|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Państwowa Wyższa Szkoła Informatyki i Przedsiębiorczości**  **w Łomży**  **Wydział Informatyki i Nauk O Żywności** | | | | | |
| **Kierunek:**  Informatyka | **Przedmiot:**  Projekt zespołowy II | **Rok:**  III | **Semestr:**    6 | **Prowadzący:**  Janusz Rafałko | **Data:** |
| **Skład zespołu:**  Sebastian Brodziak  Krzysztof Jaśkowski  Sebastian Kucharczyk | | | | | |

**Parking Strzeżony**

**Projekt Systemu Zarządzania**

**Parkingiem Strzeżonym „Carceres”**

# Wstęp

# Ogólna charakterystyka

System zarządzania parkingiem strzeżonym „Carceres” ułatwiający pracę stróża parkingowego oraz obsługę klienta. Zawiera panel logowania, podgląd mapy parkingu, zakładanie użytkowników z podziałem na role, możliwość rejestracji pojazdów, możliwość generowania paragonów/faktur, obsługę płatności oraz cennik. Wszystkie wymienione funkcje dostępne są z poziomu zarówno wersji web jak i desktop. Wersja mobile przeznaczona jest dla klienta końcowego i służy do monitorowania stanu rezerwacji, daje możliwość wykupu nowego miejsca. Cechą charakterystyczną „Carceres” jest jego lekkość oraz uniwersalność – uruchomić go można niezależnie od posiadanej platformy sprzętowej jak i posiadanego systemu operacyjnego.

# Przyczyny wyboru tematu

Temat okazał się ciekawym wyzwaniem dzięki hobbystycznemu podejściu do tematu oprogramowania do zarządzania flota samochodów. Z racji zbliżonej tematyki nasza grupa postanowiła spróbować stworzyć własną wersję. Dodatkowym założeniem było stworzenie systemu prostego w wdrożeniu jak i obsłudze, skierowanego głównie do mniejszych parkingów strzeżonych, gdzie nie jest wymagane posiadanie „ciężkiego” i zaawansowanego systemu z rozwiązań konkurencyjnych.

# Cel projektu

Celem projektu jest stworzenie systemu zarządzania parkingiem strzeżonym, ułatwiającego pracownikom parkingu codzienną pracę oraz nie używającego sporych zasobów sprzętowych. Kluczowa tu ma być mobilność systemu, umożliwiająca uruchomienie go bez względu na zastosowany sprzęt czy system operacyjny.

# Przegląd istniejących rozwiązań

Na rynku istnieją rozwiązania tego typu. Charakteryzują się zróżnicowaną złożonością, możliwością konfiguracji, współpracą z peryferiami typu bramki, wpłatomaty, pętle indukcyjne – rozwiązania zaawansowane technologicznie i wymagające wyższych konfiguracji sprzętowych, szczególnie od strony backendowej.

PROPARK – <https://www.propark.pl/parkingi/parkingi-bezobslugowe-automatyczne/>

TAAB – <https://taab.pl/systemy-parkingowe-taab/>

DEJW – <https://www.dejw.eu/systemy-parkingowe/>

# Zakres projektu

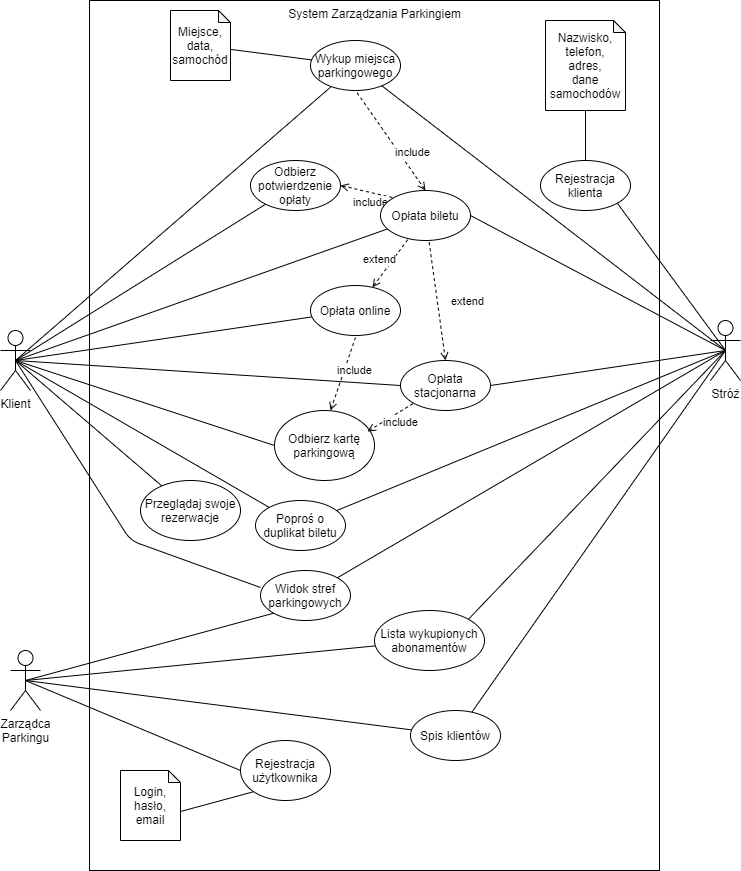
# Opis biznesowy

Carceres to oprogramowanie do zarządzania parkingiem strzeżonym, stworzone specjalnie dla małych parkingów prowadzonych przez mikro i małych przedsiębiorców, nie posiadających dużego budżetu na zakup i obsługę oprogramowania. Uniwersalność, prostota oraz lekkość – to cechy wyróżniające go na tle konkurencyjnych rozwiązań.

# Wymagania funkcjonalne i niefunkcjonalne

|  |  |
| --- | --- |
| **WYMAGANIA FUNKCJONALNE** | **WYMAGANIA NIEFUNKCJONALNE** |
| tworzenie użytkowników systemu, nadawanie im uprawnień oraz logowanie | prosta, intuicyjna obsługa |
| ewidencja miejsc parkingowych (sprawdzanie ogólnej liczby, liczby miejsc wolnych/zajetych, strefy parkingowe) | Niskie wymagania sprzętowe |
| poglądowa mapa parkingu | Możliwość uruchomienia na każdym komputerze |
| rejestracja i ewidencja klientów | Możliwość uruchomienia na smartfonie klasy niższej i w górę |
| rejestracja i ewidencja pojazdów korzystających z parkingu |  |
| generowanie biletów/kart parkingowych (wjazd/wyjazd) |  |
| możliwość rejestracji online oraz płatności online |  |
| duplikaty biletów/kart |  |
| generowanie rachunków |  |
| kontrola płatności |  |
| przypominanie o końcu abonamentu mail/sms |  |
| generowanie raportów okresowych |  |
| generowanie grafików (tworzenie, ewidencja pracowników, urlopy, drukowanie) |  |

# Przypadki użycia



**Aktor – Klient:**

Jestem klientem. Chciałbym móc wykupić miejsce na parkingu online lub na miejscu, z możliwością wcześniejszego wybrania go na mapce. Chciałbym mieć możliwość wyboru czy zapłacę online czy gotówką na miejscu, a także na jaki czas chciałbym miejsce: godzinowo, 1 dzień, kilka dni, tydzień, miesiąc, rok. Gdy będzie kończył mi się abonament chciałbym dostać powiadomienie. Po wykupieniu online chciałbym dostać potwierdzenie zakupu oraz bilet parkingowy w formie PDF. Na miejscu potwierdzenie chciałbym otrzymać wydrukowane. Gdybym zgubił bilet to chciałbym móc poprosić o duplikat.

**Aktor – Stróż:**

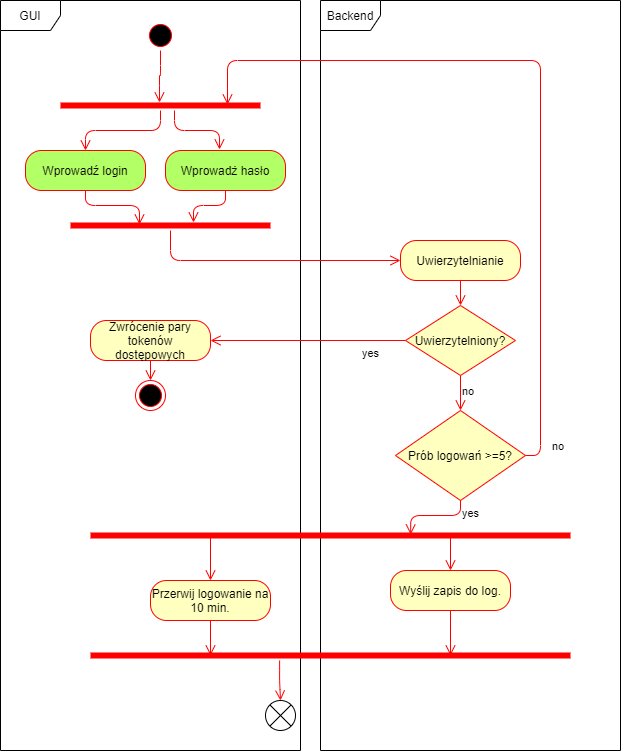
Jestem stróżem. Jako stróż muszę widzieć ile miejsc mam w całości, ile zajętych i ile wolnych. Muszę widzieć które miejsce należy do kogo. Nowy klient musi zostać zarejestrowany, zarejestrowany samochód oraz przypisane miejsce parkingowe i czas abonamentu. Po zarejestrowaniu klient otrzymuje potwierdzenie zakupu i bilet uprawniający do wjazdu i wyjazdu. W momencie wjazdu powinienem widzieć czy klient ma opłacony abonament. Gdy klient zgubi bilet, mogę mu wygenerować duplikat.

**Aktor – Zarządca parkingu:**

Jestem zarządcą parkingu. Chciałbym mieć wgląd na strefy parkingowe oraz widzieć ile miejsc mam zajętych a ile dostępnych. Chciałbym mieć możliwość wglądu do spisu klientów oraz samochodów, wglądu do wykupionych abonamentów i wiedzieć czy każdy zapłacił na czas. Chciałbym też mieć możliwość sporządzenia raportów: dziennych, tygodniowych, miesięcznych, kwartalnych, rocznych, zarówno jeżeli chodzi o statystykę miejsc jak i kasowe oraz pojazdów.

### Diagramy czynności

Diagram aktywności – Logowanie



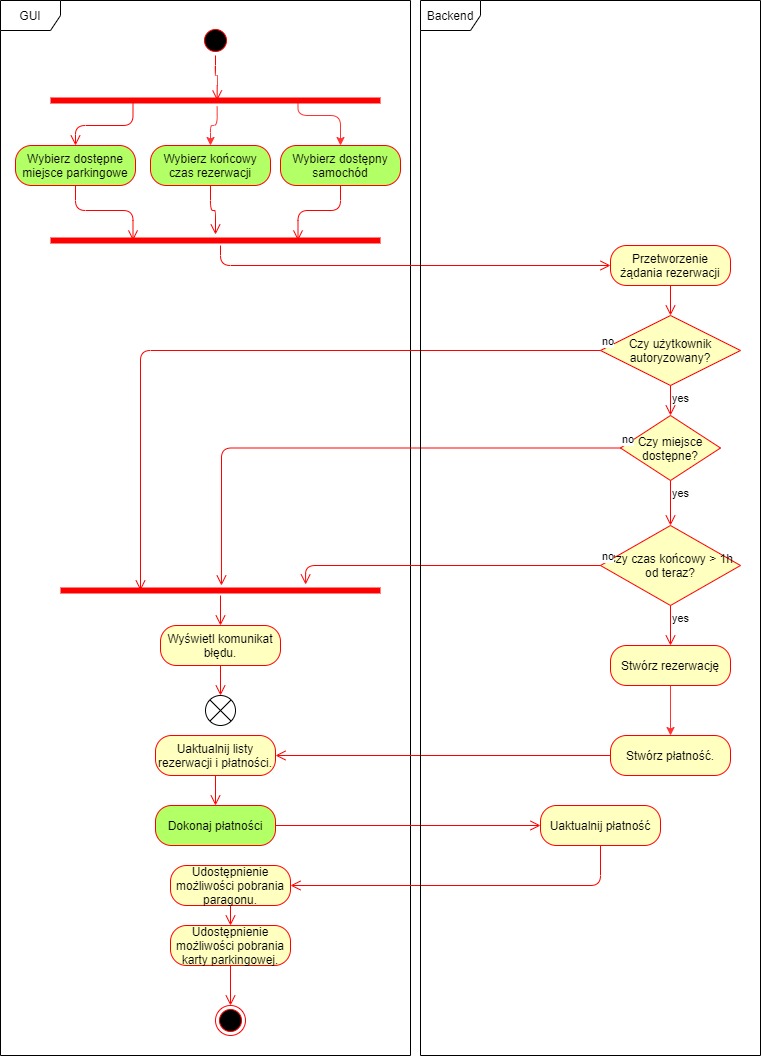
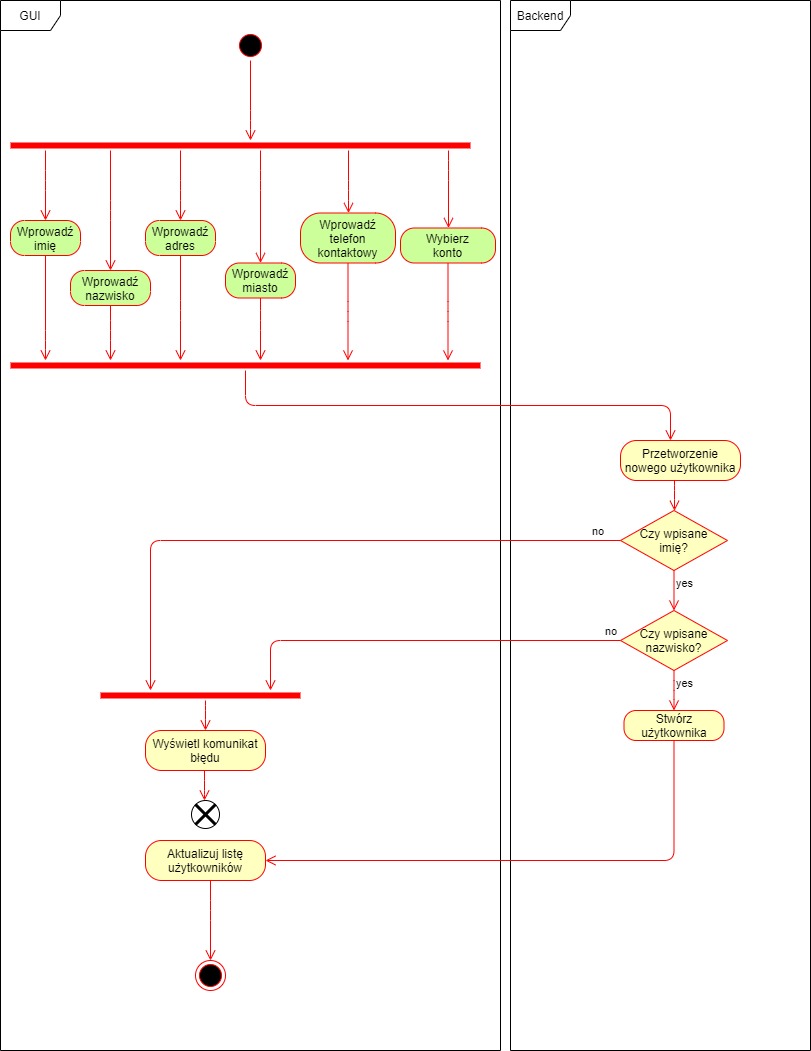
Diagram aktywności – Rezerwacja miejsca parkingowego

Diagram aktywności – Dodanie nowego klienta



### Diagramy interakcji

Diagram interakcji – Logowanie

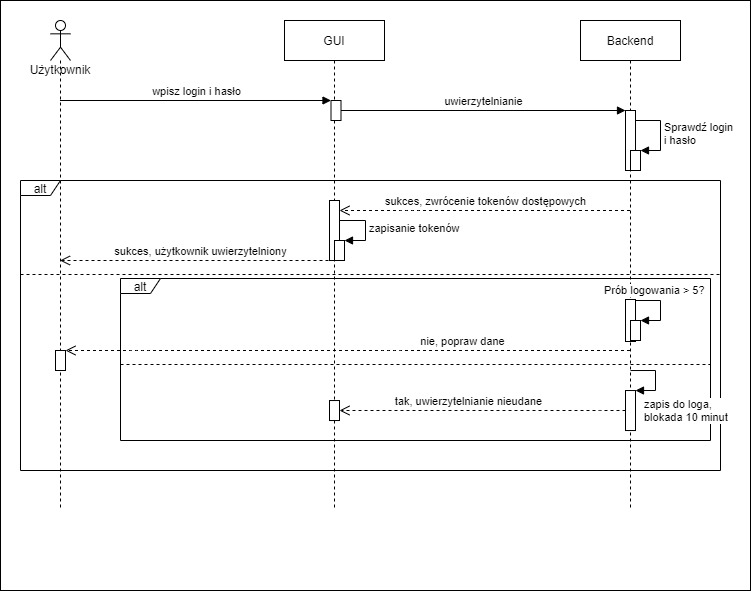


Diagram interakcji – Rezerwacja miejsca parkingowego

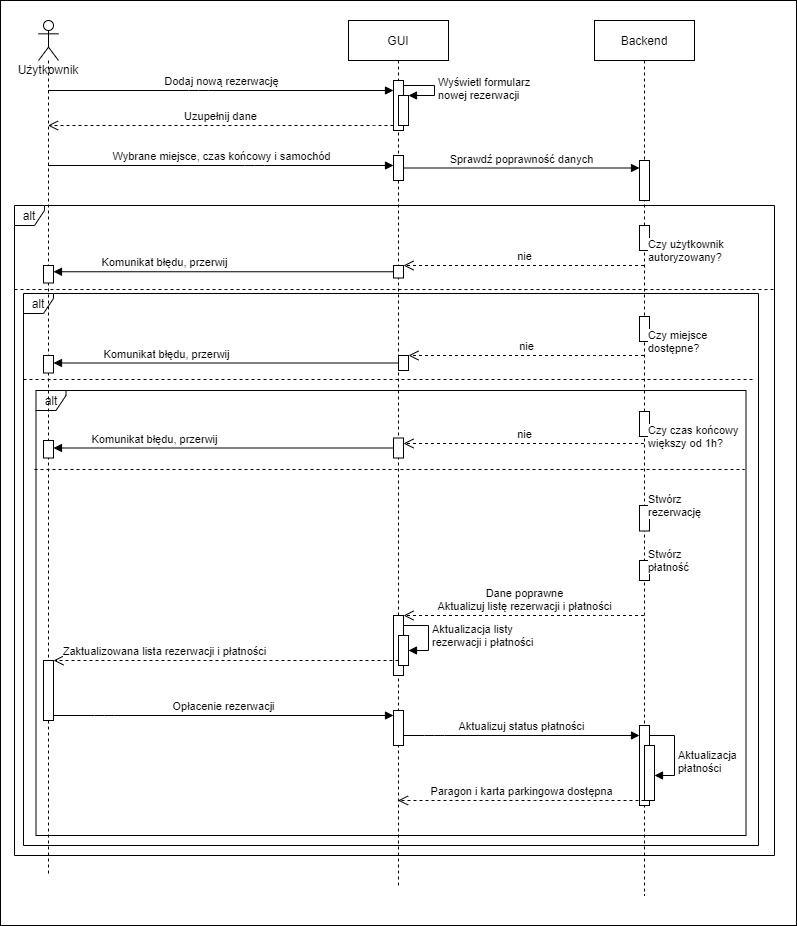
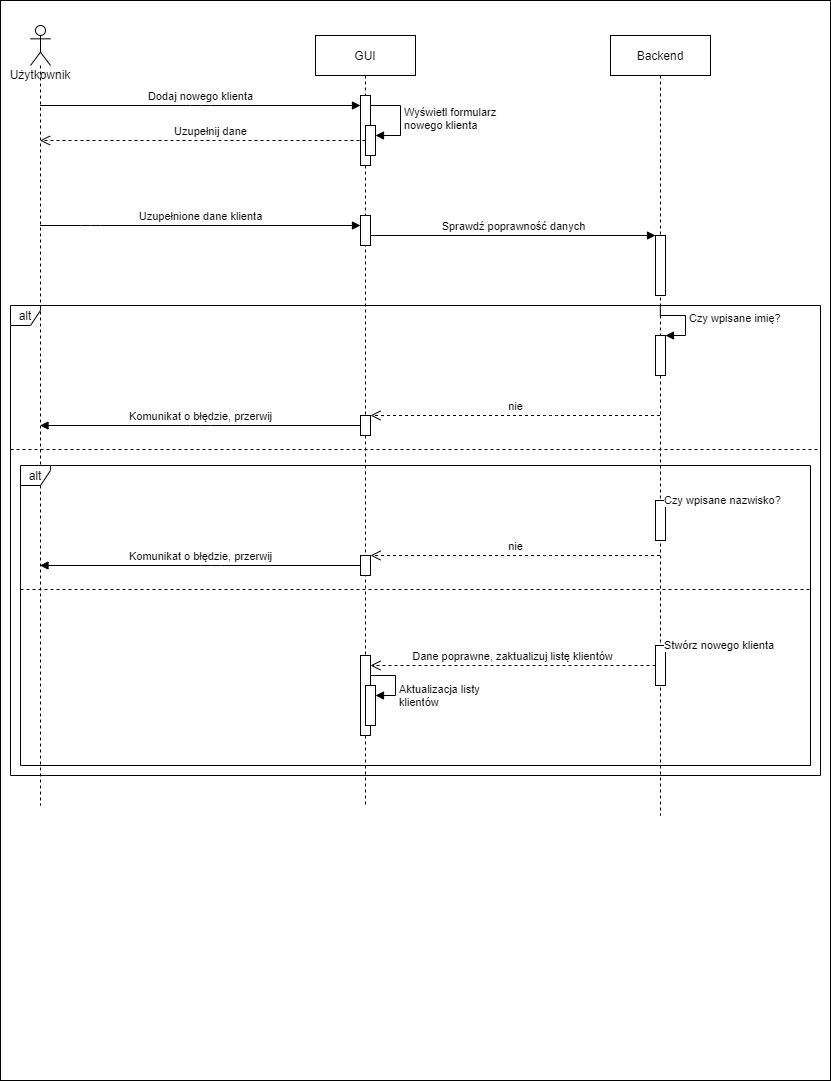


Diagram interakcji – Dodanie nowego klienta



# Członkowie zespołu i harmonogram

Odpowiedzialność poszczególnych członków zespołu:

Sebastian Brodziak – Project Manager, GUI

Krzysztof Jaśkowski – projekt dokumentacji, mobile, testy

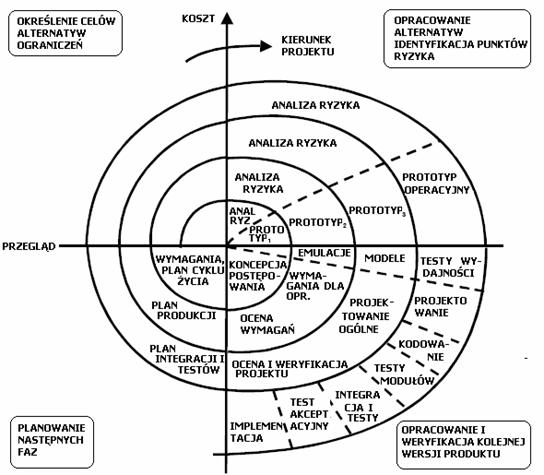
Sebastian Kucharczyk – desktop, mobile

Przewidywany harmonogram prac:

* 31.03 – mockupy GUI
* 10.04 – testy, TDD
* 17.04 – działające GUI
* 24.04 – wersje 1.0 mobile/desktop

# Metodyka

# Metodyka pracy

Model spiralny – model opracowany w 1988 roku. Każde okrążenie spirali reprezentuje planowanie i wytworzenie pewnego elementu produktu projektu. Cały projekt składa się z szeregu obrotów po rozkręcającej się spirali.  


Rys. <https://www.cri.agh.edu.pl/uczelnia/tad/PSI6/wyklady_html/wyklad12_pliki/image008.jpg>

# Metody modelowania

# Środki implementacji

Do tworzenia oprogramowania zastosowano następujące technologie:

Frontend wersji web:

* Node.js
* Vue
* Bootstrap

Backend:

* Python + framework Flask
* MariaDB do obsługi bazy danych

Docker do łatwej implementacji na różnych maszynach

Wersja desktop:

* Electron

Wersja mobile:

* Nativescript
* Node.js

# Model systemu

# Architektura

# Opis modułów

# Model systemu

# Struktury danych i sterowania

# Testy

# Użytkowanie

# Instrukcja wdrożeniowa

# Instrukcja użytkowania

# Podsumowanie

# Cele zrealizowane i niezrealizowane

# Problemy podczas tworzenia

# Kierunki rozbudowy

# Wnioski

# Literatura

# Słownik

**Klient –** osoba korzystająca z parkingu,

**Strefa parkingowa –** teren wykorzystany i podzielony na miejsca parkingowe,

**Miejsce parkingowe –** obszar strefy parkingowej na którym może znajdować się pojazd klienta

**Abonament –** czas przez który klient może korzystać w parkingu,

**Karta parkingowa –** dokument umożliwiający wjazd oraz wyjazd z parkingu, wydawany klientowi po dokonaniu rezerwacji i jej opłaceniu,

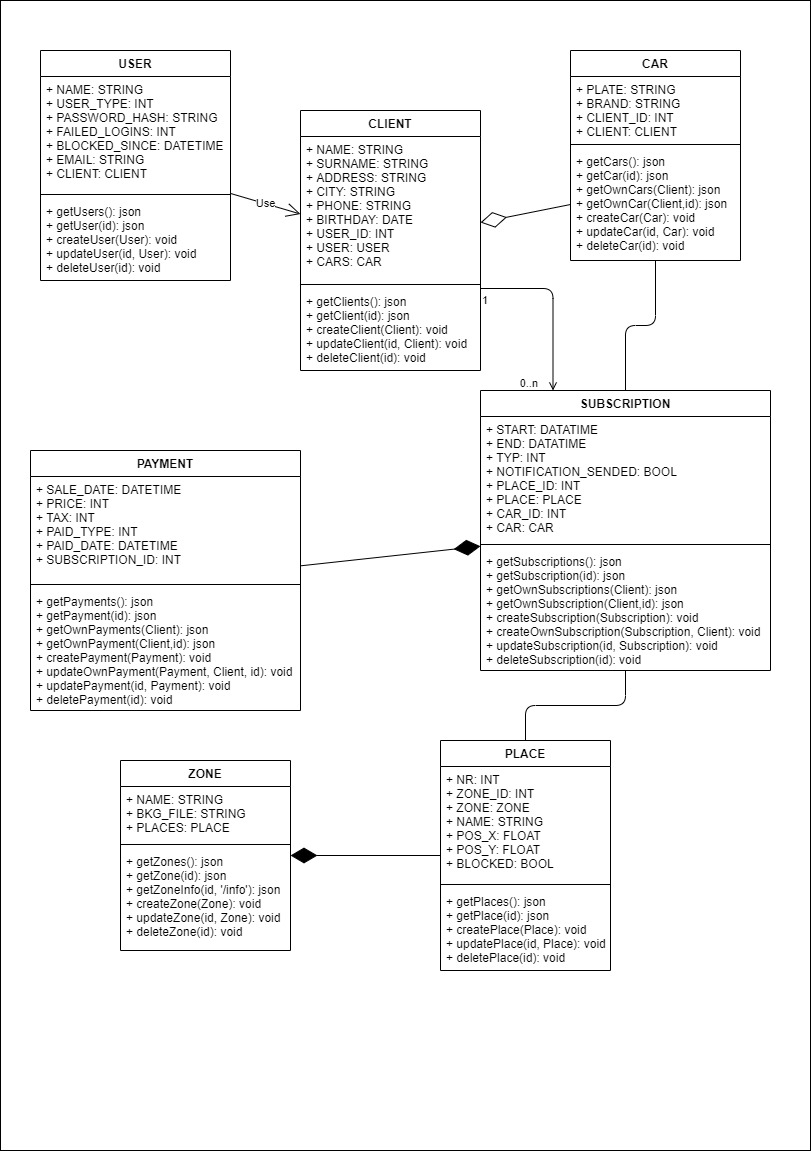
**Stróż –** pracownik parkingu obsługujący system,

**Zarządca parkingu –** właściciel parkingu

# Perspektywa przypadków użycia

# Perspektywa projektowa

### Diagram klas



### Diagramy stanów

Diagram stanu – Autoryzacja użytkownika

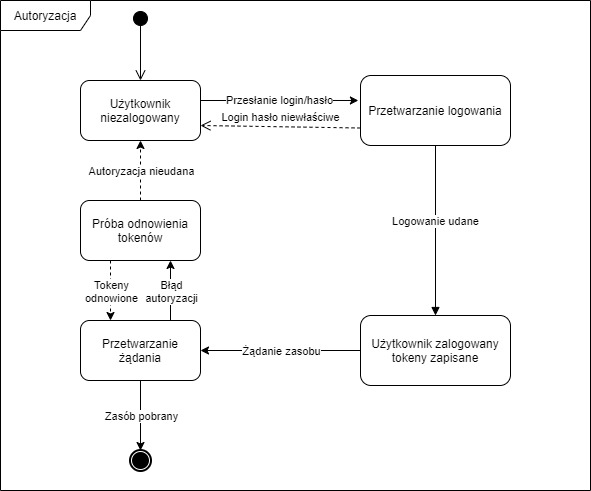
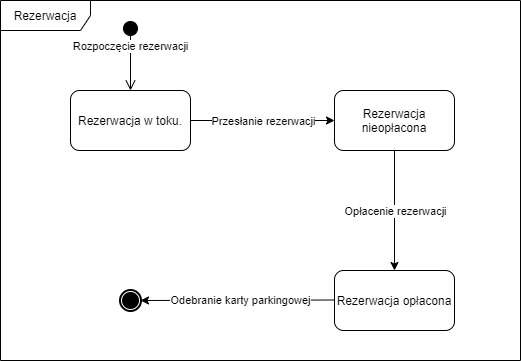


Diagram stanu – Rezerwacja miejsca parkingowego



# Baza danych

Schemat bazy danych

# 

# Diagramy wdrożenia

Diagram wdrożenia z podziałem na wiele maszyn

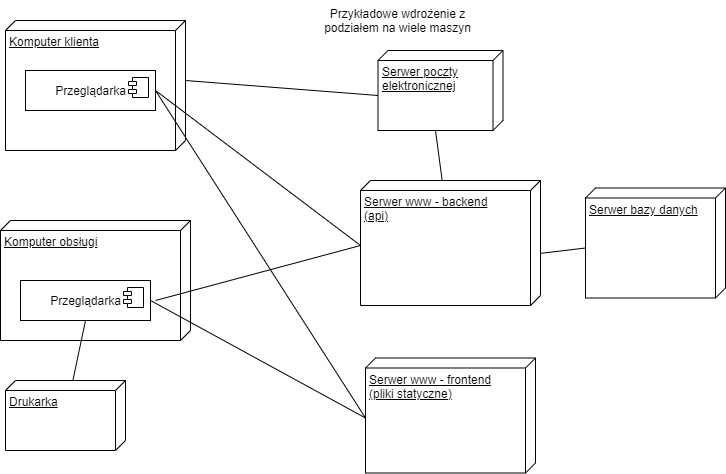


Diagram wdrożenia na jednej maszynie

