejazdu offline (z synchronizacją późniejszą)?
- Czy system ma wysyłać powiadomienie push o kwocie opłaty?

S7. Zarządzanie metodami płatności

S7.1. Opis

Scenariusz umożliwia użytkownikowi dodawanie, edytowanie i usuwanie metod płatności w aplikacji. Dane karty są tokenizowane i przekazywane do zewnętrznego operatora płatności.

S7.2. Aktorzy

Użytkownik (zalogowany), System aplikacji, System płatności (zewnętrzny).

S7.3. Warunki początkowe

- Użytkownik jest zalogowany.
- System ma połączenie z siecią.
- Użytkownik posiada aktywne konto płatnicze.

S7.4. Warunki końcowe

- Karta płatnicza została dodana/usunięta/zaktualizowana.
- System przechowuje tylko token karty (bez danych wrażliwych).

S7.5. Przebieg główny

1. Użytkownik otwiera sekcję „Płatności”.
2. System wyświetla listę aktualnych metod płatności.
3. Użytkownik wybiera „Dodaj kartę”.
4. Użytkownik wprowadza dane karty (numer, data ważności, CVC).
5. System przekazuje dane do operatora płatności.
6. Operator zwraca token autoryzacyjny.
7. System zapisuje token i potwierdza dodanie karty.
8. Użytkownik może ustawić kartę jako domyślną lub usunąć istniejącą.

S7.6. Przebiegi alternatywne

- PA1. Użytkownik wybiera metodę Apple Pay / Google Pay.
→ System przekierowuje do autoryzacji przez portfel cyfrowy.
→ Po sukcesie zapisuje token płatności.

S7.7. Sytuacje wyjątkowe

- SW1. Błąd autoryzacji karty – komunikat: „Nie udało się dodać karty. Spróbuj ponownie.”
SW2. Przerwanie połączenia z operatorem płatności – operacja anulowana.

S7.8. Wymagania niefunkcjonalne

- Zgodność z PCI DSS.
- Czas dodawania karty ≤ 5 sekund.
- Dane nie mogą być lokalnie buforowane.

S7.9. Uwagi i pytania otwarte

- Czy użytkownik może dodać więcej niż 3 metody płatności?
- Czy planowana jest integracja z BLIK?

S8. Przeglądanie historii przejazdów

S8.1. Opis

Scenariusz pozwala użytkownikowi na przeglądanie wszystkich zrealizowanych przejazdów, ich szczegółów i podsumowań finansowych.

S8.2. Aktorzy

Użytkownik (zalogowany), System aplikacji.

S8.3. Warunki początkowe

- Użytkownik jest zalogowany.
- W systemie istnieją zapisane dane o przejazdach.
- Połączenie z Internetem jest aktywne.

S8.4. Warunki końcowe

- Lista przejazdów została wyświetlona.
- Użytkownik może zapoznać się ze szczegółami każdego przejazdu.

S8.5. Przebieg główny

1. Użytkownik otwiera sekcję „Historia przejazdów”.
2. System pobiera dane o przejazdach z serwera.
3. System wyświetla listę z datami, trasą i kosztami.
4. Użytkownik wybiera konkretny przejazd.
5. System prezentuje szczegóły (mapa, czas, dystans, koszt, metoda płatności).
6. Użytkownik może wyeksportować lub udostępnić dane (PDF, e-mail).

S8.6. Przebiegi alternatywne

PA1. Brak przejazdów – system wyświetla komunikat „Nie zrealizowano jeszcze żadnego przejazdu.”

S8.7. Sytuacje wyjątkowe

SW1. Błąd połączenia z serwerem – system wyświetla dane lokalne (ostatnia kopia).

SW2. Błąd renderowania mapy – system pokazuje trasę uproszczoną bez wizualizacji.

S8.8. Wymagania niefunkcjonalne

- Historia przejazdów ładuje się w <3 sek.
- Dane prezentowane w kolejności chronologicznej.
- Możliwość filtrowania po dacie lub kwocie.

S8.9. Uwagi i pytania otwarte

- Czy historia ma limit czasowy (np. ostatnie 12 miesięcy)?

- Czy użytkownik może usunąć zapis przejazdu?

S9. Doładowanie portfela

S9.1. Opis

Scenariusz umożliwia użytkownikowi zasilenie swojego portfela w aplikacji określoną kwotą za pomocą zdefiniowanej metody płatności.

S9.2. Aktorzy

Użytkownik (zalogowany), System aplikacji, System płatności (zewnętrzny).

S9.3. Warunki początkowe

- Użytkownik posiada aktywne konto i przynajmniej jedną metodę płatności.
- System ma połączenie z Internetem.

S9.4. Warunki końcowe

- Portfel użytkownika został zasilony wybraną kwotą.
- Historia transakcji została zaktualizowana.

S9.5. Przebieg główny

1. Użytkownik wybiera „Doładuj portfel”.
2. System proponuje kwoty predefiniowane (np. 10 zł, 25 zł, 50 zł).
3. Użytkownik wybiera kwotę i metodę płatności.
4. System przekazuje żądanu do systemu płatności.
5. System płatności realizuje transakcję i zwraca potwierdzenie.
6. System zwiększa saldo portfela o wskazaną kwotę.
7. Użytkownik otrzymuje powiadomienie o sukcesie.

S9.6. Przebiegi alternatywne

PA1. Użytkownik wprowadza własną kwotę (spoza predefiniowanych).

→ System waliduje wartość (np. min. 5 zł, max. 200 zł) i kontynuuje.

S9.7. Sytuacje wyjątkowe

SW1. Transakcja odrzucona – komunikat: „Płatność nieudana.”

SW2. Brak połączenia – system zapisuje próbę i synchronizuje po odzyskaniu Internetu.

S9.8. Wymagania niefunkcjonalne

- Doładowanie ≤ 5 sek.
- Wszystkie kwoty w PLN, dokładność 2 miejsca po przecinku.
- Powiadomienia push o każdym doładowaniu.

S9.9. Uwagi i pytania otwarte

- Czy przewidziane są promocje przy doładowaniu (np. +10%)?
- Czy portfel może mieć saldo ujemne?

S10. Zgłaszcenie usterki hulajnogi

S10.1. Opis

Scenariusz opisuje proces zgłoszania usterki lub problemu technicznego z hulajnogą przez użytkownika. Zgłoszenie trafia do systemu serwisowego.

S10.2. Aktorzy

Użytkownik (zalogowany), System aplikacji, Administrator, Serwisant.

S10.3. Warunki początkowe

- Użytkownik jest zalogowany.
- Hulajnoga jest zidentyfikowana (np. po numerze QR lub ID przejazdu).
- System ma połączenie z Internetem.

S10.4. Warunki końcowe

- Zgłoszenie usterki zostało zapisane w systemie.
- Administrator/serwisant widzi nowe zadanie w panelu serwisowym.

S10.5. Przebieg główny

1. Użytkownik wybiera „Zgłoś usterkę” z poziomu aplikacji.
2. System wyświetla formularz zgłoszenia.
3. Użytkownik wybiera typ usterki (np. hamulce, bateria, opony, inne).
4. Użytkownik może dodać opis i zdjęcie.
5. System zapisuje zgłoszenie w bazie i przypisuje mu numer referencyjny.
6. System powiadamia administratora o nowym zgłoszeniu.
7. Użytkownik otrzymuje potwierdzenie utworzenia zgłoszenia.

S10.6. Przebiegi alternatywne

PA1. Użytkownik anuluje zgłoszenie przed wysłaniem – system nie zapisuje danych.

S10.7. Sytuacje wyjątkowe

SW1. Brak połączenia – zgłoszenie zapisane lokalnie i wysłane po odzyskaniu sieci.

SW2. Niepowodzenie zapisu – komunikat: „Nie udało się przesłać zgłoszenia.”

S10.8. Wymagania niefunkcjonalne

- Maksymalny rozmiar zdjęcia: 5 MB.
- Formularz działa offline z synchronizacją po odzyskaniu połączenia.
- Czas zapisu w sek.

S10.9. Uwagi i pytania otwarte

- Czy użytkownik otrzymuje informację o postępie naprawy?
- Czy można edytować zgłoszenie po wysłaniu?

S11. Zarządzanie profilem

S11.1. Opis

Scenariusz opisuje możliwość edycji danych osobowych i ustawień konta przez użytkownika zalogowanego. Obejmuje zmianę imienia, nazwiska, adresu e-mail, hasła oraz preferencji powiadomień.

S11.2. Aktorzy

Użytkownik (zalogowany), System aplikacji.

S11.3. Warunki początkowe

- Użytkownik jest zalogowany.
- Połączenie z Internetem jest aktywne.
- Dane użytkownika są dostępne w systemie.

S11.4. Warunki końcowe

- Dane profilu zostały zaktualizowane i zapisane w systemie.
- Użytkownik otrzymał potwierdzenie o pomyślnej aktualizacji.

S11.5. Przebieg główny

1. Użytkownik wybiera w aplikacji zakładkę „Profil”.
2. System wyświetla aktualne dane osobowe i ustawienia.
3. Użytkownik dokonuje zmian w polach danych.
4. Użytkownik potwierdza zapis zmian.
5. System waliduje wprowadzone dane.
6. System zapisuje zaktualizowany profil w bazie danych.
7. System wyświetla komunikat o pomyślnej aktualizacji.

S11.6. Przebiegi alternatywne

PA1. Użytkownik zmienia tylko preferencje powiadomień.

→ System zapisuje nowe ustawienia lokalnie, bez ponownej autoryzacji.

PA2. Użytkownik próbuje zmienić e-mail na już istniejący w systemie.

→ System odrzuca zmianę i informuje o konieczności podania innego adresu.

S11.7. Sytuacje wyjątkowe

SW1. Błąd zapisu danych – system wyświetla komunikat „Nie udało się zapisać zmian.”

SW2. Brak połączenia z serwerem – zmiany są zapisywane lokalnie i synchronizowane po odzyskaniu Internetu.

S11.8. Wymagania niefunkcjonalne

- Dane muszą być szyfrowane (HTTPS, SSL).

- Zmiana hasła wymaga ponownej autoryzacji użytkownika.
- Operacja zapisu \leq 3 sekundy.

S11.9. Uwagi i pytania otwarte

- Czy użytkownik może usunąć swoje konto z poziomu aplikacji?
- Czy edycja danych wymaga potwierdzenia e-mail?

S12. Lokalizowanie hulajnogi dźwiękiem

S12.1. Opis

Scenariusz opisuje możliwość uruchomienia sygnału dźwiękowego w wybranej hulajnodze, aby pomóc użytkownikowi ją odnaleźć w terenie (np. w gęsto zaparkowanym obszarze).

S12.2. Aktorzy

Użytkownik (zalogowany), Hulajnoga (IoT), System aplikacji.

S12.3. Warunki początkowe

- Użytkownik jest zalogowany.
- Hulajnoga znajduje się w promieniu 50 m od użytkownika.
- Połączenie z siecią jest aktywne.

S12.4. Warunki końcowe

- Hulajnoga emituje dźwięk przez określony czas (np. 15 sek.).
- Użytkownik może ją zlokalizować fizycznie.

S12.5. Przebieg główny

1. Użytkownik wybiera hulajnogę z mapy.
2. System weryfikuje, czy hulajnoga jest w zasięgu.
3. Użytkownik wybiera opcję „Znajdź hulajnogę”.
4. System wysyła komendę „włącz dźwięk” do hulajnogi.
5. Hulajnoga potwierdza odbiór komendy.
6. Hulajnoga uruchamia sygnał dźwiękowy.
7. Po określonym czasie dźwięk automatycznie się wyłącza.

S12.6. Przebiegi alternatywne

PA1. Hulajnoga jest poza zasięgiem sieci.

→ System informuje użytkownika: „Nie można połączyć z hulajnogą. Spróbuj ponownie później.”

S12.7. Sytuacje wyjątkowe

SW1. Awaria głośnika hulajnogi – system zgłasza problem techniczny.

SW2. Użytkownik anuluje lokalizację – komenda „wyłącz dźwięk” wysyłana do hulajnogi.

S12.8. Wymagania niefunkcjonalne

- Czas reakcji hulajnogi \leq 2 sekundy.
- Dźwięk powinien być słyszalny w promieniu min. 20 m.
- Funkcja dostępna tylko dla aktywnych hulajnóg.

S12.9. Uwagi i pytania otwarte

- Czy funkcja ma być ograniczona czasowo (np. tylko po 5 minutach od zakończenia jazdy)?

- Czy sygnał ma różnić się w zależności od typu hulajnogi

S13. Zarządzanie kontami użytkowników

S13.1. Opis

Scenariusz opisuje działania administratora w zakresie przeglądania, edytowania, blokowania i usuwania kont klientów oraz serwisantów.

S13.2. Aktorzy

Administrator, System aplikacji.

S13.3. Warunki początkowe

- Administrator jest zalogowany.
- System ma połączenie z bazą danych użytkowników.

S13.4. Warunki końcowe

- Konto zostało zaktualizowane, заблокowane lub usunięte.
- Zmiany są odzwierciedlone w bazie i logach systemowych.

S13.5. Przebieg główny

1. Administrator otwiera panel administracyjny.
2. System wyświetla listę wszystkich kont.
3. Administrator wybiera użytkownika z listy.
4. Administrator wykonuje akcję: „edytuj”, „zablokuj” lub „usuń”.
5. System weryfikuje poprawność operacji (np. uprawnienia).
6. System zapisuje zmiany w bazie danych.
7. Administrator otrzymuje potwierdzenie o sukcesie.

S13.6. Przebiegi alternatywne

PA1. Administrator przywraca wcześniej zablokowane konto.
→ System zmienia status użytkownika na „Aktywny”.

S13.7. Sytuacje wyjątkowe

SW1. Błąd bazy danych – zmiana nie zostaje zapisana.

S13.8. Wymagania niefunkcjonalne

- Dziennik audytowy zapisuje wszystkie operacje administratora.
- Interfejs powinien pozwalać na filtrowanie kont.
- Operacje edycji ≤ 2 sekundy.

S13.9. Uwagi i pytania otwarte

- Czy usunięcie konta jest nieodwracalne?
- Czy administrator może tymczasowo zawiesić konto serwisanta?

S14. Zarządzanie flotą hulajnóg

S14.1. Opis

Scenariusz umożliwia administratorowi dodawanie, edytowanie i dezaktywowanie hulajnóg w systemie. Obejmuje też monitorowanie ich statusu.

S14.2. Aktorzy

Administrator, Hulajnoga (IoT), System aplikacji.

S14.3. Warunki początkowe

- Administrator jest zalogowany.
- System posiada dostęp do listy hulajnóg.
- Połączenie z urządzeniami IoT jest aktywne.

S14.4. Warunki końcowe

- Dane o hulajnogach zostały zaktualizowane
- Nowe hulajnogi są gotowe do udostępnienia użytkownikom.

S14.5. Przebieg główny

1. Administrator otwiera panel „Flota”.
2. System wyświetla listę wszystkich hulajnóg.
3. Administrator dodaje nową hulajnogę, podając numer seryjny i lokalizację.
4. System wysyła do hulajnogi żądanie testowe.
5. Hulajnoga potwierdza połączenie.
6. System zapisuje hulajnogę jako „Aktywna”.
7. Administrator może edytować dane lub oznaczyć hulajnogę jako „W serwisie”.

S14.6. Przebiegi alternatywne

PA1. Hulajnoga nie odpowiada na test – system oznacza ją jako „Nieaktywna”.

PA2. Administrator importuje dane wielu hulajnóg z pliku CSV.

S14.7. Sytuacje wyjątkowe

SW1. Awaria komunikacji z IoT – system zapisuje status „nieznany”.

SW2. Błąd walidacji numeru seryjnego – komunikat „Nieprawidłowy format numeru.”

S14.8. Wymagania niefunkcjonalne

- Dodanie hulajnogi ≤ 10 sek.
- Panel floty odświeżany w czasie rzeczywistym.
- Dostęp zabezpieczony tokenem administratora.

S14.9. Uwagi i pytania otwarte

- Czy administrator może zdalnie aktualizować oprogramowanie hulajnogi?
- Czy system ma automatycznie wykrywać nieaktywne pojazdy?

S15. Zarządzanie taryfami i cennikiem

S15.1. Opis

Scenariusz umożliwia administratorowi konfigurację cen, opłat startowych i stawek minutowych dla użytkowników aplikacji.

S15.2. Aktorzy

Administrator, System aplikacji.

S15.3. Warunki początkowe

- Administrator jest zalogowany.
- System ma uprawnienia do edycji konfiguracji taryf.

S15.4. Warunki końcowe

- Nowe taryfy zostały zapisane i opublikowane.
- Użytkownicy widzą zaktualizowany cennik.

S15.5. Przebieg główny

1. Administrator otwiera sekcję „Cennik i taryfy”.
2. System wyświetla aktualne wartości opłat.
3. Administrator wprowadza nowe stawki (np. startowa, za minutę, pakiety).
4. **System waliduje dane.**
5. Administrator zatwierdza zmiany.
6. System zapisuje i publikuje nowy cennik.
7. System wysyła powiadomienie do wszystkich użytkowników.

S15.6. Przebiegi alternatywne

PA1. Administrator zapisuje zmiany jako „wersję roboczą” – system nie publikuje jeszcze taryfy.

PA2. Administrator przywraca poprzednią wersję taryfy.

S15.7. Sytuacje wyjątkowe

SW1. Błąd zapisu konfiguracji – komunikat: „Nie udało się zapisać zmian.”

SW2. Brak uprawnień – system blokuje dostęp.

S15.8. Wymagania niefunkcjonalne

- Dane taryf wersjonowane automatycznie.
- Publikacja ≤ 5 sek.
- Użytkownicy widzą nowe ceny natychmiast po publikacji.

S15.9. Uwagi i pytania otwarte

- Czy można ustawiać różne taryfy dla stref geograficznych?
- Czy system powinien archiwizować historię zmian cen?

S16. Zarządzanie strefami

S16.1. Opis

Scenariusz opisuje możliwość definiowania i modyfikowania stref operacyjnych przez administratora. Strefy te mogą obejmować obszary parkowania, strefy zakazu parkowania, obszary o ograniczonej prędkości lub obszary serwisowe.

S16.2. Aktorzy

Administrator, System aplikacji, Usługa map/GPS (zewnętrzna).

S16.3. Warunki początkowe

- Administrator jest zalogowany.
- System ma aktywne połączenie z usługą mapową.
- Baza danych zawiera aktualną listę stref.

S16.4. Warunki końcowe

- Nowa lub zmodyfikowana strefa została zapisana i opublikowana.
- Zmiany są widoczne w aplikacji użytkowników i serwisantów.

S16.5. Przebieg główny

1. Administrator otwiera panel „Zarządzanie strefami”.
2. System pobiera dane mapowe z usługi GPS i wyświetla mapę miasta.
3. Administrator wybiera opcję „Dodaj nową strefę”.
4. Administrator rysuje obszar na mapie (wielokąt lub okrąg).
5. Administrator nadaje nazwę i typ strefy (np. *Parkowanie dozwolone, Zakaz parkowania*).
6. System waliduje poprawność danych.
7. Administrator zapisuje zmiany.
8. System publikuje nową strefę w systemie mapowym i bazie danych.
9. Użytkownicy aplikacji widzą zmienione granice stref po synchronizacji mapy.

S16.6. Przebiegi alternatywne

PA1. Administrator modyfikuje istniejącą strefę → system aktualizuje dane i oznacza starszą wersję jako nieaktywną.

PA2. Administrator usuwa nieaktualną strefę → system przenosi ją do archiwum.

S16.7. Sytuacje wyjątkowe

SW1. Brak połączenia z usługą mapową – komunikat „Nie można wczytać mapy. Spróbuj ponownie później.”

SW2. Błąd zapisu – system wycofuje zmiany i informuje administratora.

S16.8. Wymagania niefunkcjonalne

- Interfejs mapy musi umożliwiać precyzyjne rysowanie wielokątów.
- Czas aktualizacji mapy \leq 10 sekund.
- System powinien zapisywać historię wszystkich zmian granic stref.

S16.9. Uwagi i pytania otwarte

- Czy różne strefy mogą się nakładać?
- Czy użytkownik ma być ostrzegany przed wjazdem do strefy zakazu?

S17. Generowanie raportów i statystyk

S17.1. Opis

Scenariusz opisuje proces tworzenia przez administratora raportów zbiorczych dotyczących aktywności użytkowników, liczby przejazdów, przychodów, popularności tras i wykorzystania floty.

S17.2. Aktorzy

Administrator, System aplikacji.

S17.3. Warunki początkowe

- Administrator jest zalogowany.
- System ma dostęp do bazy danych operacyjnych.

S17.4. Warunki końcowe

- Raport został wygenerowany w wybranym formacie (CSV, PDF, XLSX).
- Administrator może go przeglądać lub pobrać.

S17.5. Przebieg główny

1. Administrator otwiera moduł „Raporty i statystyki”.
2. System wyświetla listę dostępnych raportów (np. *Przychody dzienne, Najczęściej wybierane trasy*).
3. Administrator wybiera typ raportu i zakres dat.
4. System przetwarza dane z bazy.
5. System prezentuje podgląd raportu.
6. Administrator może wyeksportować raport w wybranym formacie.
7. System zapisuje historię wygenerowania raportu.

S17.6. Przebiegi alternatywne

PA1. Administrator planuje raport cykliczny → system automatycznie generuje go co tydzień/miesiąc.

PA2. Administrator łączy dane z kilku raportów w zestawienie zbiorcze.

S17.7. Sytuacje wyjątkowe

SW1. Brak danych dla wybranego okresu – komunikat „Brak danych do wyświetlenia.”

SW2. Przekroczenie limitu danych – system dzieli raport na kilka plików.

S17.8. Wymagania niefunkcjonalne

- Czas generowania raportu \leq 15 sek.
- Możliwość eksportu do formatów: PDF, CSV, XLSX.
- Dostępność raportów tylko dla kont z rolą administratora.

S17.9. Uwagi i pytania otwarte

- Czy raporty mają być możliwe do wysłania automatycznie e-mailem?

- Czy system ma wizualizować dane (np. wykresy w panelu)?

S18. Zarządzanie zgłoszeniami serwisowymi

S18.1. Opis

Scenariusz przedstawia proces obsługi zgłoszeń o usterkach hulajnóg, przesyłanych przez użytkowników. Administrator może przeglądać zgłoszenia, filtrować je, przypisywać do serwisantów i monitorować ich realizację.

S18.2. Aktorzy

Administrator, Serwisant, System aplikacji.

S18.3. Warunki początkowe

- Administrator jest zalogowany.
- W systemie istnieją aktywne zgłoszenia.
- Co najmniej jeden serwisant jest przypisany do systemu.

S18.4. Warunki końcowe

- Zgłoszenie zostało przypisane do serwisanta lub zamknięte.
- Status zgłoszenia został zaktualizowany w systemie.

S18.5. Przebieg główny

1. Administrator otwiera panel „Zgłoszenia serwisowe”.
2. System wyświetla listę aktywnych zgłoszeń z filtrami (np. *Data, Status, Priorytet*).
3. Administrator wybiera konkretne zgłoszenie.
4. System wyświetla szczegóły usterki, lokalizację i dane hulajnogi.
5. Administrator przypisuje zgłoszenie do wybranego serwisanta.
6. System aktualizuje status na „W trakcie realizacji”.
7. Serwisant otrzymuje powiadomienie o nowym zadaniu.
8. Administrator może monitorować postęp i zamknąć zgłoszenie po zakończeniu naprawy.

S18.6. Przebiegi alternatywne

PA1. Administrator zmienia priorytet zgłoszenia → system aktualizuje sortowanie i kolorystykę listy.

PA2. Administrator ponownie przydziela zgłoszenie innemu serwisantowi.

S18.7. Sytuacje wyjątkowe

SW1. Brak dostępnych serwisantów – system ustawia zgłoszenie w kolejce oczekujących.
SW2. Serwisant odrzuca przydzielone zgłoszenie – administrator otrzymuje powiadomienie i może przypisać je ponownie.

S18.8. Wymagania niefunkcjonalne

- Czas reakcji interfejsu ≤ 2 sekundy.

- Dane o zgłoszeniach aktualizowane w czasie rzeczywistym (WebSocket).
- System rejestruje historię wszystkich operacji administracyjnych.

S18.9. Uwagi i pytania otwarte

- Czy administrator może nadawać priorytety automatycznie (np. po typie usterki)?
- Czy zgłoszenia można eksportować do raportów miesięcznych?

S19. Przeglądanie listy zadań serwisowych

S19.1. Opis

Scenariusz opisuje możliwość przeglądania przez serwisanta wszystkich przypisanych mu zadań związanych z utrzymaniem floty hulajnóg. Serwisant ma wgląd w szczegóły każdego zgłoszenia, takie jak typ usterki, lokalizacja hulajnogi, priorytet zadania oraz historia wcześniejszych interwencji.

S19.2. Aktorzy

Serwisant, System aplikacji, Administrator (jako aktor pośredni).

S19.3. Warunki początkowe

- Serwisant jest zalogowany w systemie.
- System posiada przypisane zadania serwisowe dla tego konta.
- Aplikacja mobilna lub panel webowy serwisanta jest połączony z internetem.

S19.4. Warunki końcowe

- Serwisant uzyskał dostęp do pełnej listy swoich zadań.
- Może otworzyć szczegóły konkretnego zgłoszenia i rozpocząć jego realizację.

S19.5. Przebieg główny

1. Serwisant loguje się do panelu serwisowego.
2. System uwierzytelnia dane i otwiera ekran główny.
3. Serwisant wybiera zakładkę „Moje zadania”.
4. System pobiera z bazy danych wszystkie aktywne zgłoszenia przypisane do danego serwisanta.
5. System wyświetla listę zadań wraz z informacjami: numer zgłoszenia, typ usterki, lokalizacja, priorytet i status.
6. Serwisant może filtrować zadania według statusu (np. *oczekujące*, *w trakcie*, *zakończone*).
7. Serwisant wybiera jedno zgłoszenie, aby przejść do jego szczegółów.
8. System otwiera ekran szczegółów, zawierający dane hulajnogi, opis problemu, zdjęcia od użytkownika oraz przycisk „Rozpocznij naprawę”.

S19.6. Przebiegi alternatywne

PA1. Serwisant przegląda ukończone zadania → system pobiera archiwalne dane i umożliwia ich sortowanie według daty.

PA2. Serwisant wyszukuje konkretne zgłoszenie po numerze seryjnym hulajnogi → system filtry wyniki w czasie rzeczywistym.

S19.7. Sytuacje wyjątkowe

SW1. Brak przypisanych zadań – system wyświetla komunikat: „Brak zadań do wykonania.”

SW2. Utrata połączenia z serwerem – system wyświetla informację o błędzie i oferuje tryb offline (dane z ostatniej synchronizacji).

SW3. Błąd pobierania danych – system loguje zdarzenie i informuje administratora.

S19.8. Wymagania niefunkcjonalne

- Czas wczytania listy zadań ≤ 3 sekundy.
- Interfejs mobilny powinien umożliwiać przeglądanie danych w terenie (duże przyciski, tryb offline)
- Synchronizacja danych automatyczna co 5 minut lub przy zmianie statusu zadania.

S19.9. Uwagi i pytania otwarte

- Czy zadania powinny być sortowane domyślnie po priorytecie czy dacie utworzenia?
- Czy serwisant może otrzymywać powiadomienia push o nowych zadaniach?

S20. Aktualizacja statusu hulajnogi

S20.1. Opis

Scenariusz opisuje proces zmiany statusu hulajnogi przez serwisanta po wykonaniu czynności serwisowej. Może to być zmiana z „W naprawie” na „Dostępna”, „Wyłączona z użytku”, „Oczekuje na części” itp. Zmiana statusu wpływa na widoczność hulajnogi w systemie użytkowników oraz administratora.

S20.2. Aktorzy

Serwisant, System aplikacji, Administrator (jako odbiorca aktualizacji), Urządzenie hulajnoga (IoT).

S20.3. Warunki początkowe

- Serwisant jest zalogowany.
- Hulajnoga jest zarejestrowana w systemie.
- Zadanie serwisowe dotyczące tej hulajnogi jest aktywne lub zakończone.

S20.4. Warunki końcowe

- Status hulajnogi został zaktualizowany w bazie danych.
- Administrator i system użytkowników widzą nowy stan w czasie rzeczywistym.

S20.5. Przebieg główny

1. Serwisant otwiera szczegóły zadania związanego z hulajnogą.
2. System wyświetla aktualny status (np. *W naprawie*).
3. Serwisant wybiera przycisk „Zmień status”.
4. System wyświetla listę dostępnych stanów (np. *Dostępna*, *W naprawie*, *Wyłączona z użytku*).
5. Serwisant wybiera odpowiedni status.
6. System prosi o potwierdzenie zmiany.
7. Serwisant zatwierdza operację.
8. System aktualizuje status hulajnogi w bazie danych.
9. System wysyła powiadomienie do administratora o zmianie.
10. Hulajnoga otrzymuje ewentualne zdalne polecenie (np. odblokowanie GPS po zakończeniu naprawy).

S20.6. Przebiegi alternatywne

PA1. Serwisant dodaje komentarz przy zmianie statusu → system zapisuje notatkę w historii serwisowej hulajnogi.

PA2. Serwisant zmienia status kilku hulajnóg jednocześnie → system przetwarza zbiorcze polecenie i raportuje wynik.

S20.7. Sytuacje wyjątkowe

- SW1. Brak połączenia z serwerem – zmiana statusu zostaje zapisana lokalnie i zsynchronizowana po odzyskaniu połączenia.
- SW2. Brak autoryzacji – system odrzuca zmianę i wymaga ponownego logowania.
- SW3. Błąd komunikacji z hulajnogą IoT – status w bazie uaktualniony, ale system oznacza urządzenie jako „niezsynchronizowane”.

S20.8. Wymagania niefunkcjonalne

- Czas aktualizacji statusu \leq 2 sekundy.
- Operacja powinna być odporna na utratę połączenia (tryb offline).
- Wszystkie zmiany statusów muszą być rejestrowane w historii logów.

S20.9. Uwagi i pytania otwarte

- Czy system powinien automatycznie ustawiać hulajnogę jako „Dostępna” po pozytywnym zakończeniu testów?
- Czy serwisant może cofnąć błędnie wprowadzoną zmianę statusu bez udziału administratora?

4. Zapoznanie się z narzędziem CASE. Diagramy pakietów i przypadków użycia.

Diagram pakietów:

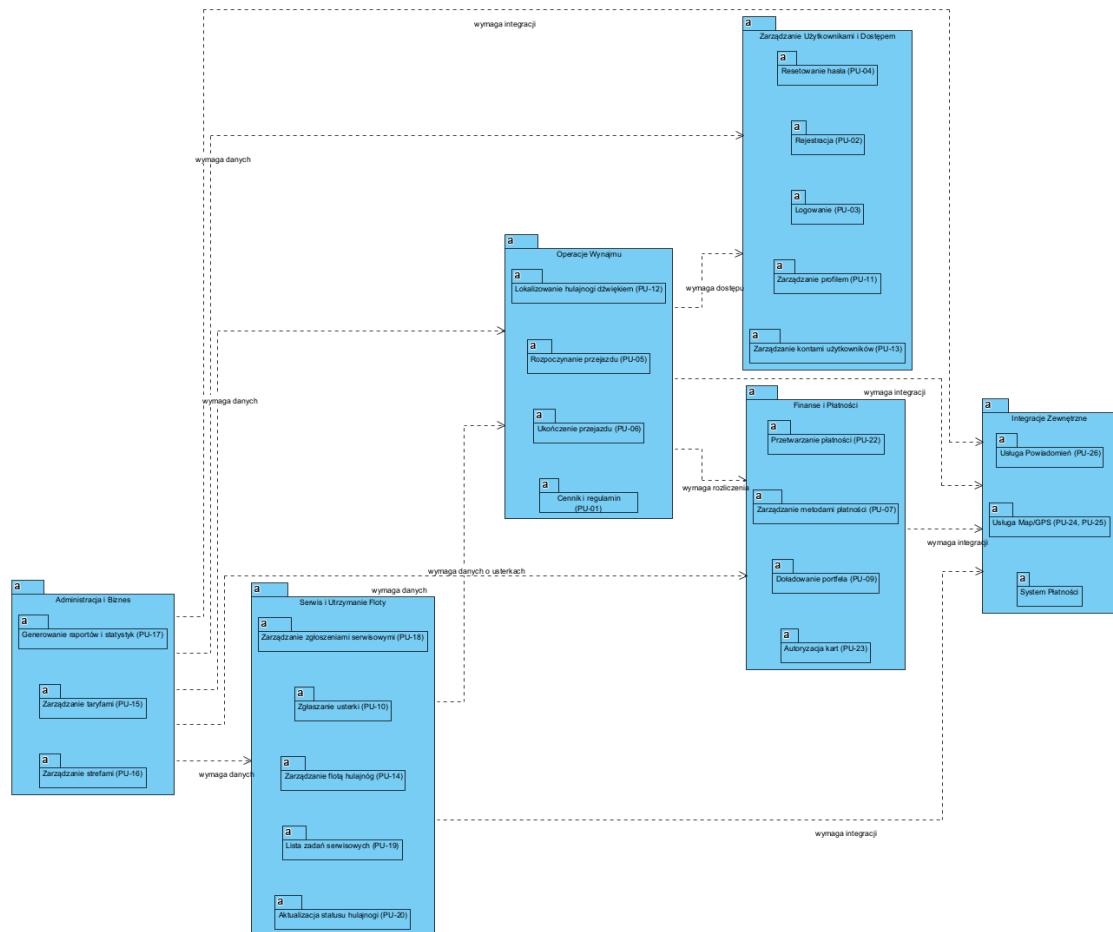
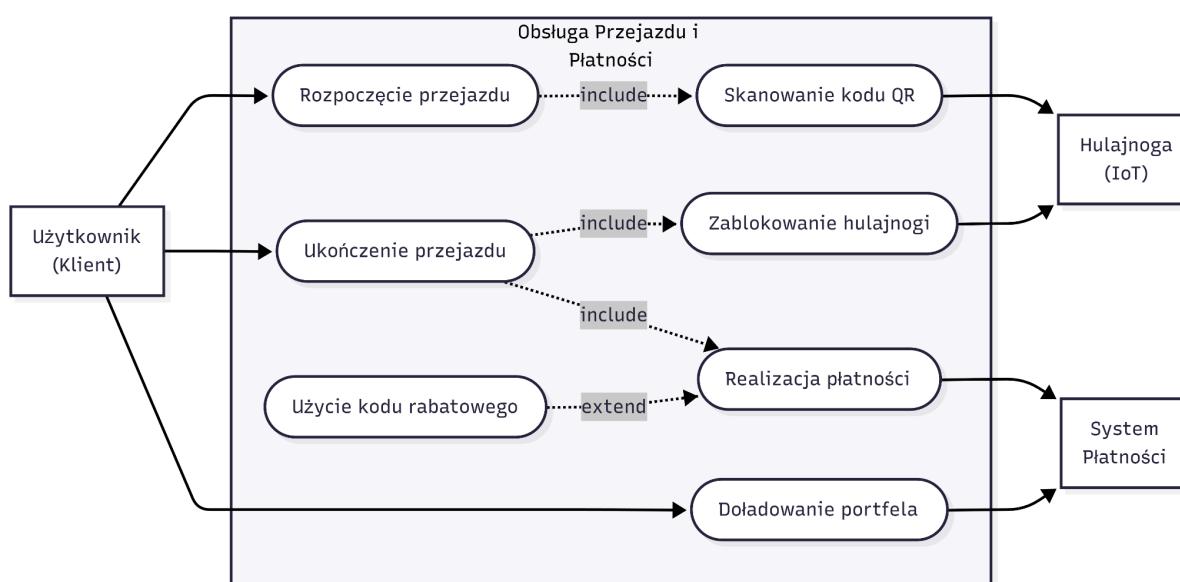


Diagram 1: Perspektywa Użytkownika



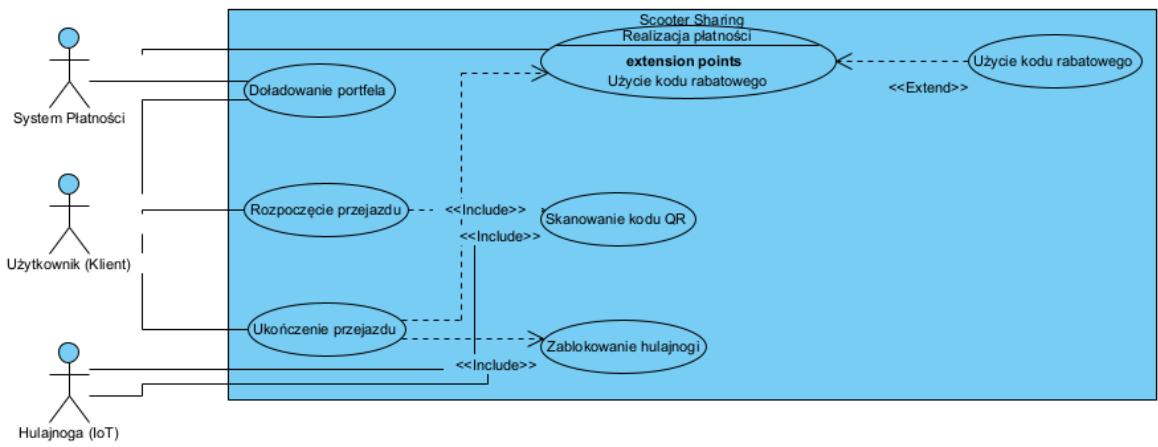
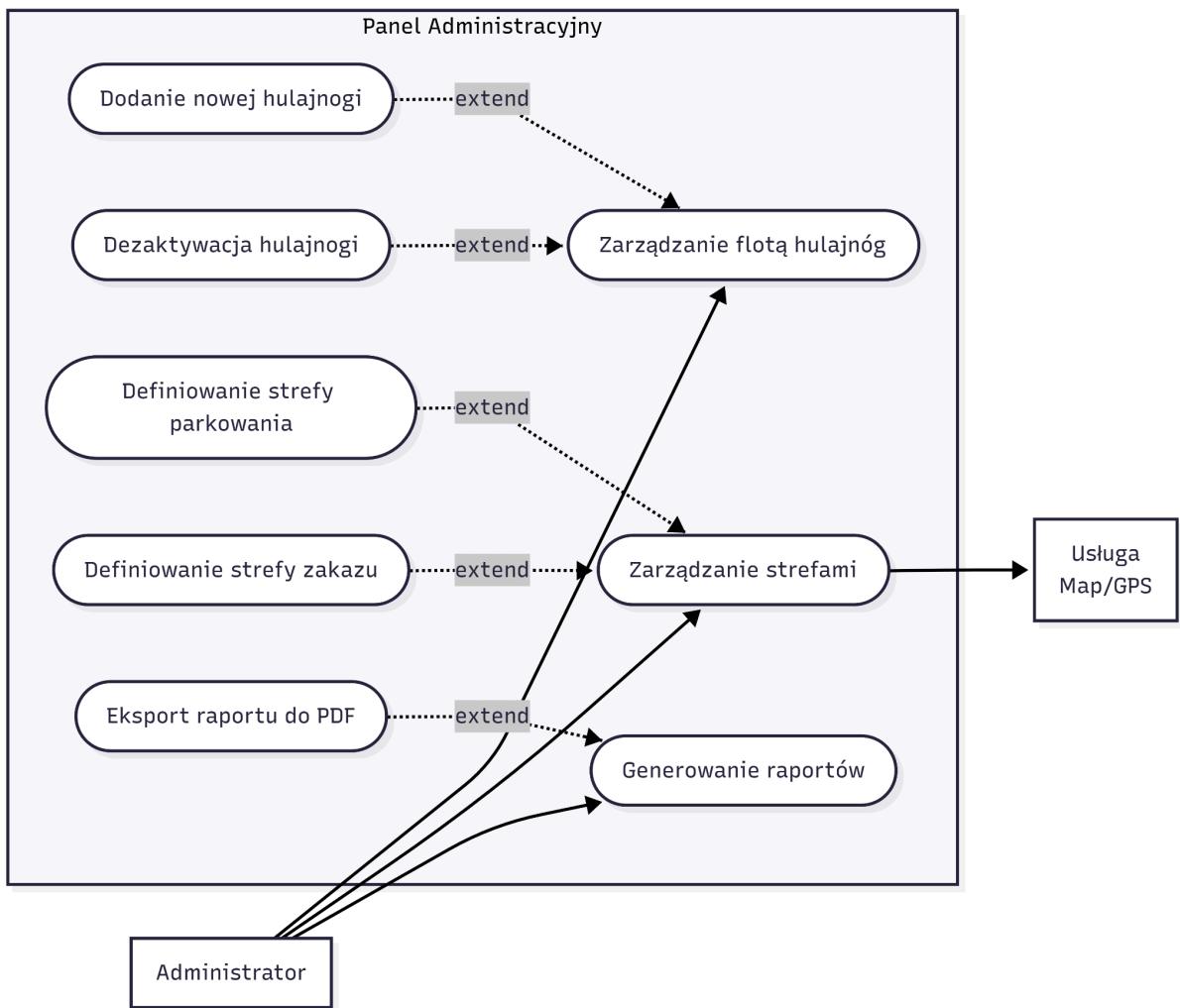


Diagram 2: Perspektywa Administratora



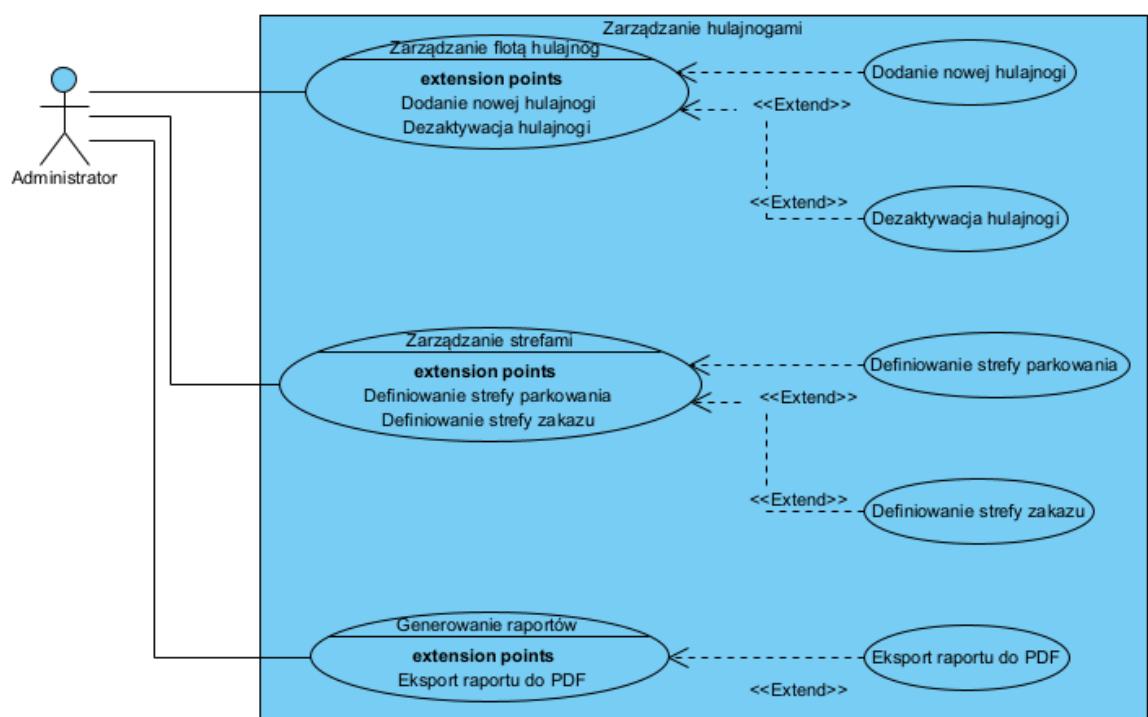
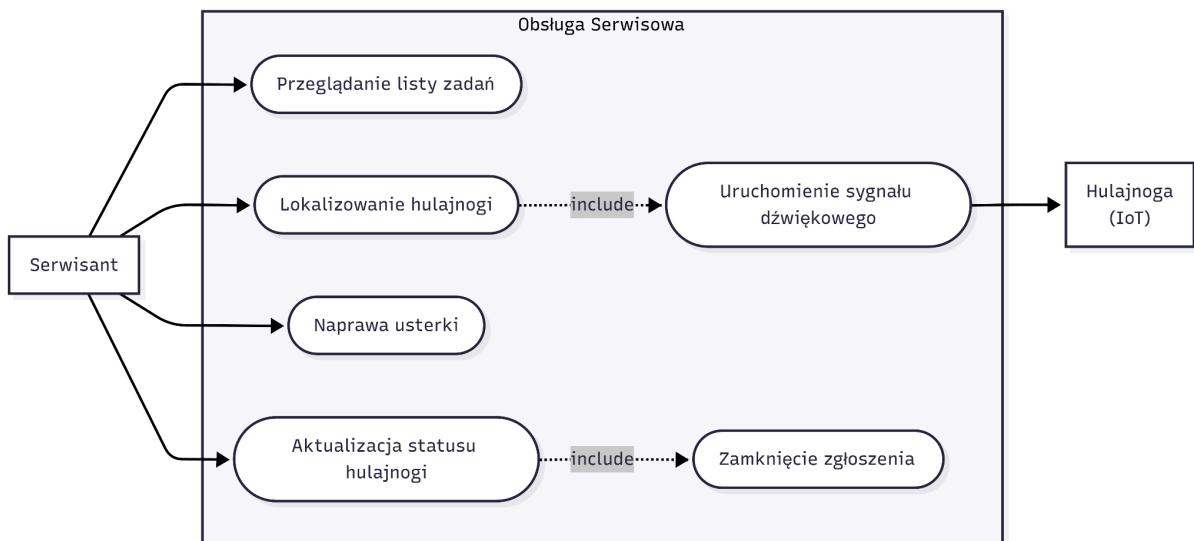


Diagram 3: Perspektywa Serwisanta



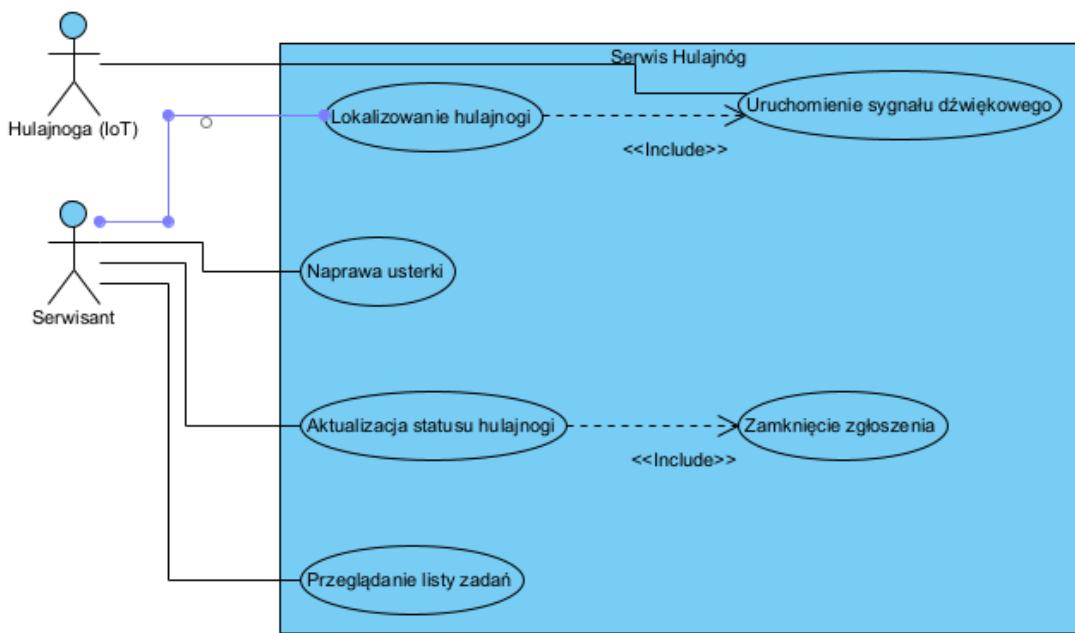
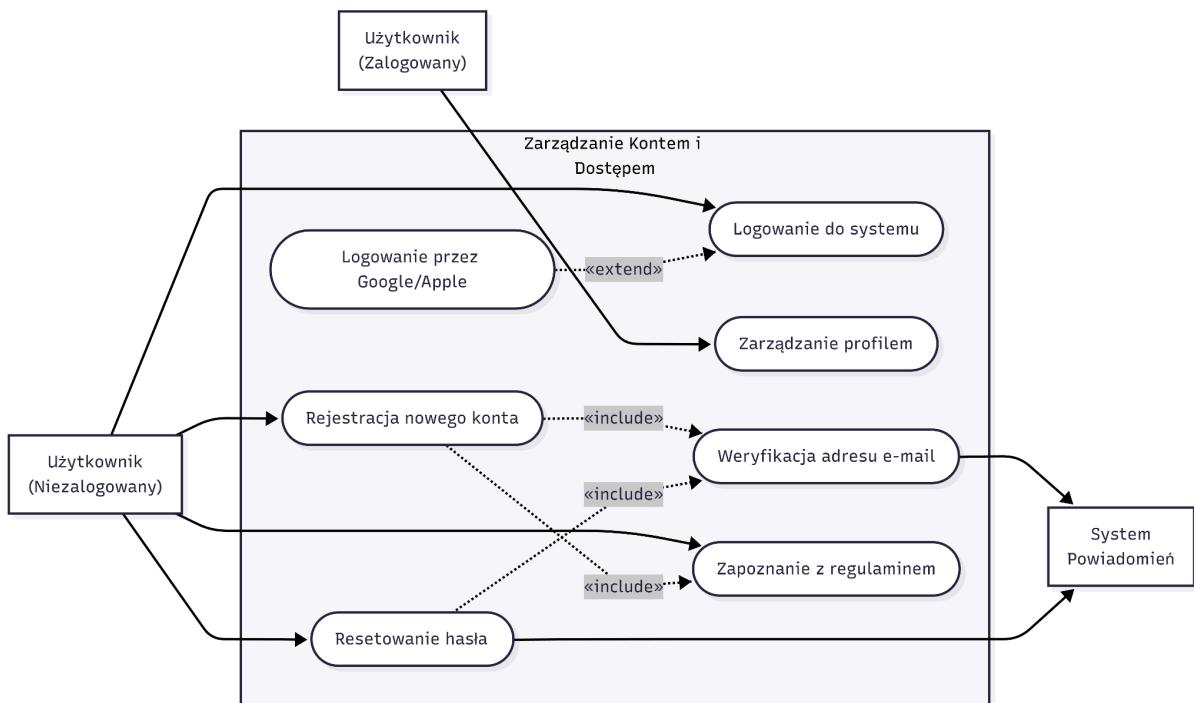


Diagram 4: Zarządzanie Kontem i Dostępu

Scenariusze: S1 (Regulamin), S2 (Rejestracja), S3 (Logowanie), S4 (Reset hasła), S11 (Edycja profilu).

Relacje: Pokazują, że weryfikacja e-mail jest niezbędna przy rejestracji, a logowanie przez social media jest opcjonalnym rozszerzeniem standardowego logowania.



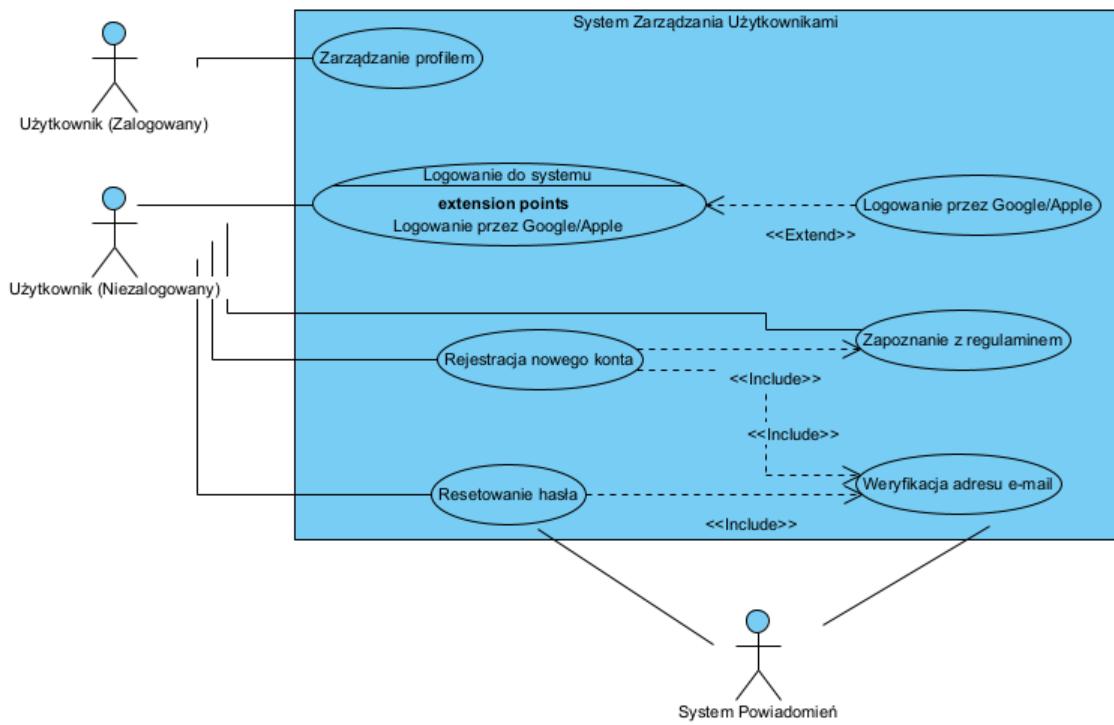
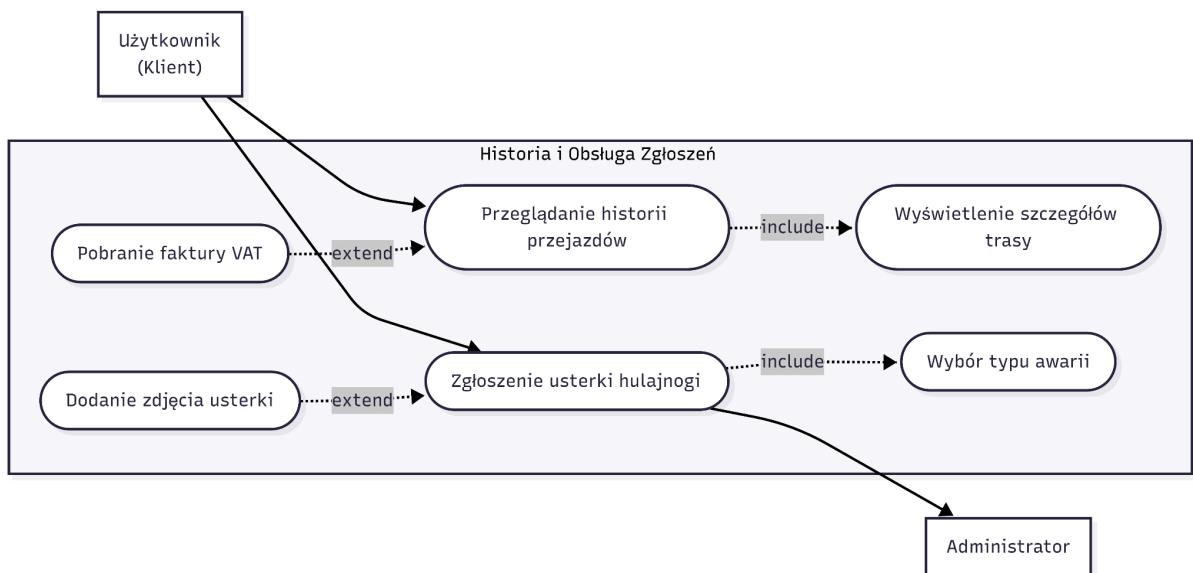


Diagram 5: Historia i Obsługa Zgłoszeń dla klienta

Scenariusze: S8 (Historia przejazdów), S10 (Zgłaszcenie usterki).

Relacje: Zgłoszenie usterki wymaga wybrania typu awarii, ale dodanie zdjęcia jest opcjonalne (extend). Eksport faktury rozszerza przeglądanie historii.



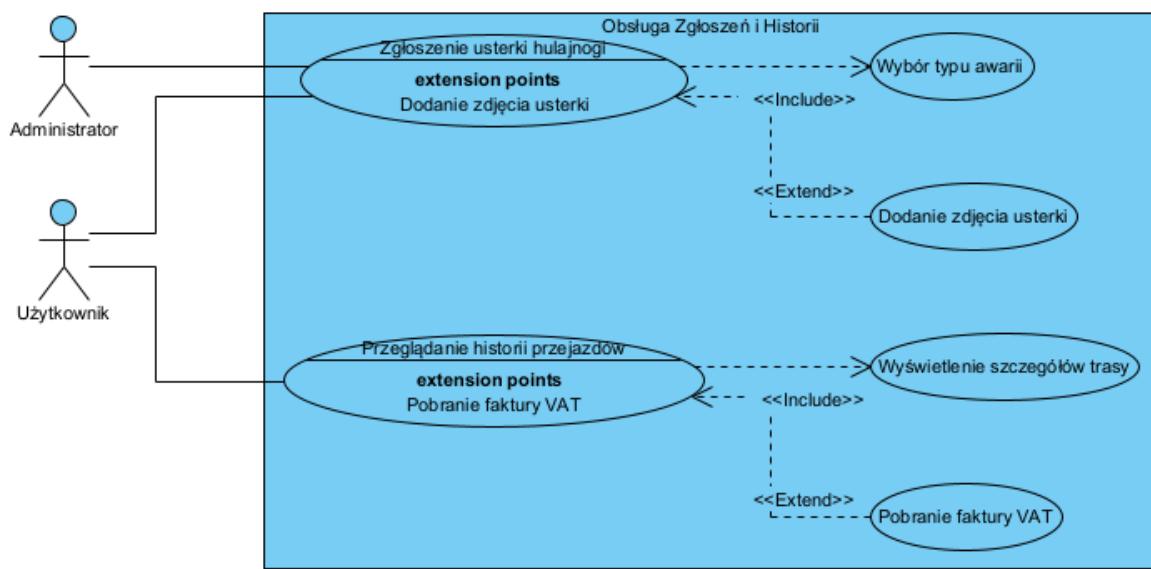
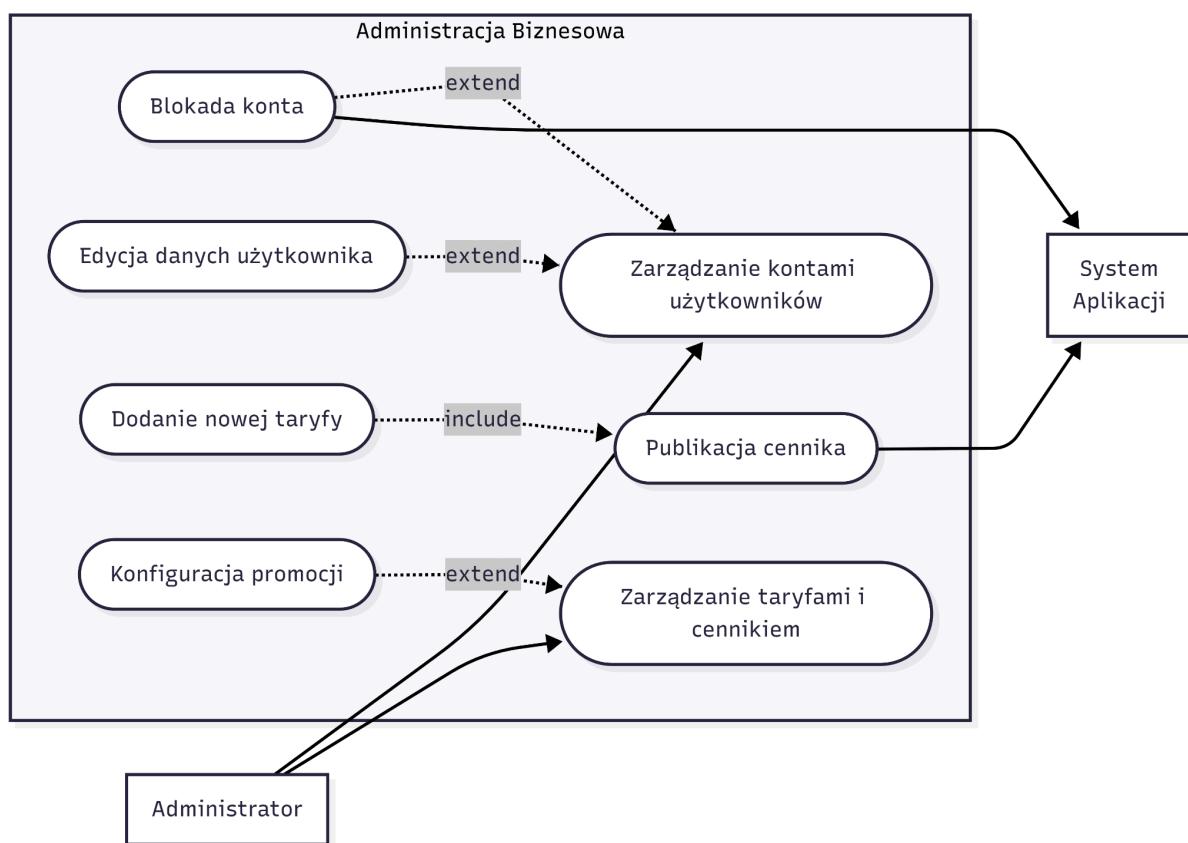


Diagram 6: Administracja Biznesowa (Użytkownicy i Cenniki)

Scenariusze: S13 (Zarządzanie kontami), S15 (Cenniki i taryfy).

Relacje: Blokada konta jest specyficką akcją w ramach zarządzania użytkownikami.

Publikacja cennika jest konieczna, by zmiany weszły w życie.



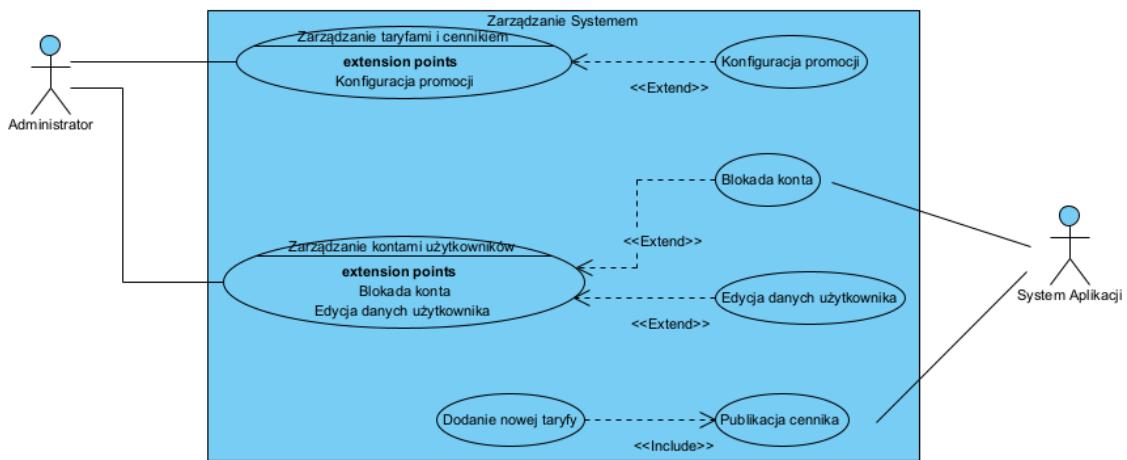


Diagram 7: Monitoring Systemowy i IoT

Scenariusze: S12 (Lokalizowanie), S20 (Status), S25 (Weryfikacja strefy), S26 (Powiadomienia).

Aktorzy: System Monitoringu (Automatyczny), Hulajnoga, Moduł Powiadomień.

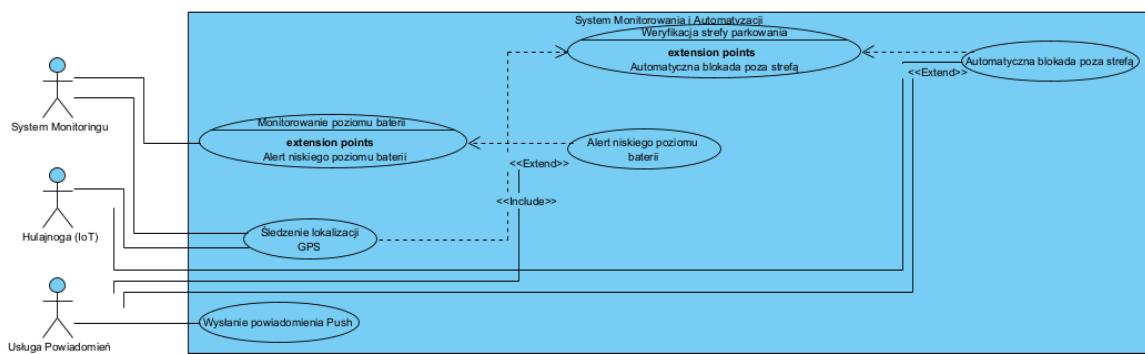
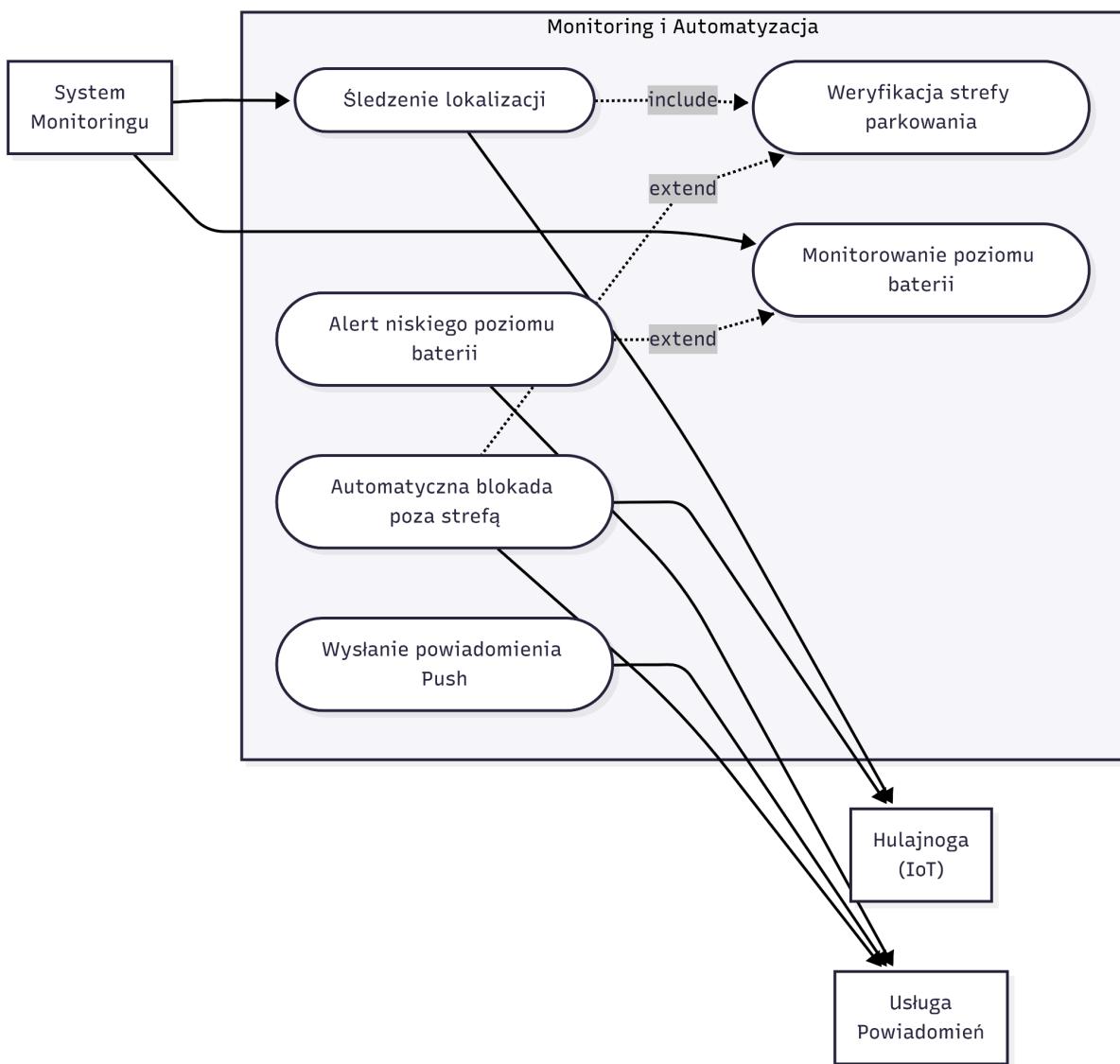
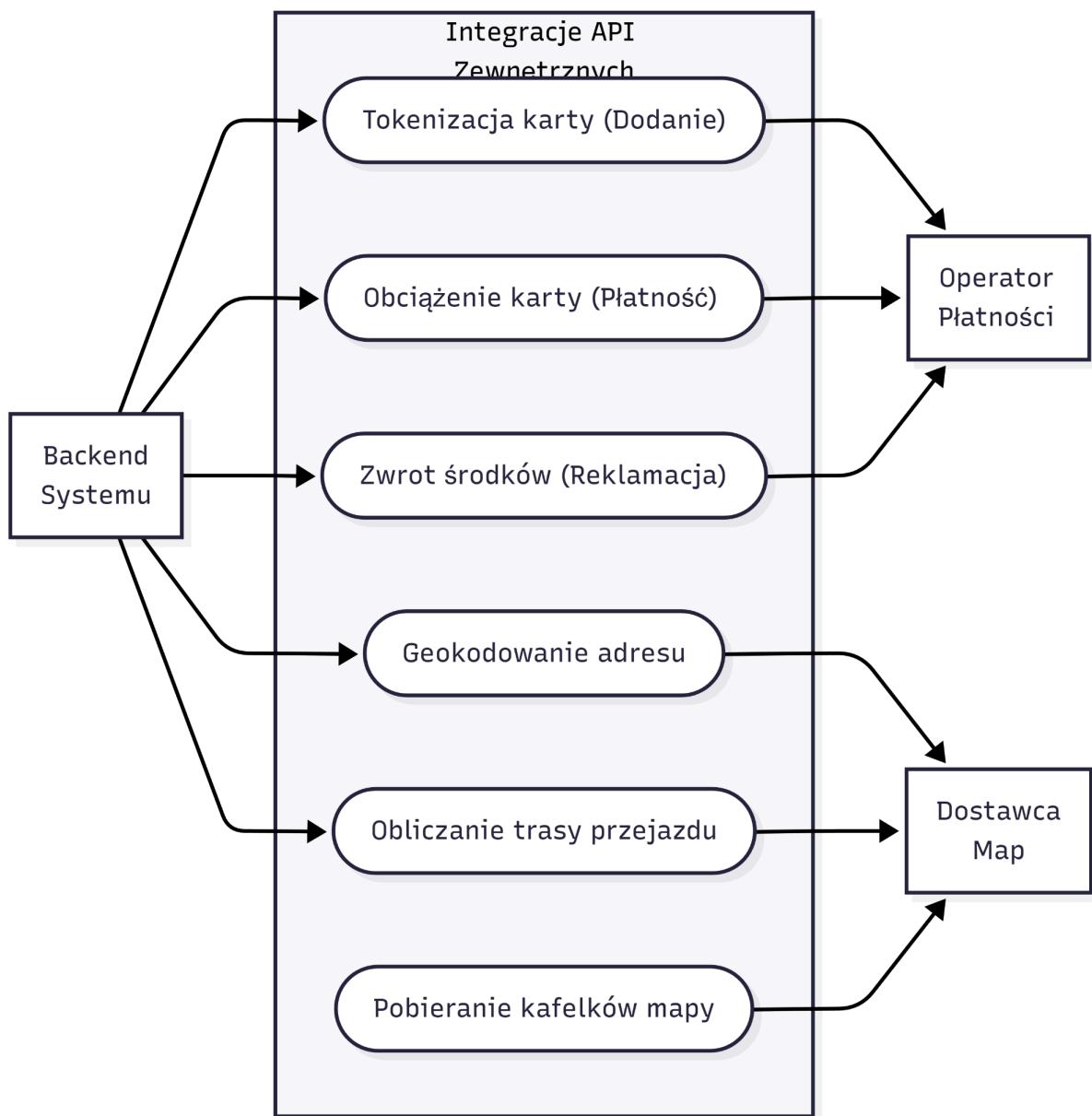


Diagram 8: Integracje Zewnętrzne

Scenariusze: S7 (Metody płatności), S22 (Przetwarzanie), S23 (Tokenizacja), S24 (Mapy).
Aktorzy: Backend Systemu, Operator Płatności (Stripe/PayU), Dostawca Map (Google/Mapbox).



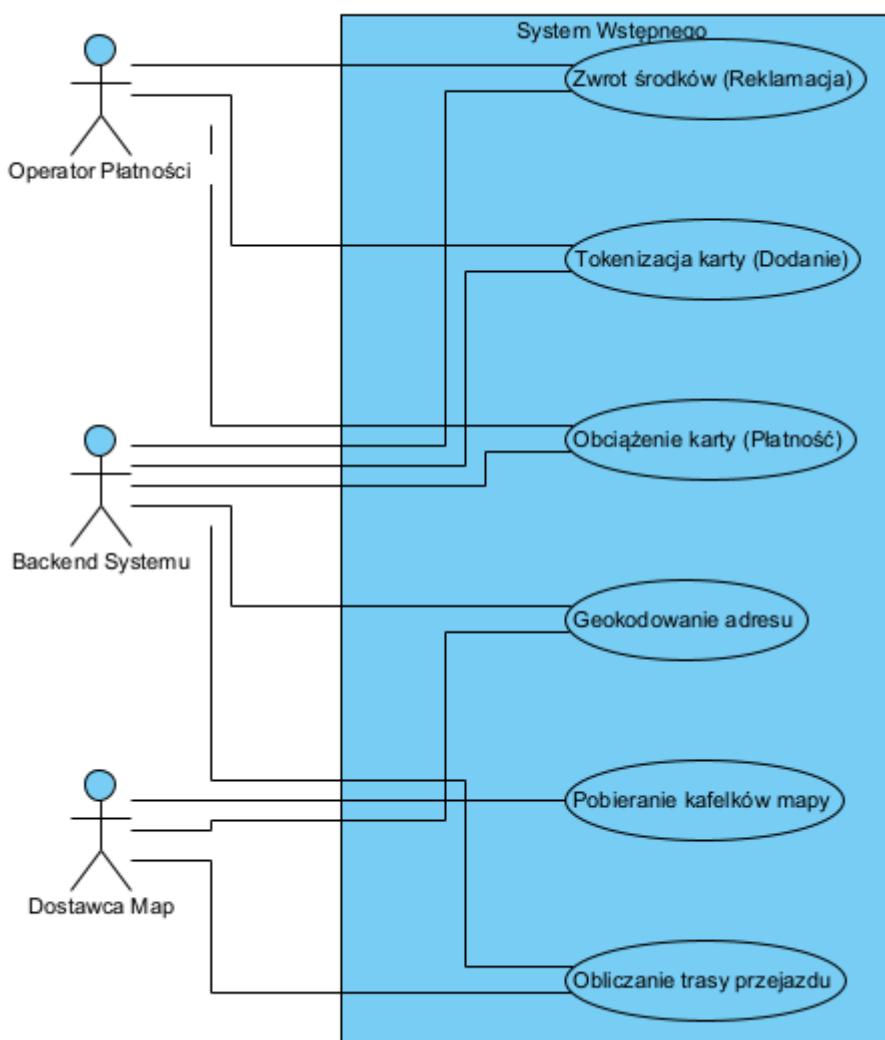
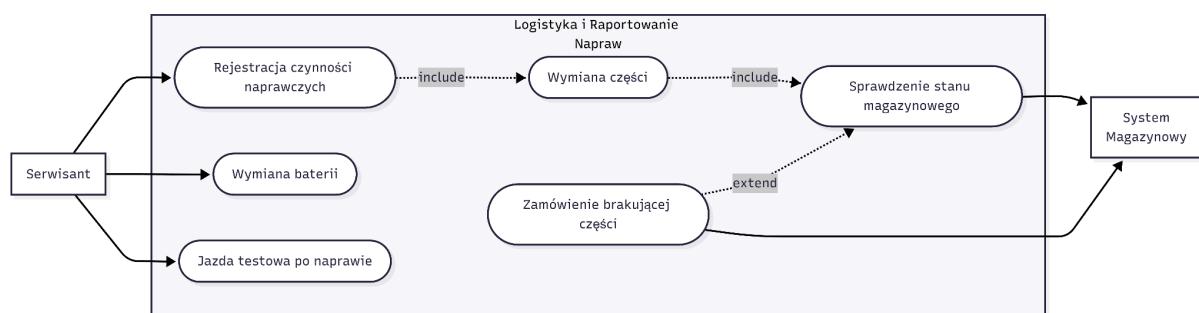


Diagram 9: Logistyka i Naprawy

Scenariusze: S19 (Zadania), S21 (Rejestracja czynności), S14 (Flota).

Aktorzy: Serwisant, System Magazynowy.



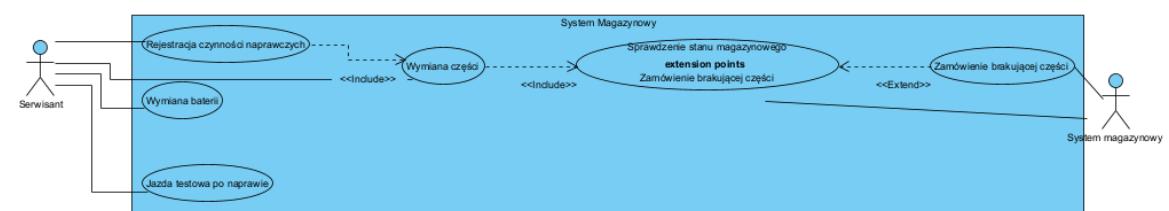


Diagram 10: Analityka i Raportowanie

Scenariusze: S17, S14 (wykorzystanie floty).

Aktorzy: Administrator, Baza Danych.

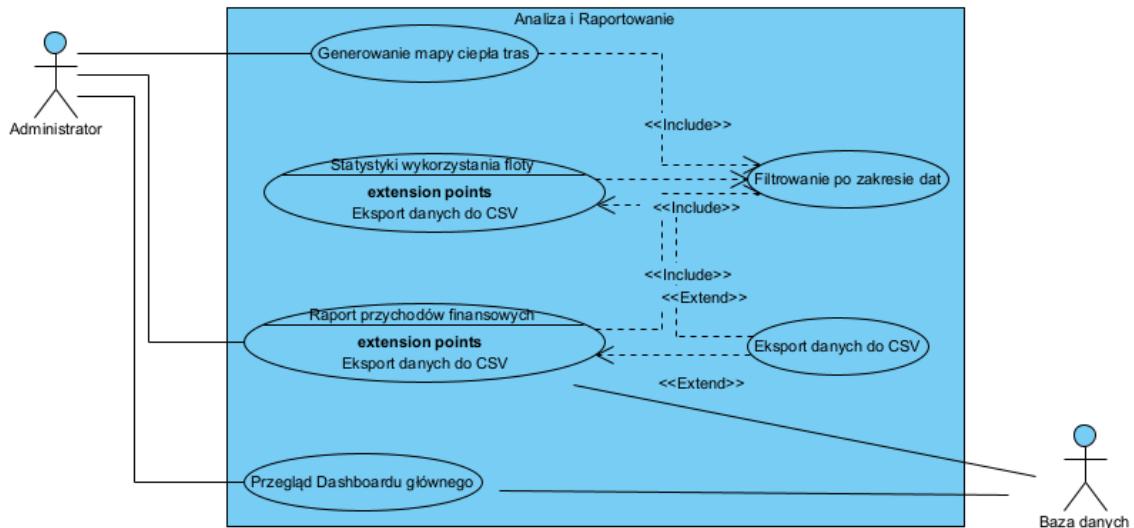
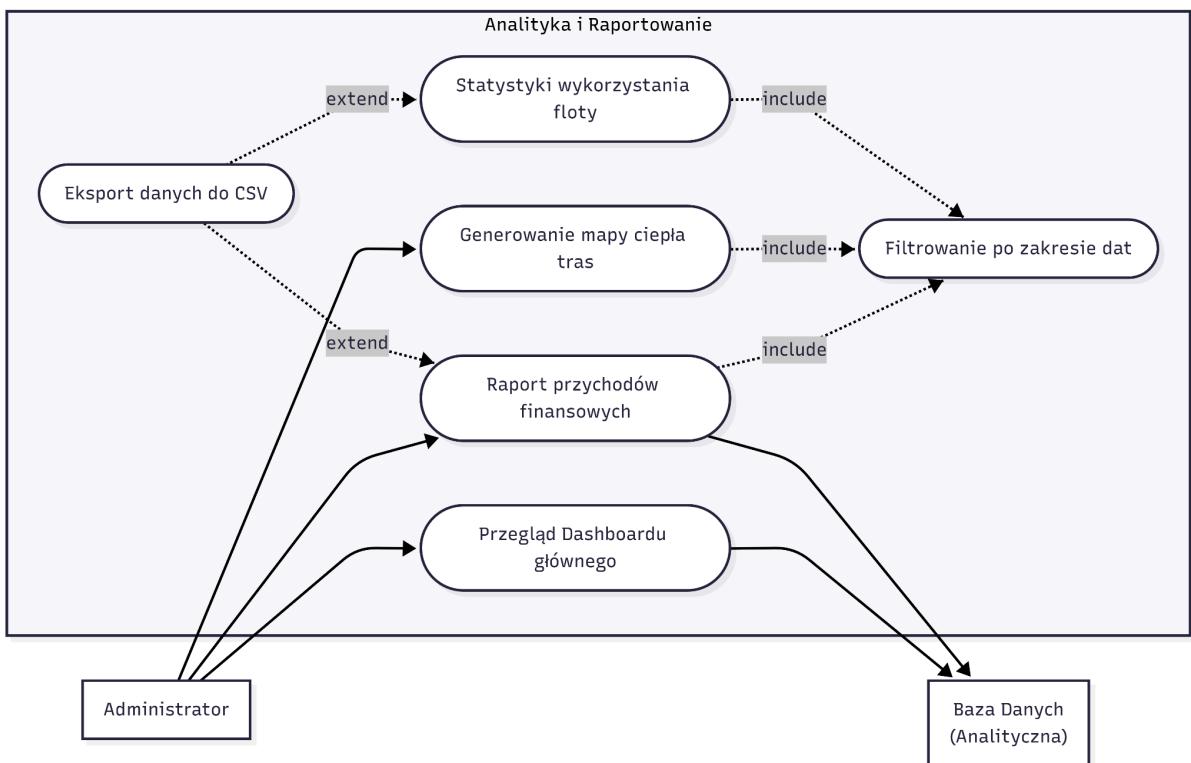


Diagram 11: Marketing i Komunikacja

Scenariusze: S15 (Zarządzanie taryfami/promocje), S26 (Powiadomienia),
 Aktorzy: Administrator (Marketingowiec), System Powiadomień.,

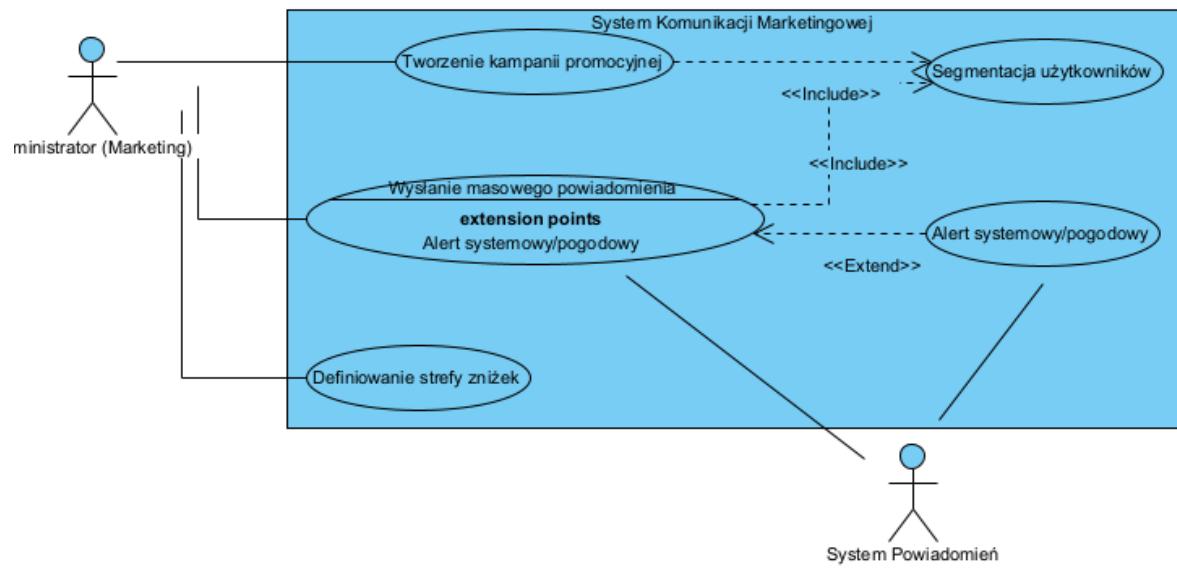
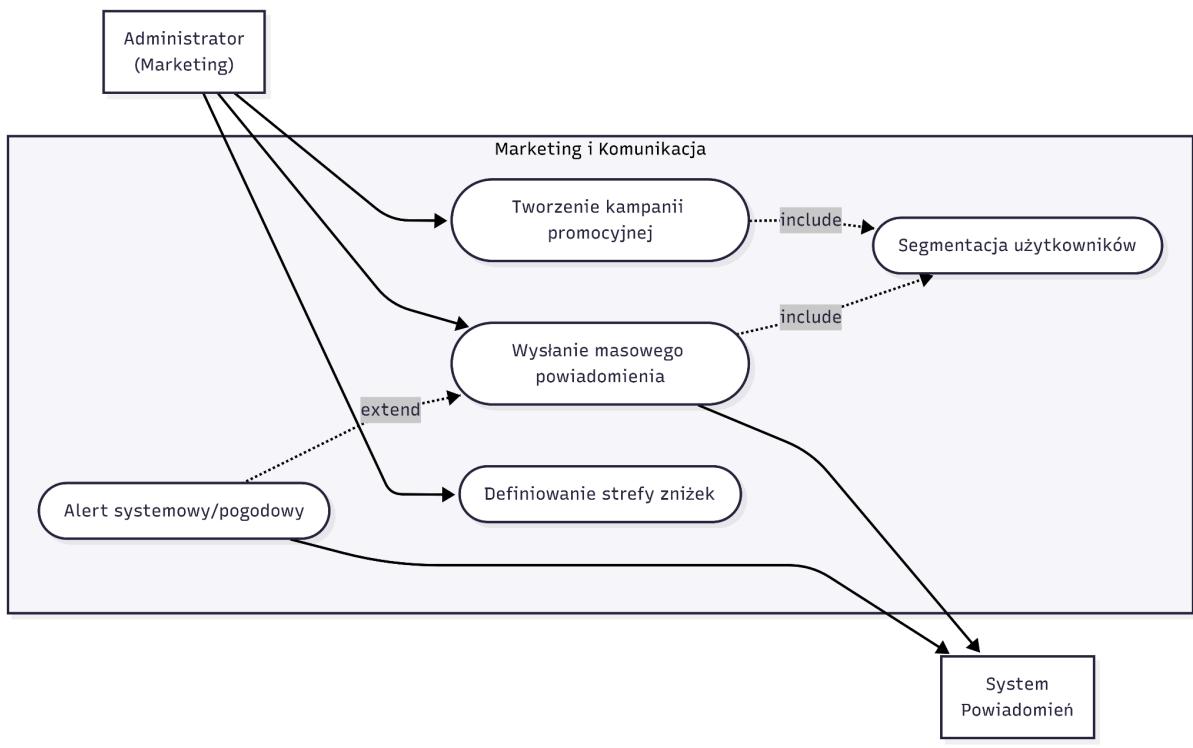
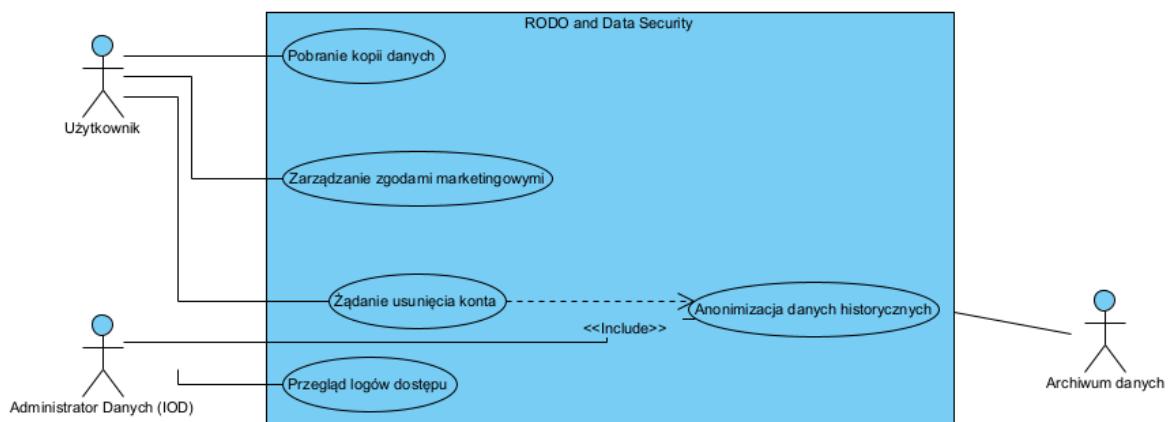
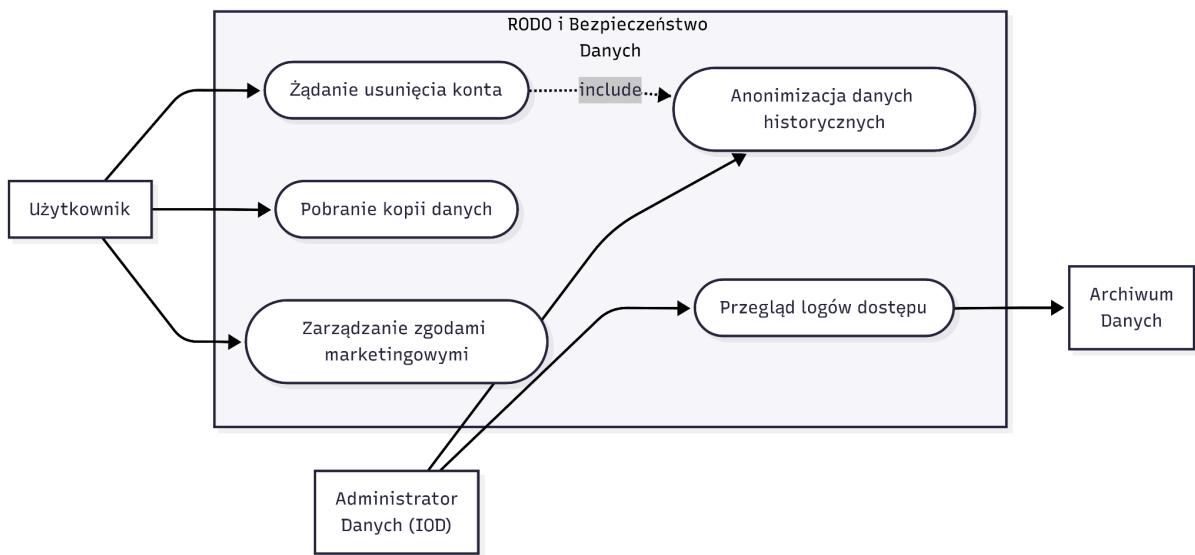


Diagram 12: Bezpieczeństwo Danych i RODO

Scenariusze: S13 (Zarządzanie kontami), S11 (Profil), Wymagania niefunkcjonalne.
Aktorzy: Użytkownik, Administrator Danych (IOD).



5. Diagramy klas

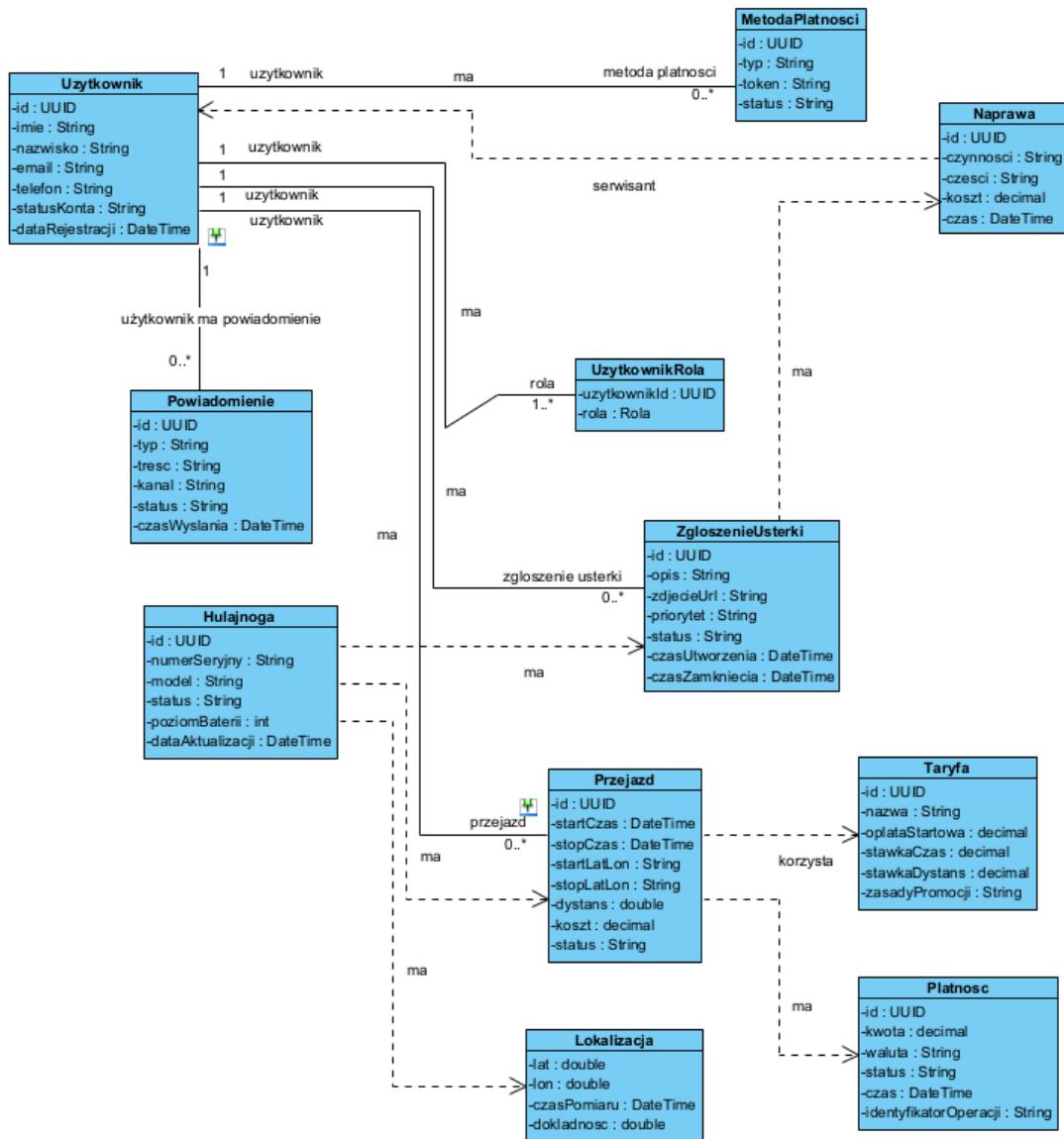


Diagram klas systemu wynajmu hulajnóg: zawiera użytkowników, ich role, płatności i taryfy. System zarządza przejazdami, lokalizacją i stanem hulajnóg oraz procesem zgłoszenia usterek i naprawami realizowanymi przez serwisantów.

6. Diagramy sekwencji.

Diagram 1: S2. Rejestracja nowego konta

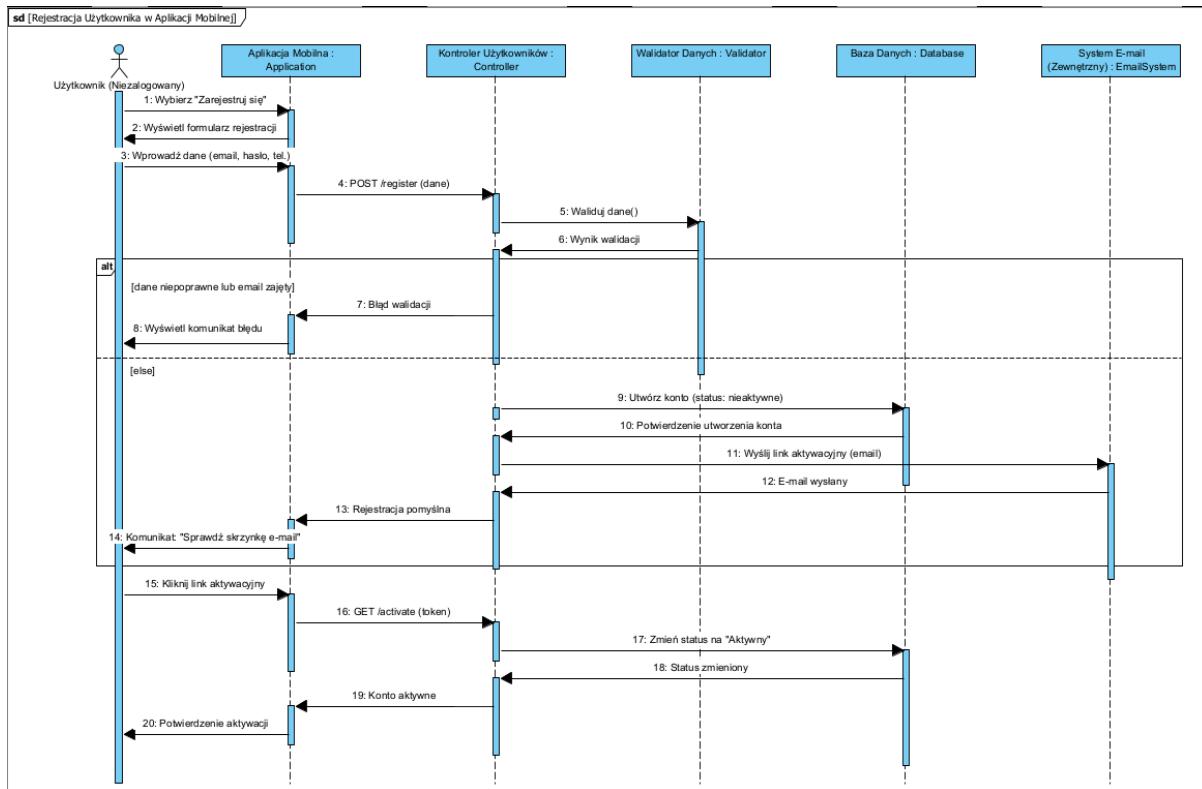


Diagram 2: S3. Logowanie do systemu

sd [Sekwencja Logowania do Aplikacji Mobilnej]

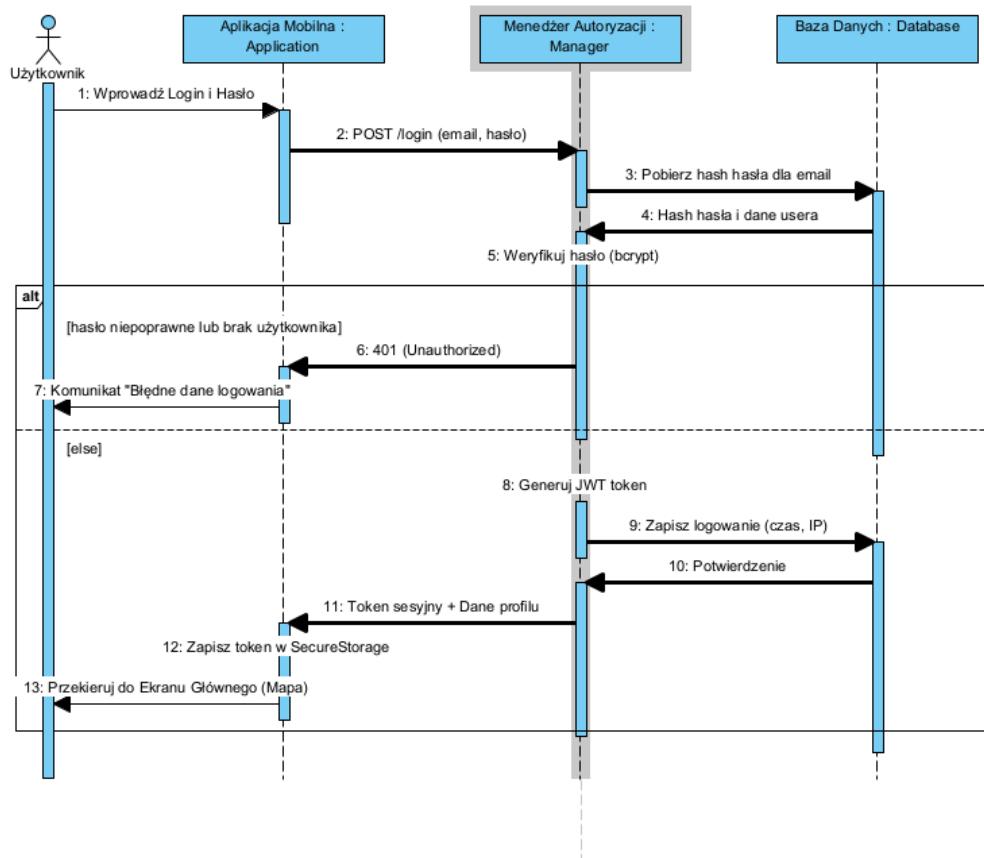


Diagram 3: S9. Doładowanie portfela

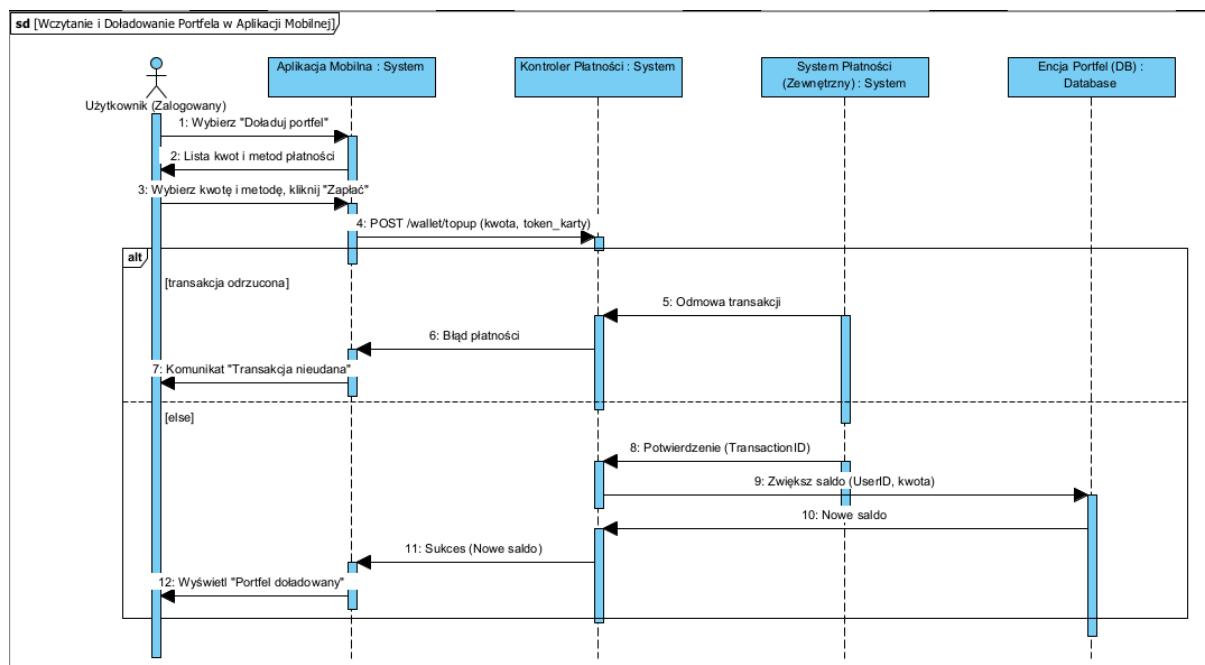


Diagram 4: S5. Rozpoczynanie przejazdu

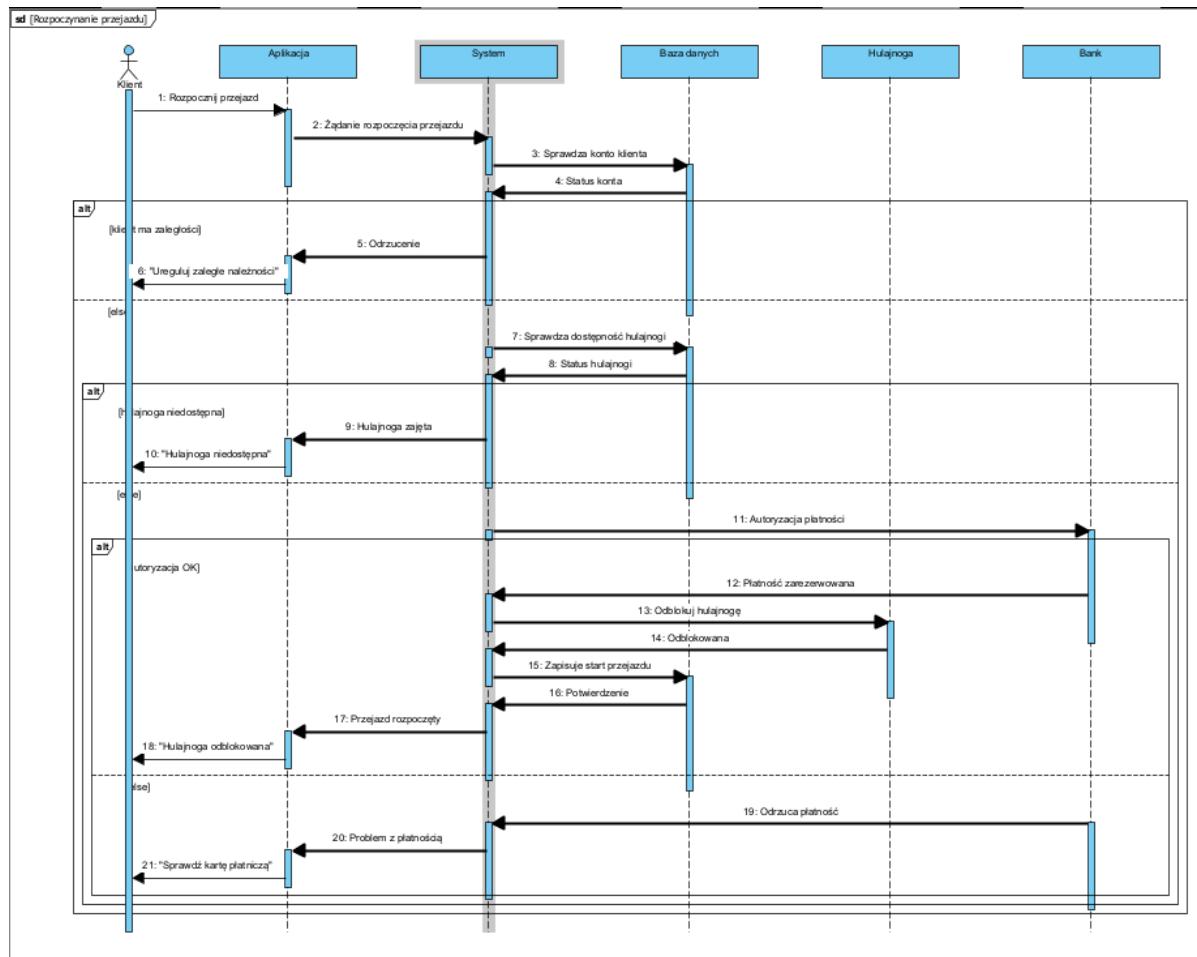


Diagram sekwencji „Rozpoczynanie przejazdu”. Ilustruje proces weryfikacji klienta, dostępności hulajnogi i autoryzacji płatności w banku. Po pomyślnej walidacji system odblokowuje pojazd i zapisuje start przejazdu w bazie danych.

Diagram 5: S6. Ukończenie przejazdu

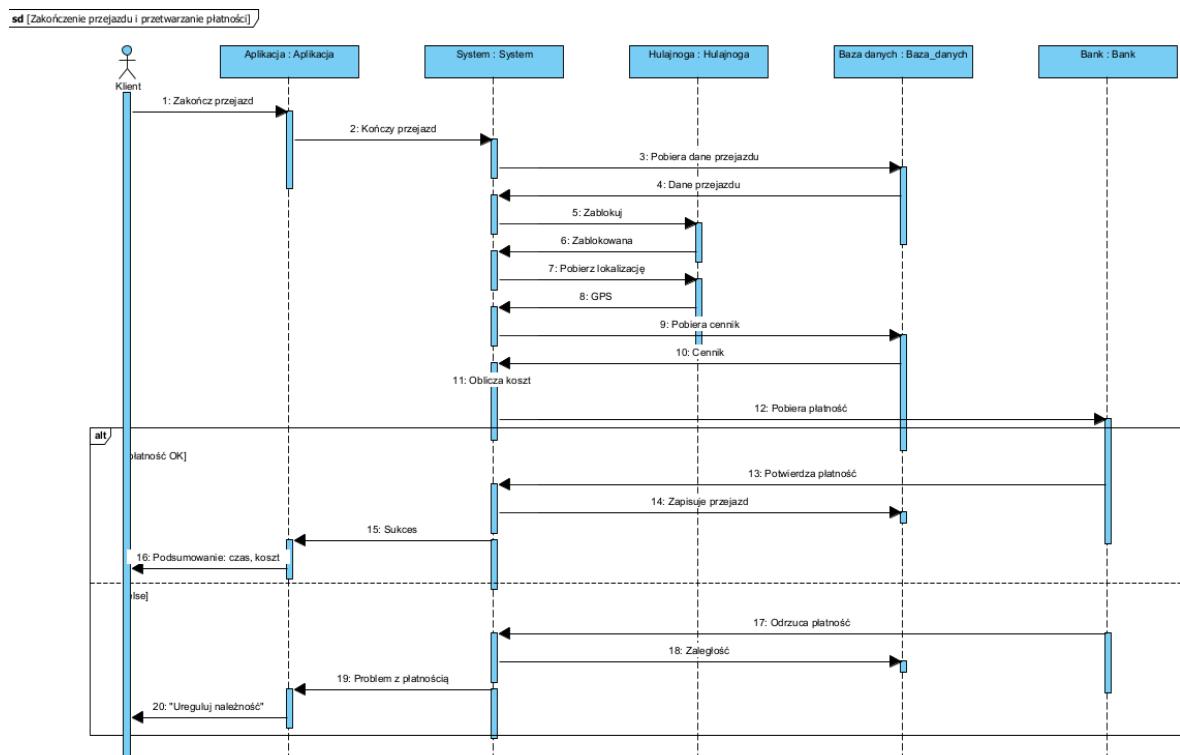


Diagram sekwencji zakończenia przejazdu. System blokuje pojazd, pobiera GPS, oblicza koszt i pobiera opłatę z banku. Zapisuje przejazd lub informuje o zaległości.

Diagram 6: S10. Zgłaszanie usterki hulajnogi

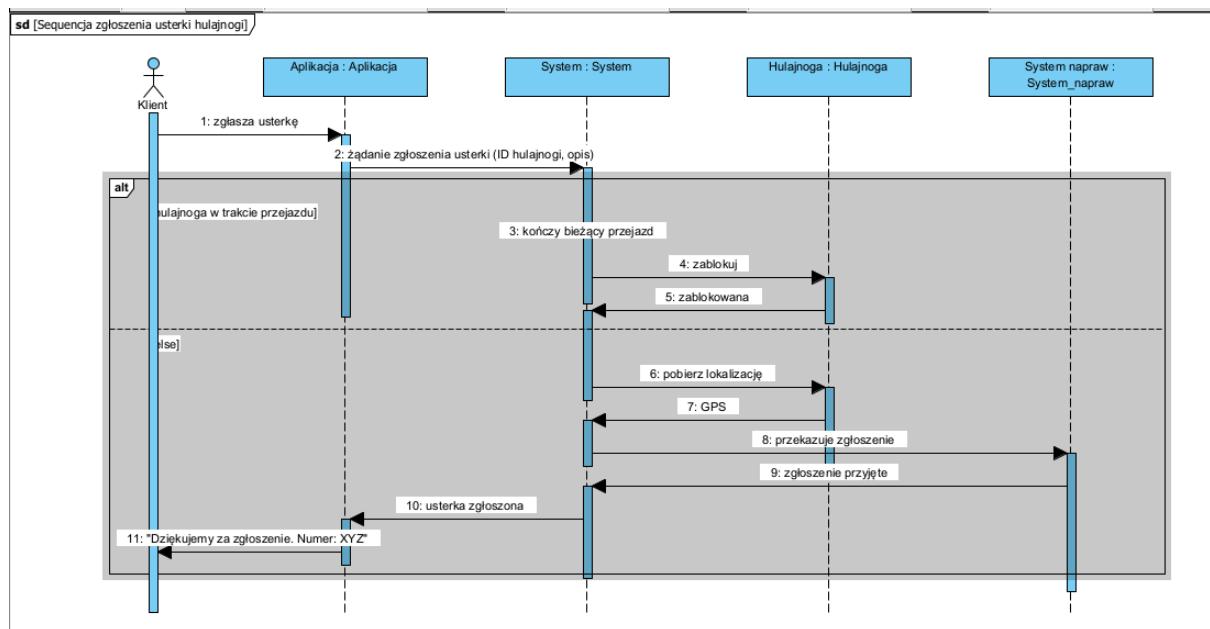


Diagram sekwencji zgłoszenia usterki. System kończy jazdę, blokuje pojazd, pobiera lokalizację i przekazuje zgłoszenie do serwisu, przesyłając klientowi potwierdzenie z numerem.

Diagram 7: S13 Zarządzanie kontami użytkowników

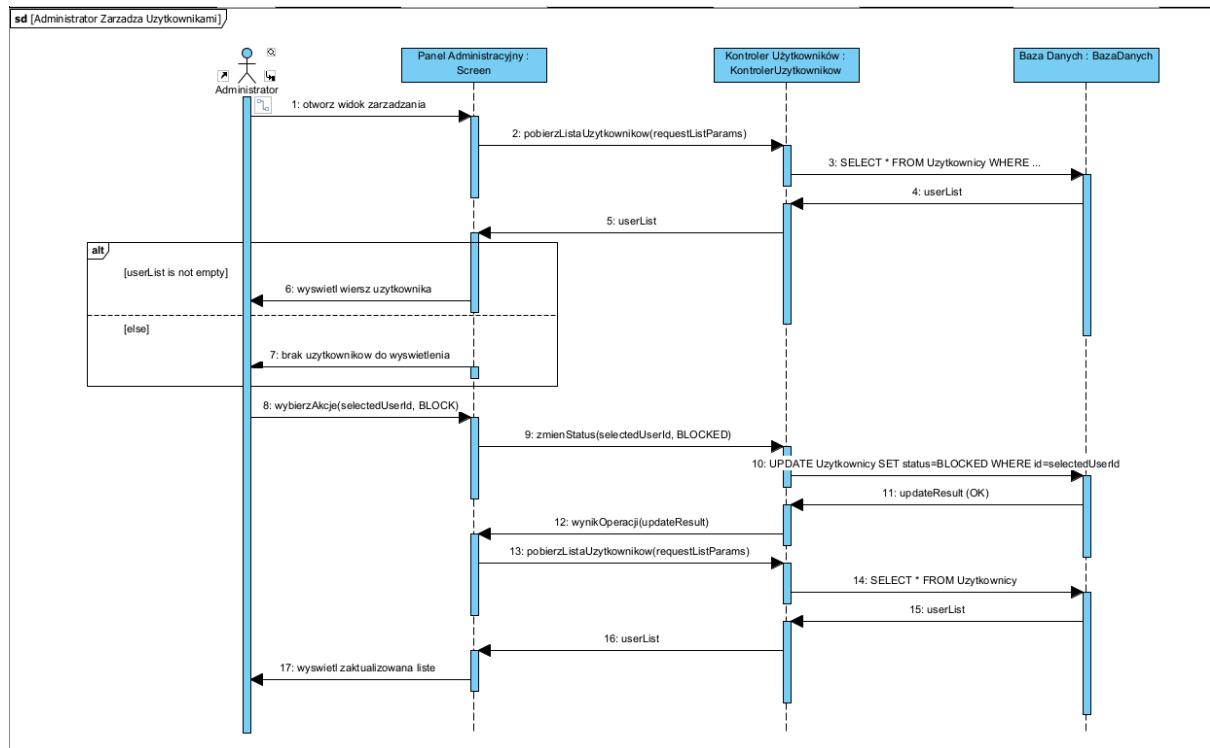


Diagram sekwencji: administrator zarządza użytkownikami. Pobiera listę z bazy, blokuje wybranego użytkownika zmieniając status) i wyświetla odświeżony widok panelu.

Diagram 8: S15 Zarządzanie taryfami i cennikiem

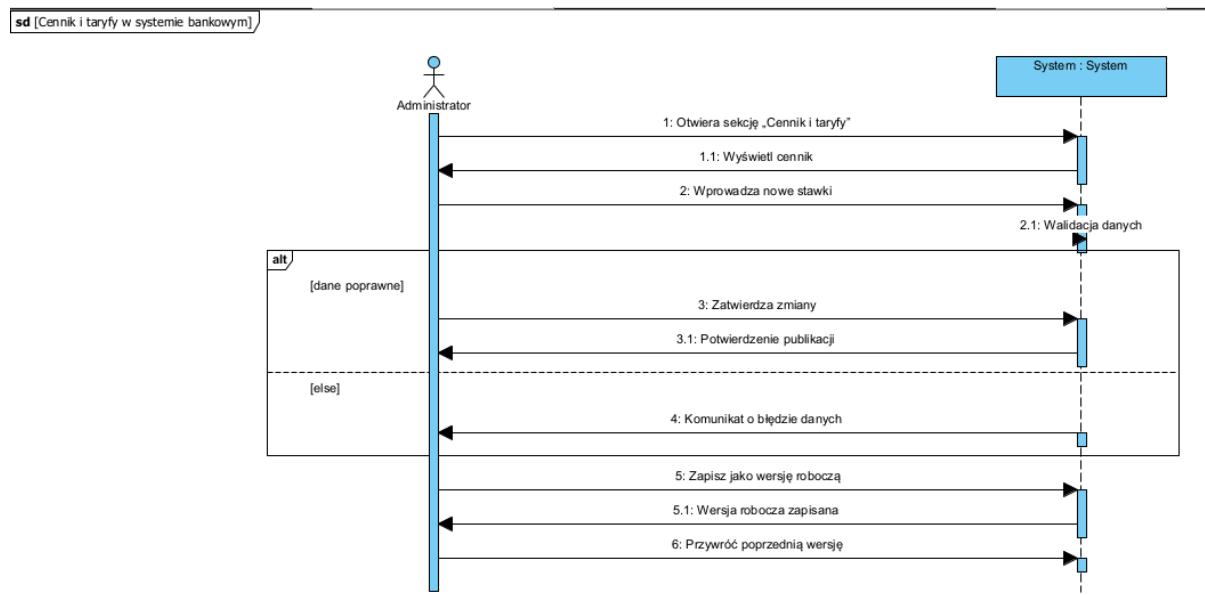


Diagram sekwencji: edycja cennika. Administrator aktualizuje stawki, system je waliduje. Możliwa publikacja zmian, zapis wersji roboczej lub przywrócenie poprzedniej konfiguracji.

Diagram 9: S16 Zarządzanie strefami

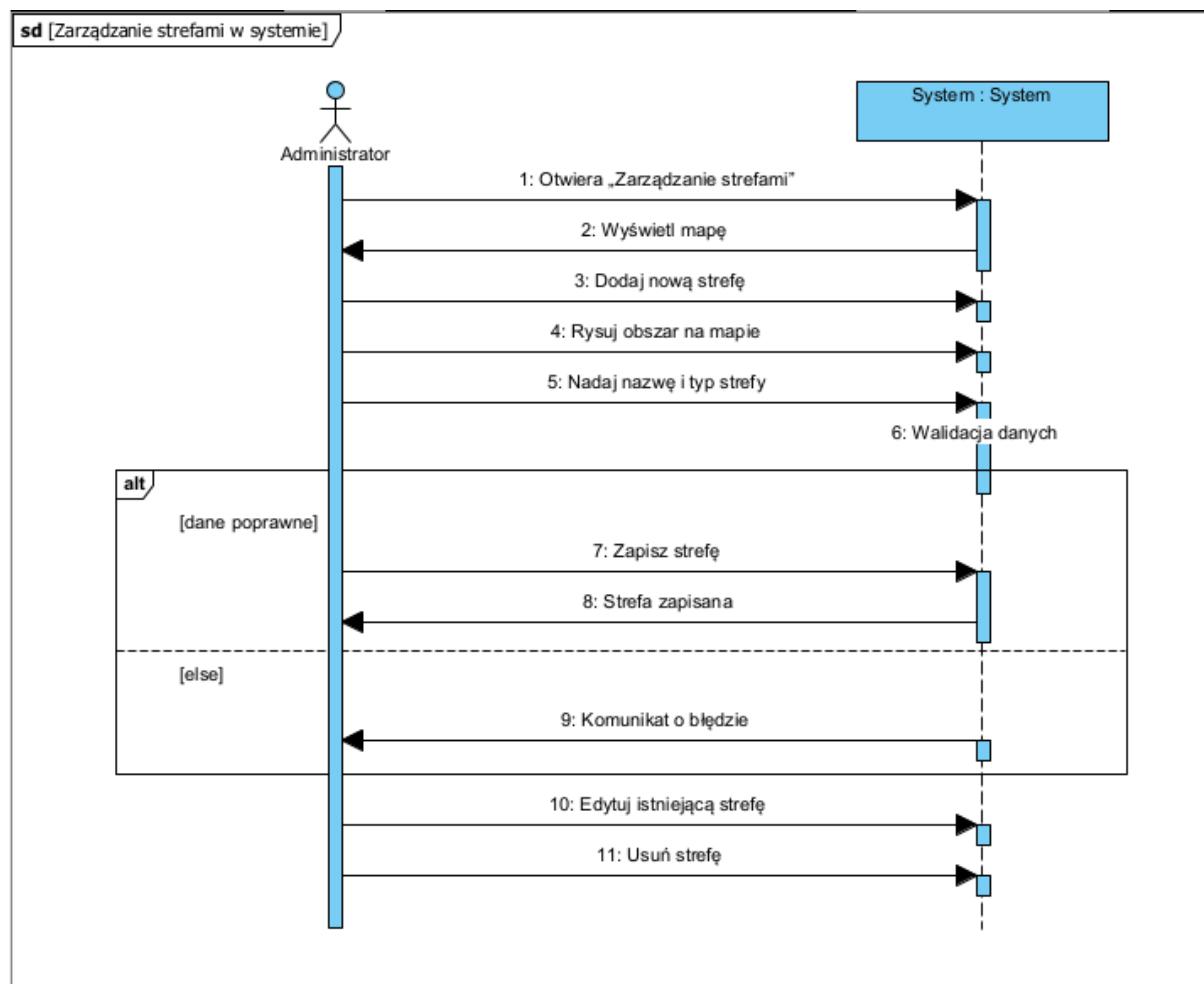


Diagram sekwencji: zarządzanie strefami. Administrator rysuje obszar na mapie, nadaje nazwę i typ. System waliduje i zapisuje nową strefę. Umożliwia także edycję i usuwanie stref.

Diagram 10: S18. Zarządzanie zgłoszeniami serwisowymi

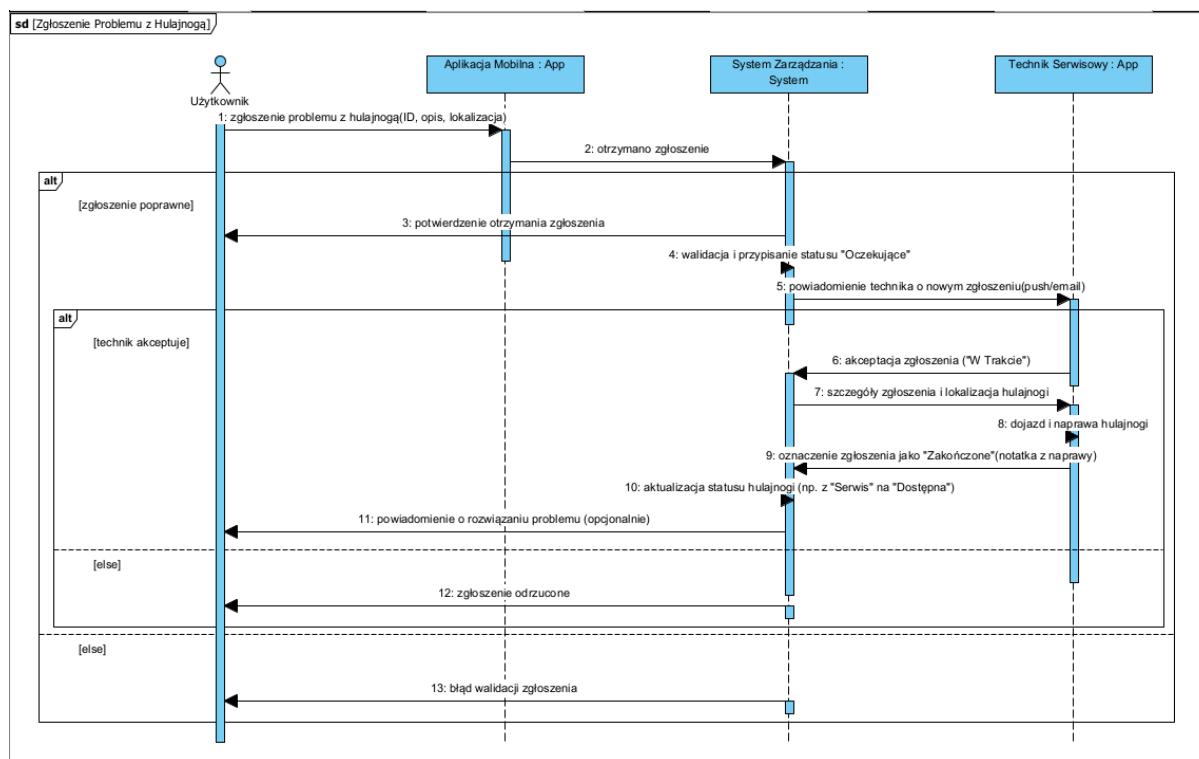


Diagram 11: S19. Przeglądanie listy zadań serwisowych

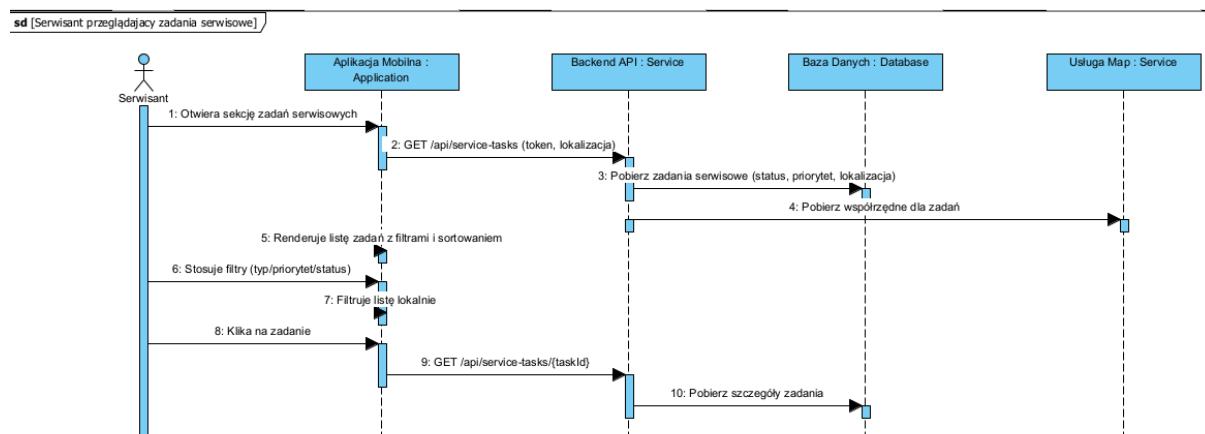
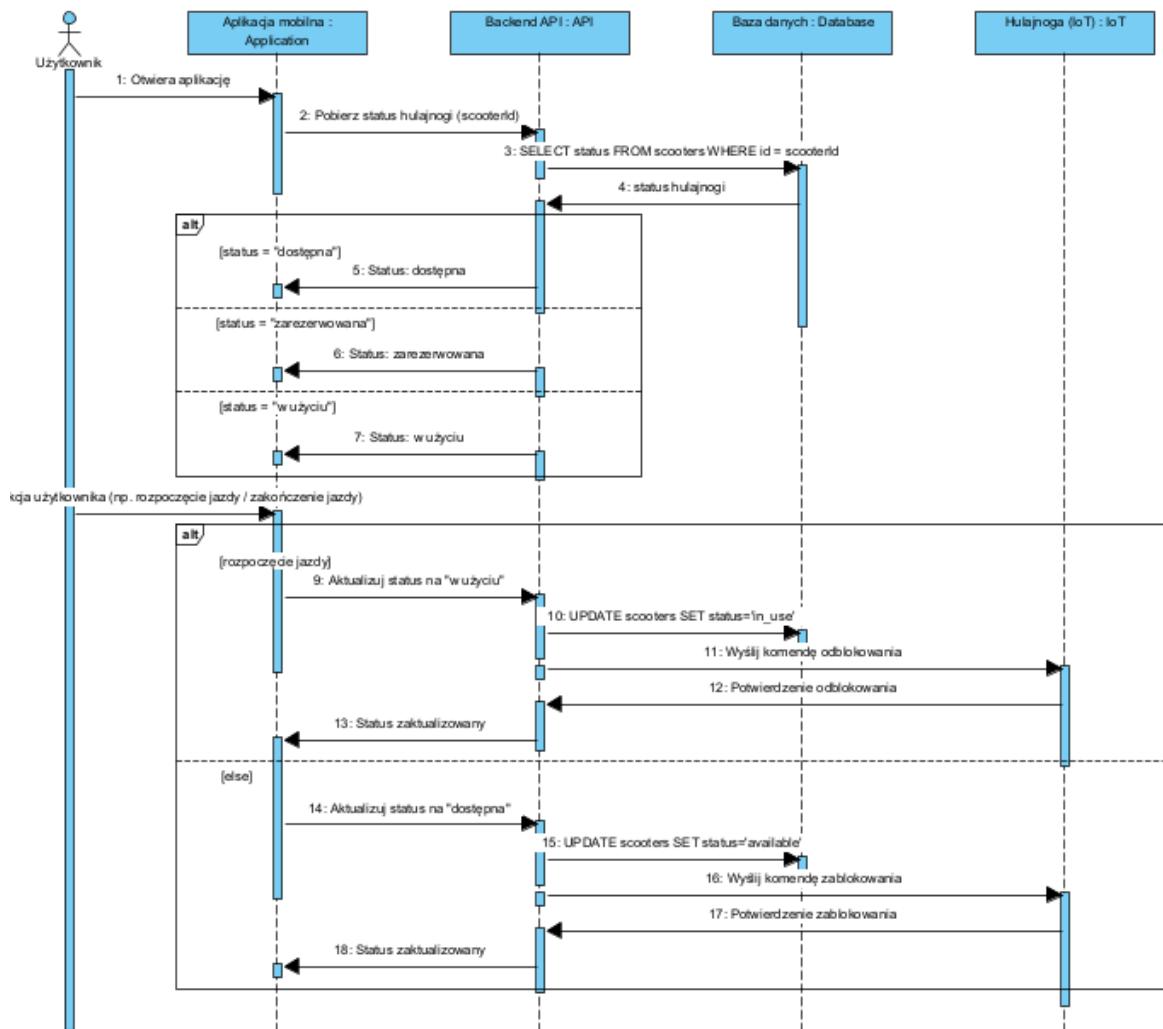
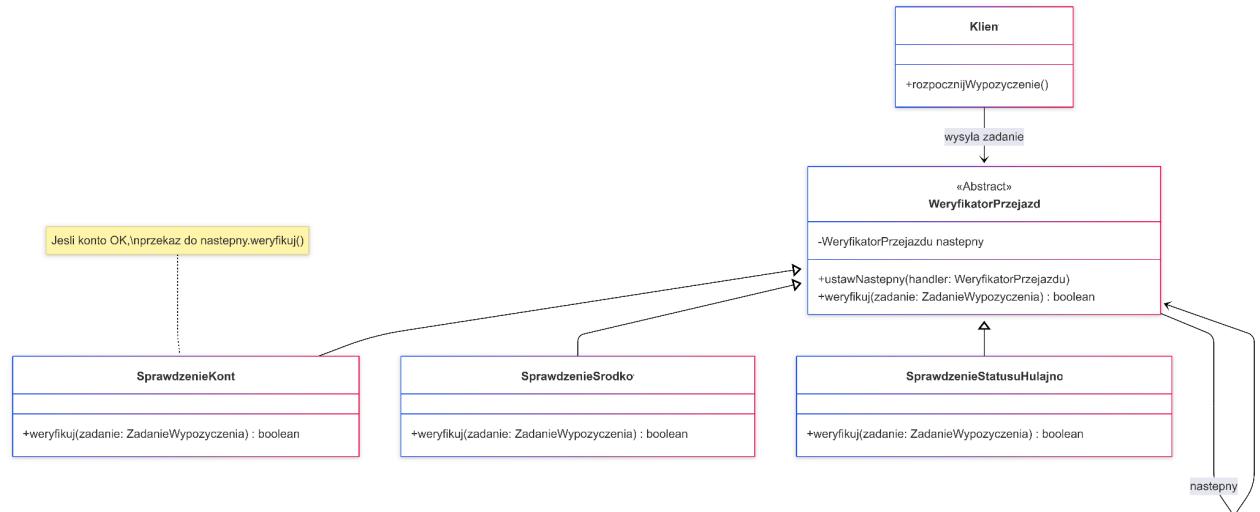


Diagram 12: S20 – Aktualizacja statusu hulajnogi

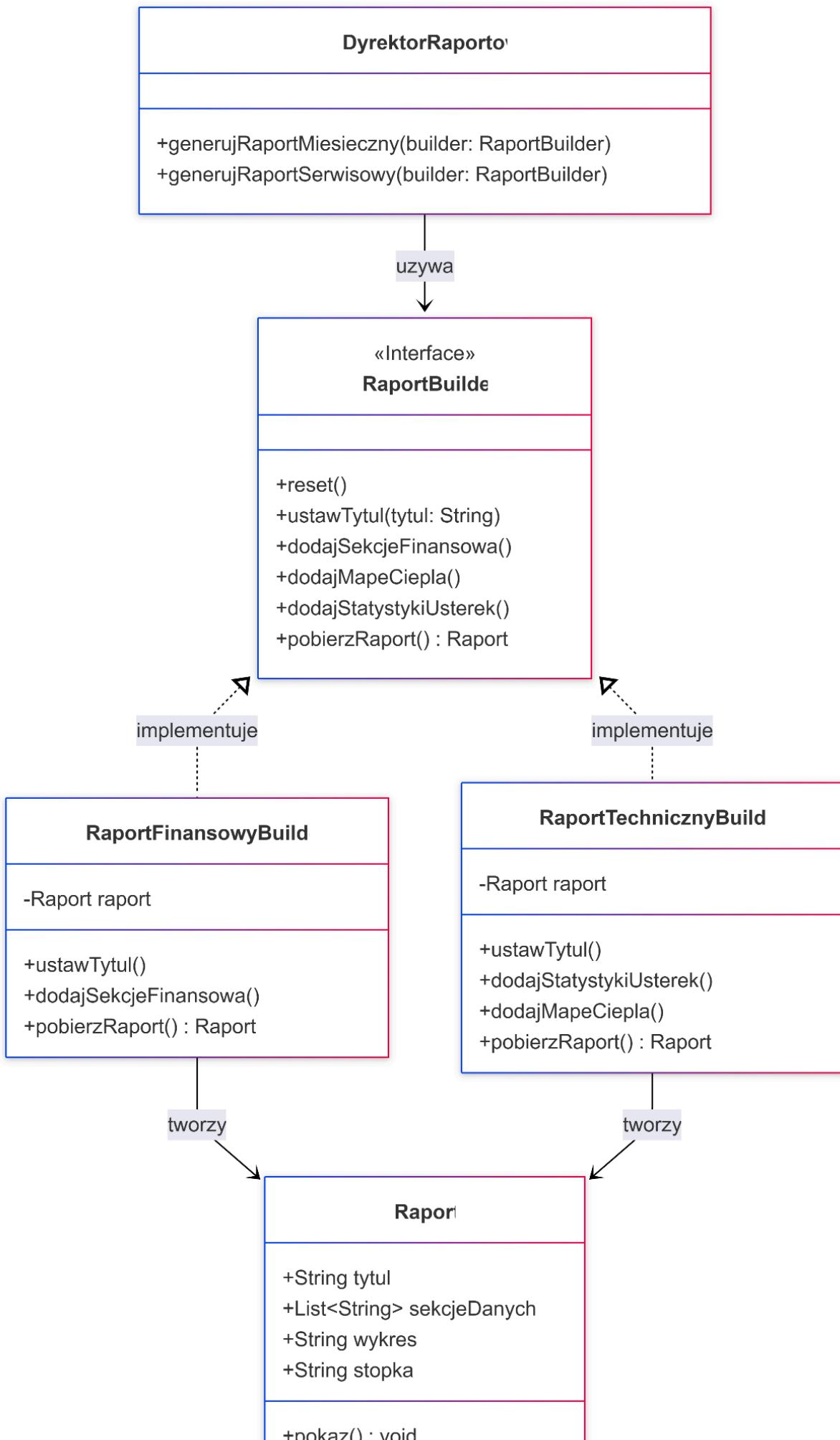


7. Wzorce projektowe.

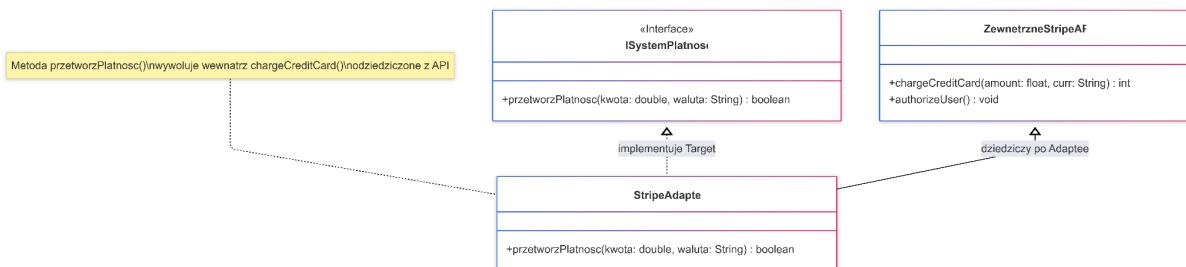
Wzorzec Singleton



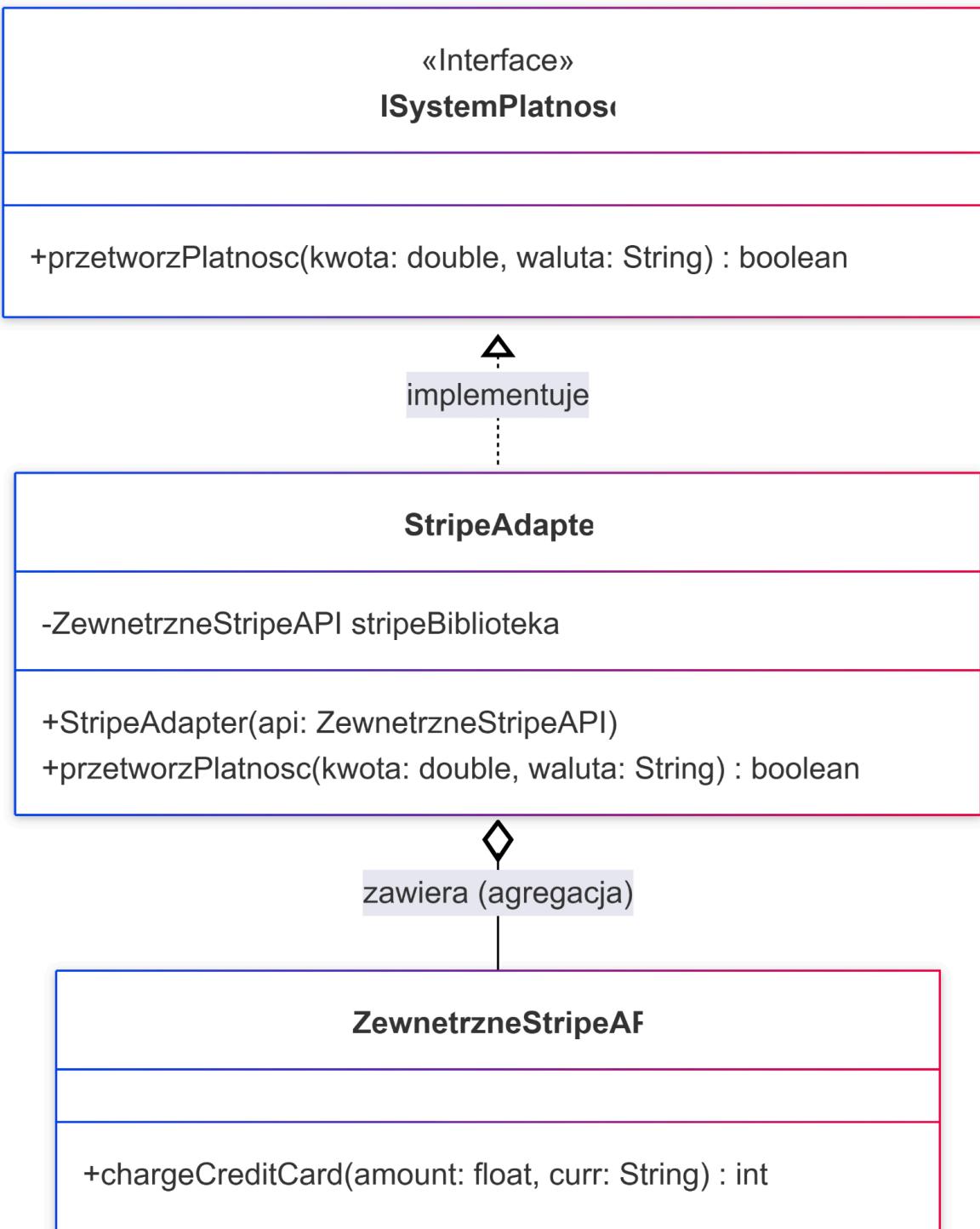
Wzorzec Builder



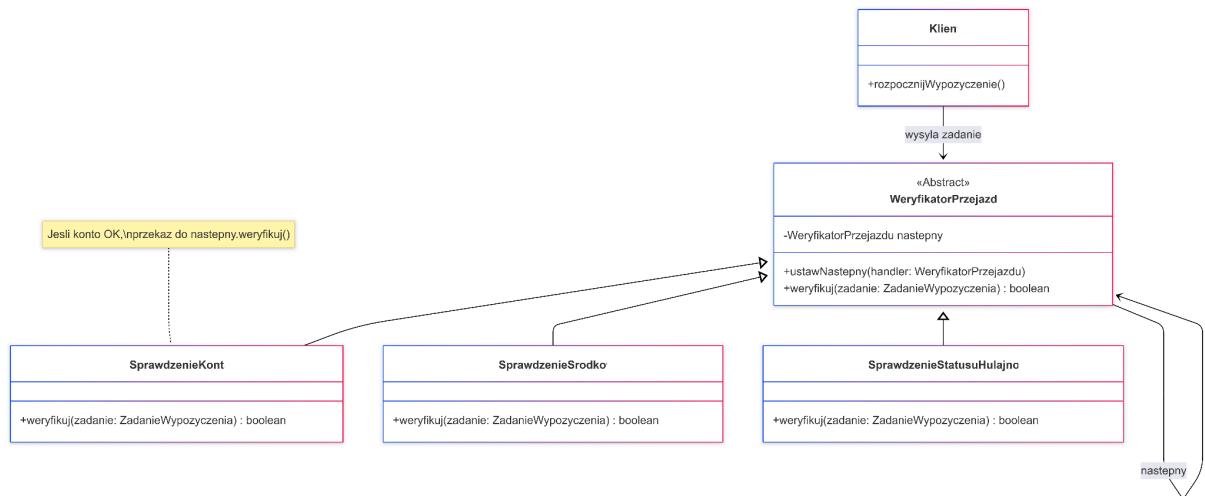
Wzorzec Adapter Klasowy



Wzorzec Adapter Obiektowy



Wzorzec Chain of Responsibility



8. Diagramy dynamiczne: czynności, aktywności.

Diagram aktywności 1: S2. Rejestracja nowego konta

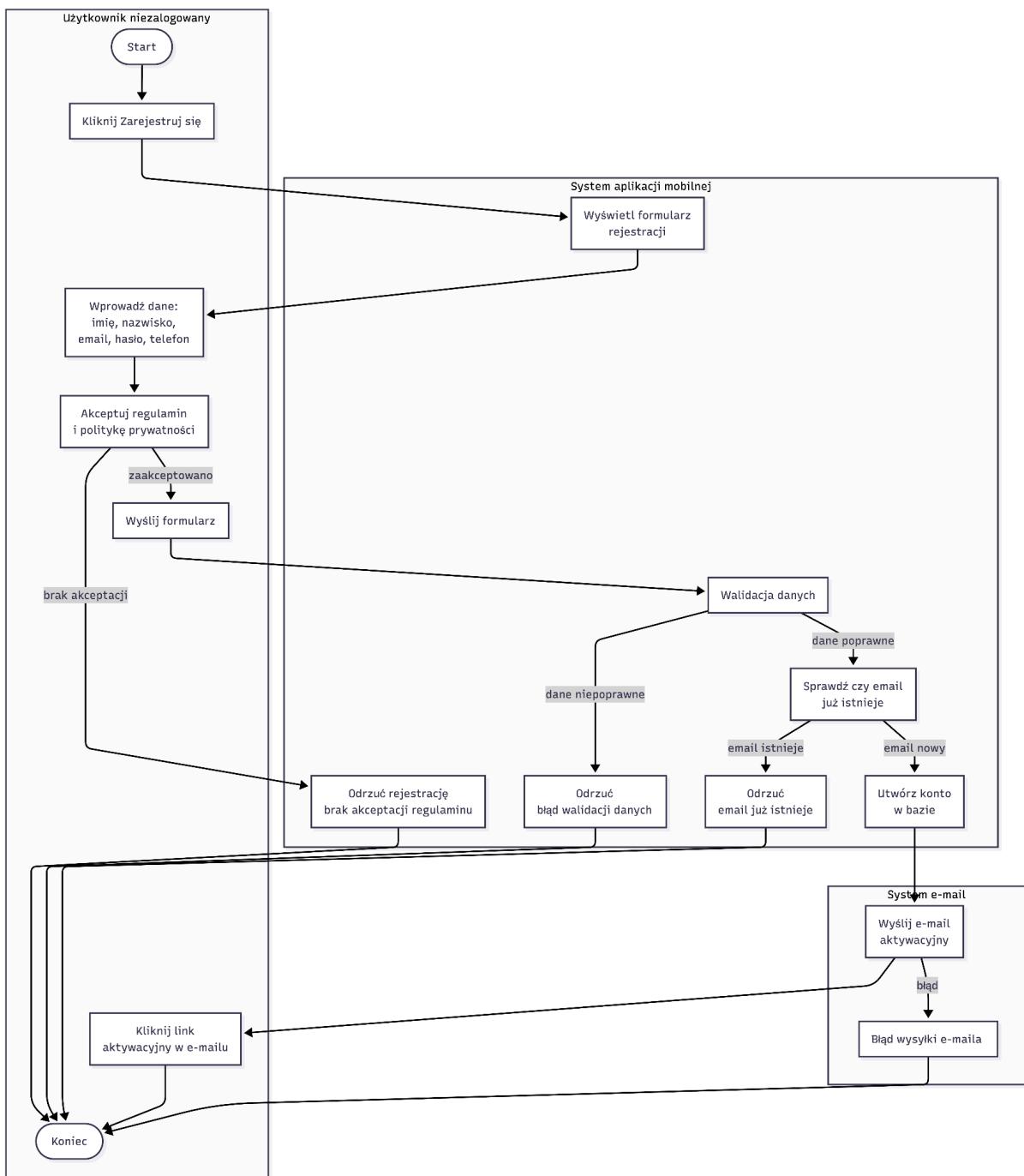


Diagram aktywności 2: S5. Rozpoczynanie przejazdu

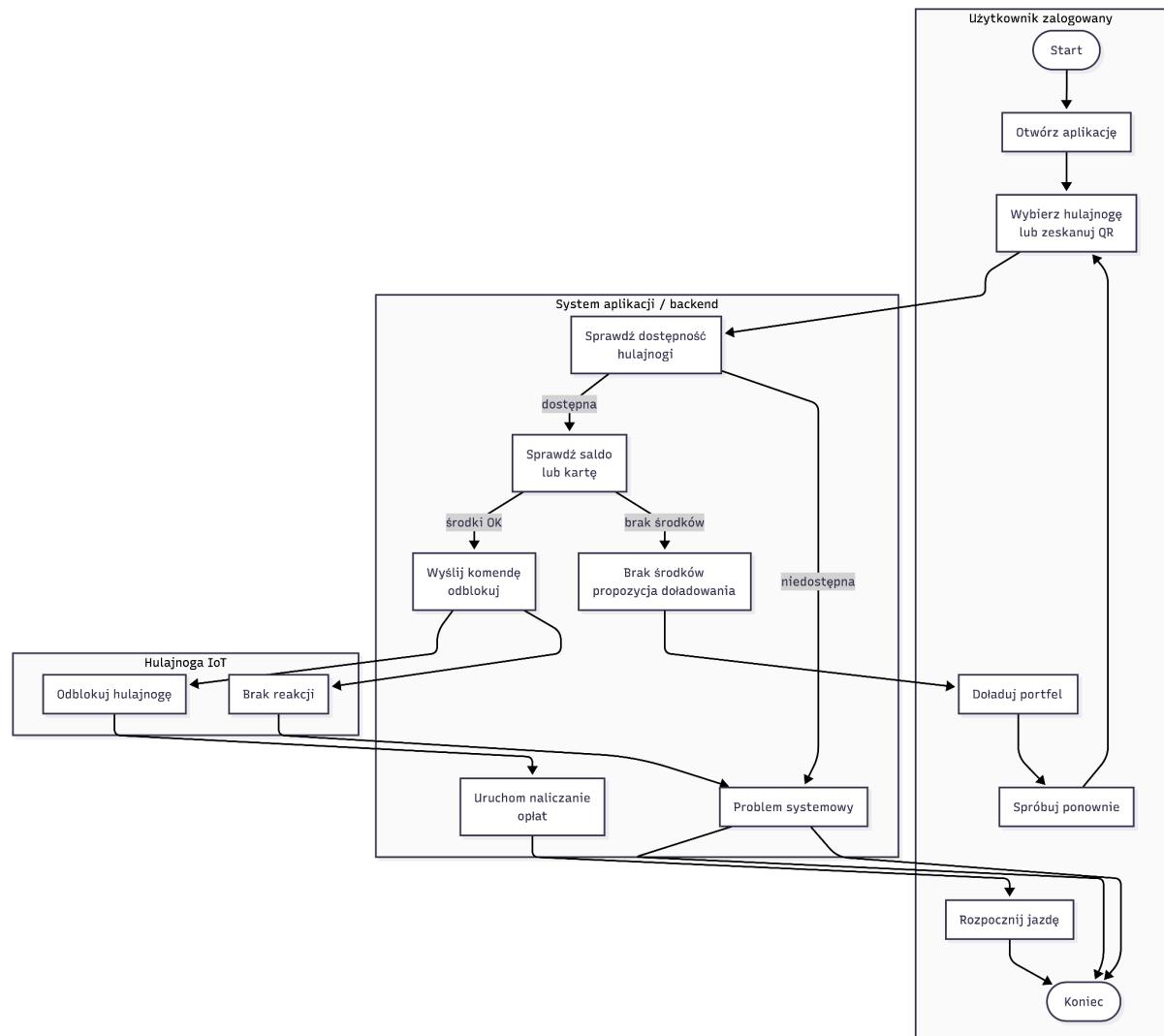


Diagram aktywności 3: S6. Zakończenie przejazdu

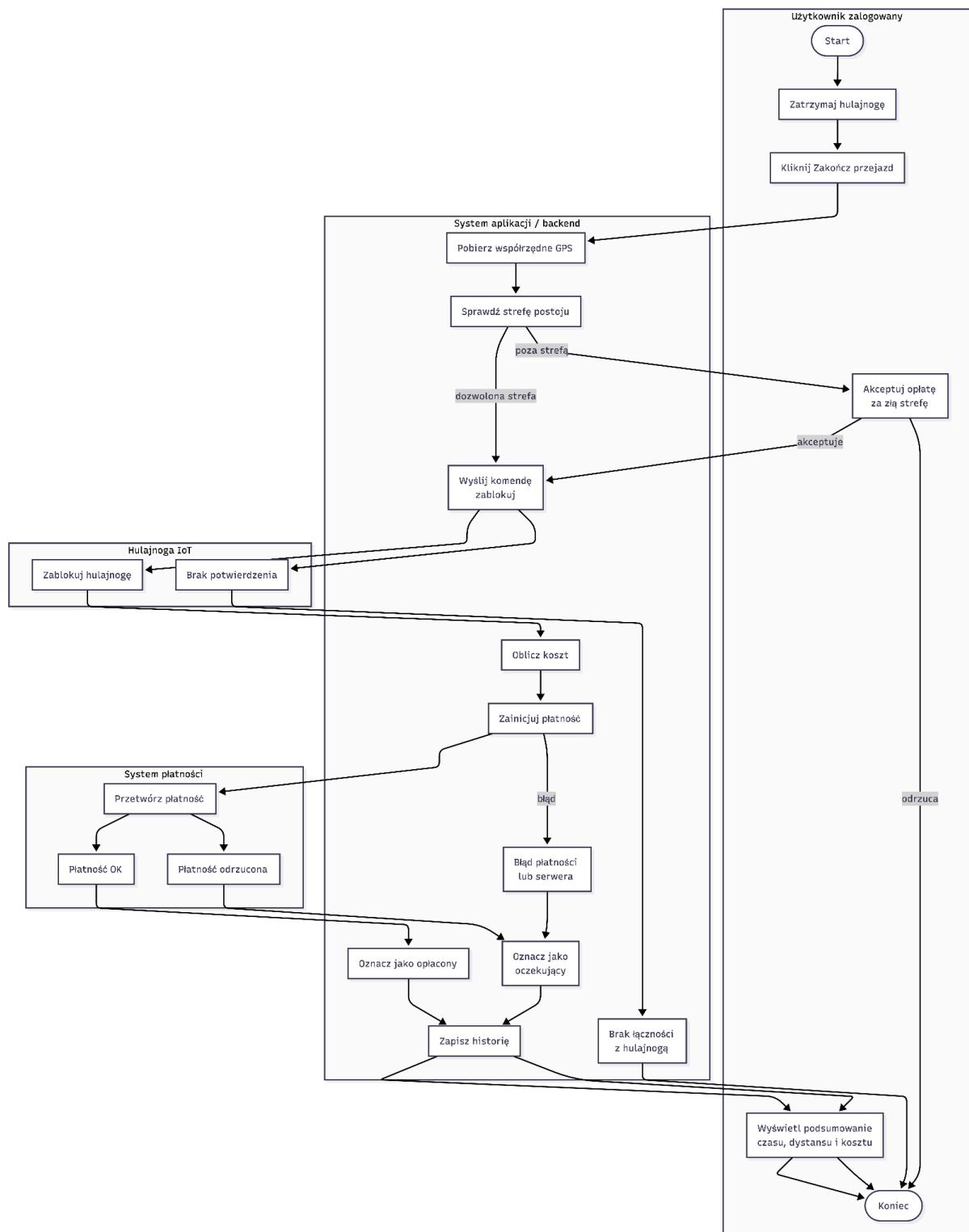
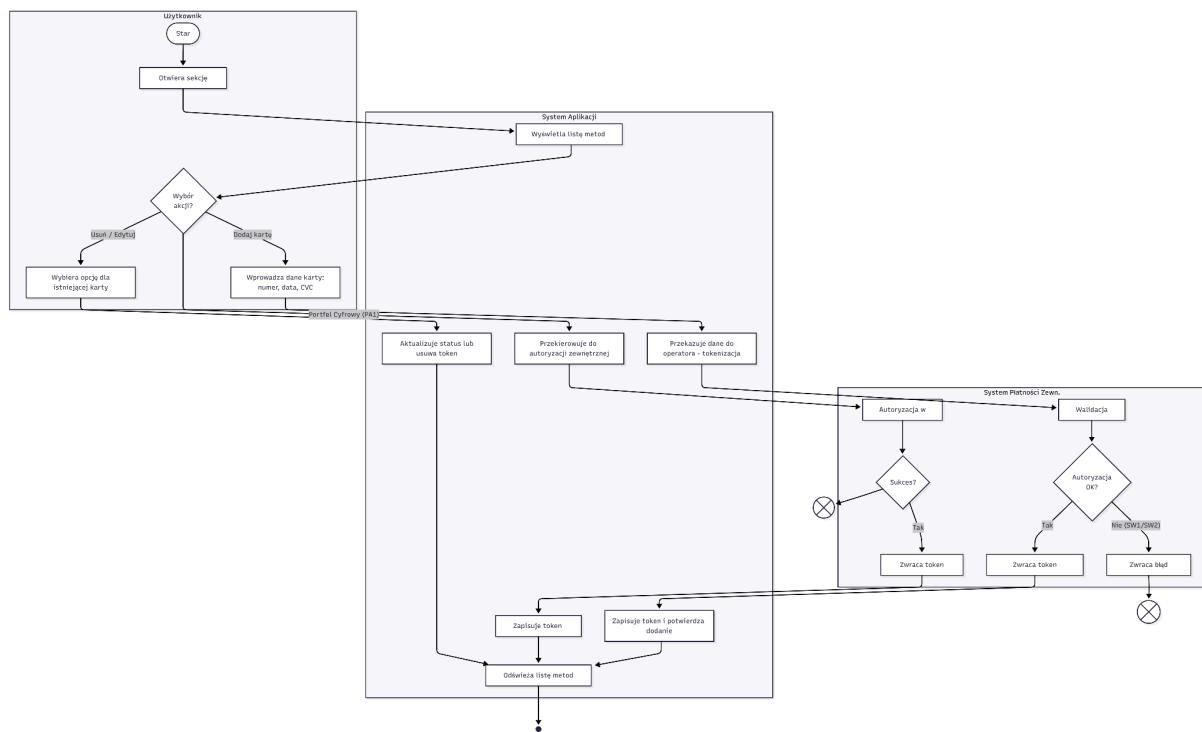


Diagram aktywności 4: S7. Zarządzanie metodami płatności



9. Diagramy rozlokowania.