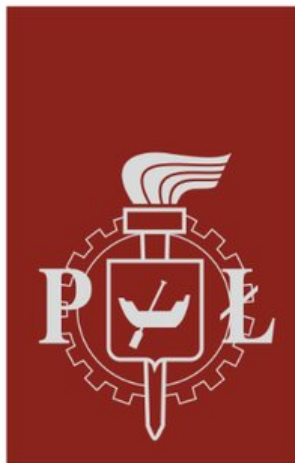


# Politechnika Łódzka

Wydział Elektrotechniki Elektroniki Informatyki i Automatyki



sem, zimowy, r ak. 2024/2025

## **Sprawozdanie z projektu BigData „Predykcja cen samochodów używanych”**



Mateusz Grzybek 240678

Kamil Młynarczyk 240757

15 grudnia 2024

# Spis treści

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Wstęp</b>                                   | <b>2</b> |
| 1.1      | Założenia projektowe . . . . .                 | 2        |
| 1.2      | Komponenty . . . . .                           | 2        |
| 1.3      | Wykorzystane technologie . . . . .             | 2        |
| <b>2</b> | <b>Aplikacja kliencka</b>                      | <b>4</b> |
| 2.1      | Opis . . . . .                                 | 4        |
| 2.2      | Widoki aplikacji . . . . .                     | 5        |
| 2.2.1    | Strona . . . . .                               | 5        |
| 2.2.2    | Okno z ceną . . . . .                          | 6        |
| 2.2.3    | Okno z błędem . . . . .                        | 6        |
| <b>3</b> | <b>Komponent pośredniczący i komunikacyjny</b> | <b>7</b> |
| 3.1      | Komponent pośredniczący . . . . .              | 7        |
| 3.1.1    | Opis . . . . .                                 | 7        |
| 3.2      | Komponent komunikacyjny . . . . .              | 7        |
| 3.2.1    | Opis . . . . .                                 | 7        |
| <b>4</b> | <b>Diagramy</b>                                | <b>8</b> |
| 4.1      | Diagram przypadków użycia . . . . .            | 8        |
| 4.2      | Diagram sekwencji zdarzeń . . . . .            | 8        |

# Rozdział 1

## Wstęp

### 1.1 Założenia projektowe

Celem projektu jest zaimplementowanie aplikacji webowej pozwalającej użytkownikom na predykcję ceny używanego samochodu na podstawie dostarczonego przez niego zestawu cech. Tematyka projektu daje możliwość wykorzystania różnorodnych technologii z dziedziny uczenia maszynowego, rozwoju aplikacji webowych, komunikacji pomiędzy serwisami, architektury oprogramowania oraz bierania i przetwarzania danych. W celu zrealizowania przewidywanych funkcjonalności, aplikacja została podzielona na cztery komponenty, każdy z nich odpowiedzialny za realizację innego aspektu aplikacji.

### 1.2 Komponenty

- Aplikacja kliencka — Interfejs graficzny użytkownika.
- Pośrednik — Komponent pośredniczący w komunikacji pomiędzy aplikacją kliencką i serwisem predykcyjnym
- Komponent komunikacyjny — Komponent zawierający szyny danych, które są wykorzystywane do dostarczania i odbierania informacji od serwisu predykcyjnego
- Serwis predykcyjny — Komponent dokonujący predykcji na podstawie dostarczonych danych, z wykorzystaniem nauczonego modelu.

### 1.3 Wykorzystane technologie

- Java — Obiektowy język programowania.
- SpringBoot — Framework dla języka Java nastawiony na wytwarzanie aplikacji webowych i mikroservisów
- Gradle — Narzędzie do automatyzacji budowania projektów.
- React — Framework JavaScript do tworzenia interfejsów użytkownika w oparciu o komponenty.
- Docker — Narzędzie do tworzenia, uruchamiania i zarządzania aplikacjami w izolowanych środowiskach zwanych kontnerami.

- Docker Compose — Narzędzie usprawniające zarządzanie wieloma kontenerami jednocześnie.
- Python — Język skryptowy.
- Apache Spark — Framework do sprawnego przetwarzania zbiorów danych w pamięci.
- Apache SparkML — Moduł Apache Spark przeznaczony do uczenia maszynowego.
- Apache Kafka — Platforma przetwarzania danych w czasie rzeczywistym.
- Apache Zookeeper — Usługa koordynacyjna systemów rozproszonych.

# Rozdział 2

## Aplikacja kliencka

### 2.1 Opis

Aplikacja kliencka stanowi pojedynczą stronę dostępną za pośrednictwem przeglądarki, udostępnianą pod adresem **localhost**<sup>1</sup>, na porcie **9091**. Strona zawiera informacje związane z aplikacją oraz pola do wprowadzania wartości, na podstawie których następnie dokonywana jest predykcja ceny samochodu. Aplikacja łączy się z komponentem middleware za pośrednictwem protokołu **HTTP**<sup>2</sup> w architekturze **REST**<sup>3</sup>.

---

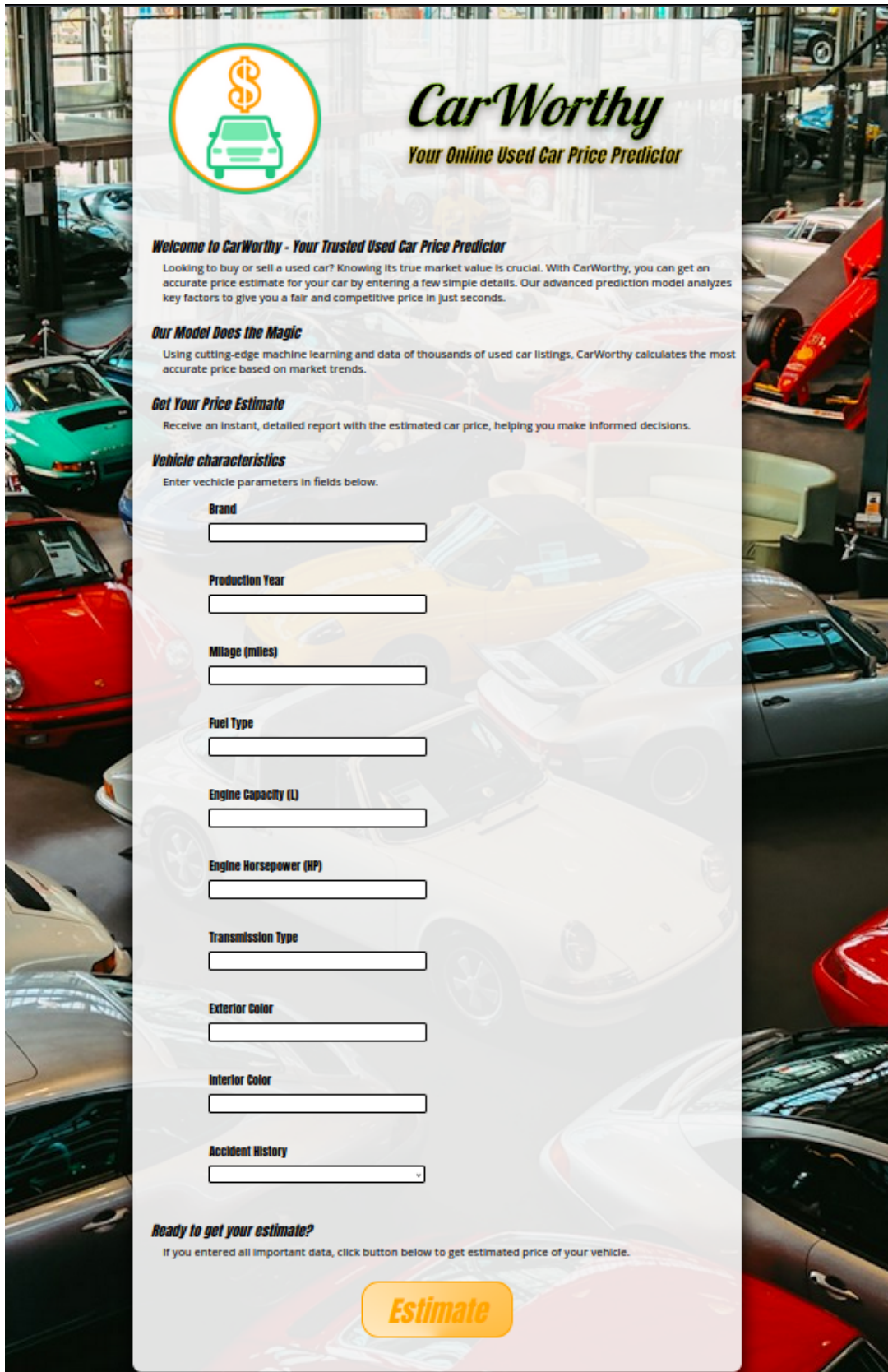
<sup>1</sup>loopback address — adres pętli zwrotnej, który jest wykorzystywany do komunikacji urządzenia z samym sobą.

<sup>2</sup>HyperText Transfer Protocol — protokół komunikacyjny używany do przesyłania danych w sieci.

<sup>3</sup>Representational State Transfer — architektura komunikacji oparta o protokół HTTP definiujący sposoby identyfikacji i manipulacji zasobami za pomocą zapytań HTTP.

## 2.2 Widoki aplikacji

### 2.2.1 Strona



The screenshot displays the CarWorthy website, which is an online used car price predictor. The page features a light gray background with a faint image of a car dealership. At the top left, there is a logo consisting of a green circle with a white car icon and a dollar sign above it. To the right of the logo, the text "CarWorthy" is written in a large, bold, black font, and "Your Online Used Car Price Predictor" is written in a smaller, italicized, black font below it.

Below the header, there is a section titled "Welcome to CarWorthy - Your Trusted Used Car Price Predictor" in bold. The text below this title reads: "Looking to buy or sell a used car? Knowing its true market value is crucial. With CarWorthy, you can get an accurate price estimate for your car by entering a few simple details. Our advanced prediction model analyzes key factors to give you a fair and competitive price in just seconds."

Next is a section titled "Our Model Does the Magic" in bold. The text below this title reads: "Using cutting-edge machine learning and data of thousands of used car listings, CarWorthy calculates the most accurate price based on market trends."

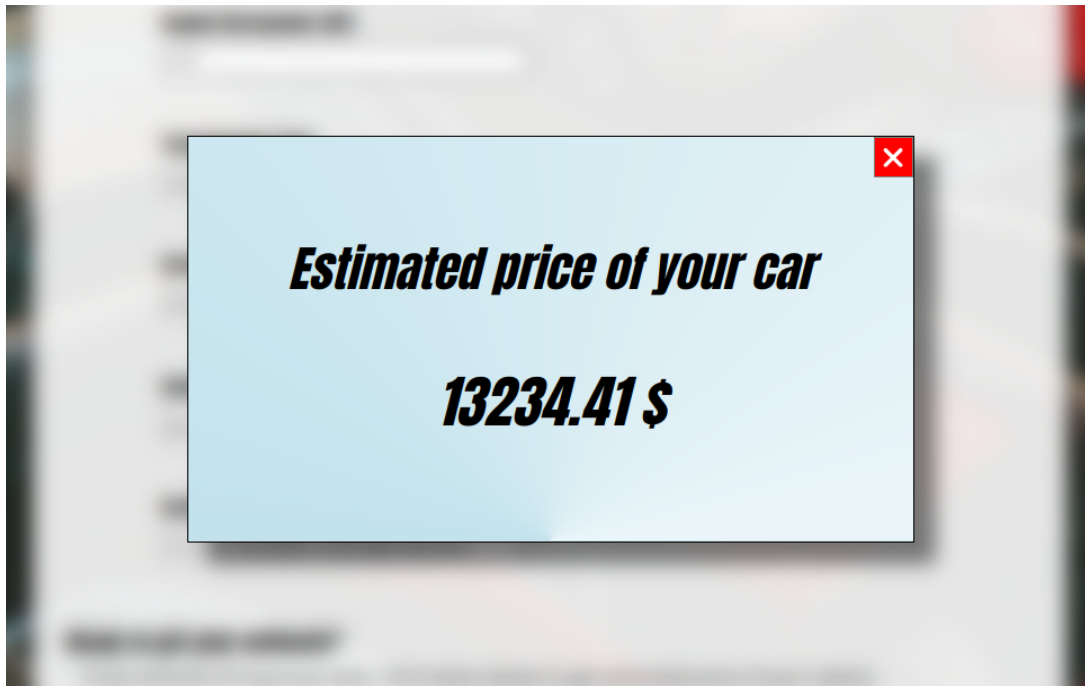
Below that is a section titled "Get Your Price Estimate" in bold. The text below this title reads: "Receive an instant, detailed report with the estimated car price, helping you make informed decisions."

The main section is titled "Vehicle characteristics" in bold. Below this title, it says "Enter vehicle parameters in fields below." There are ten input fields, each with a label to its left: "Brand", "Production Year", "Mileage (miles)", "Fuel Type", "Engine Capacity (L)", "Engine Horsepower (HP)", "Transmission Type", "Exterior Color", "Interior Color", and "Accident History". Each field is a simple white rectangle with a thin black border.

At the bottom of the form, there is a section titled "Ready to get your estimate?" in bold. Below this title, it says "If you entered all important data, click button below to get estimated price of your vehicle." There is a large, orange, rounded rectangular button with the word "Estimate" written in white, bold, italicized font.

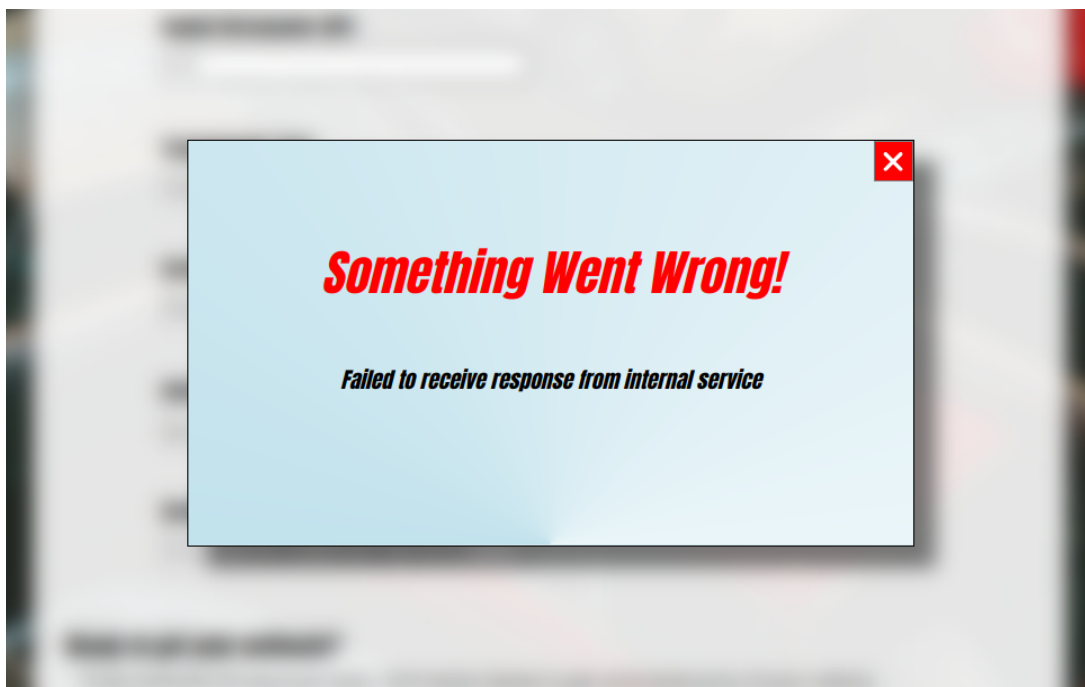
Rysunek 2.1: Widok strony

### 2.2.2 Okno z ceną



Rysunek 2.2: Widok okna z ceną

### 2.2.3 Okno z błędem



Rysunek 2.3: Widok okna z błędem

# Rozdział 3

## Komponent pośredniczący i komunikacyjny

### 3.1 Komponent pośredniczący

#### 3.1.1 Opis

Komponent pośredniczący pełni rolę pośrednika pomiędzy aplikacją kliencką i serwisem predykcyjnym. Otrzymywane od **frontendu**<sup>1</sup> dane w formie **JSON**<sup>2</sup> są w tym komponencie przetwarzane na wiadomości w formacie odpowiadającym wejściu modelu, z uwzględnieniem procesu **kodowania liczbowego**<sup>3</sup> pól. Otrzymane w tym procesie wiadomości zapisywane są na **temat**<sup>4</sup> wejściowy Kafki. Pośrednik jest również odpowiedzialny za odczytywanie danych z tematu wyjściowego i przekazywanie uzyskanych z nich informacji do klienta.

### 3.2 Komponent komunikacyjny

#### 3.2.1 Opis

Komponent komunikacyjny odpowiedzialny jest za transport danych pomiędzy komponentem pośredniczącym i serwisem predykcyjnym. Wykorzystuje w tym celu skonteneryzowany **broker**<sup>5</sup> wiadomości Apache Kafka wraz z dwoma tematami input oraz output, wykorzystywanych odpowiednio do gromadzenia danych odczytywanych przez serwis predykcyjny i gromadzenia danych odczytywanych przez pośrednika. Do zarządzania brokerem wykorzystywany jest Apache Zookeeper.

---

<sup>1</sup>Część aplikacji, z którą użytkownik wchodzi w bezpośrednią interakcję, w tym wszystko co widzi oraz elementy wizualne i interaktywne.

<sup>2</sup>JavaScript Object Notation — format danych zapewniający kompaktowe rozmiary i jest czytelny dla ludzi i maszyn.

<sup>3</sup>Technika zamiany wartości danych tekstowych na wartości liczbowe, poprzez przypisanie unikalnej liczby każdej unikalnej wartości tekstowej.

<sup>4</sup>Podstawowy komponent Apache Kafka służący do kategoryzacji napływających wiadomości.

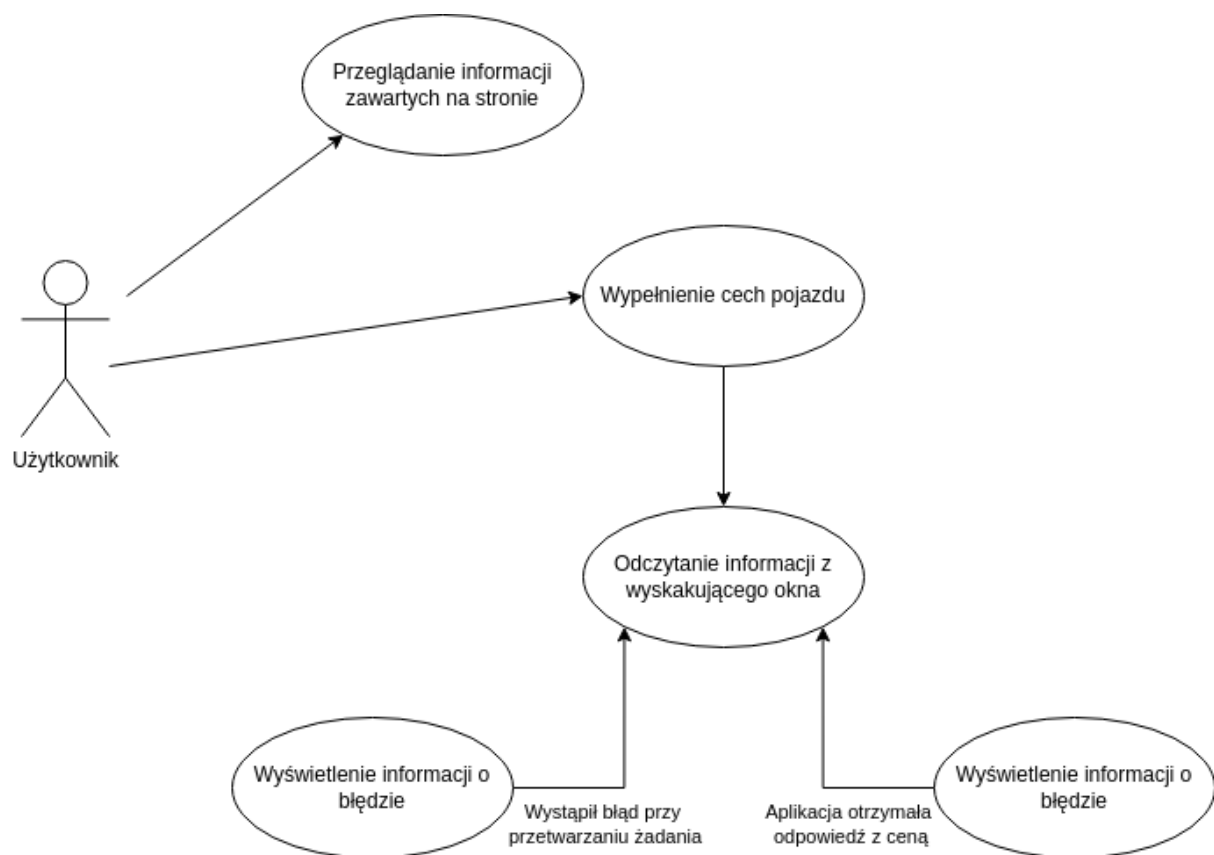
<sup>5</sup>Serwer Apache Kafka zawierający dane należące do tematów i partycji, na które może być podzielony temat



# Rozdział 4

## Diagramy

### 4.1 Diagram przypadków użycia



Rysunek 4.1: Przebieg interakcji użytkownika z aplikacją

### 4.2 Diagram sekwencji zdarzeń