

Projekt "Gazetka"

BEST Hacking League 2025
kategoria Software

Zespół **Rage Against the Machine**

w składzie:

Juliusz Winnicki
Wojciech Bakun
Mikołaj Roszczyk

G
GAZETKA

B#L

Warszawa, 30 listopada 2025

Spis treści

1	Definicja problemu	2
2	Projekt rozwiązania	2
3	Wartość biznesowa	2
4	Implementacja	3
4.1	Architektura	3
4.2	Baza danych	3
4.3	Backend	4
4.4	Aplikacja mobilna	4
4.5	Aplikacja webowa	4
5	Działanie	4
5.1	Przykładowy scenariusz	4
5.2	Aplikacja mobilna	4
5.3	Aplikacja webowa	6

1 Definicja problemu

Zadanie konkursowe polegało na stworzeniu aplikacji, która pomoże we wprowadzeniu do życia proekologicznych nawyków albo ekologicznych rozwiązań w miastach lub gospodarstwach domowych.

Na warsztat wzięto problem marnowania jedzenia, a co za tym idzie konieczności wytwarzania większych ilości porcji. Każdego roku w samej Polsce marnuje się około 4.8 mln ton jedzenia (raport z badań projektu PROM z roku 2020). W całej Europie jest to wartość około 58.4 mln ton rocznie (Eurostat, dane z 2021 r.).

Jedną z istotnych konsekwencji dużego marnowania jedzenia jest konieczność dostarczania wciąż nowych, świeżych produktów. Jest to korzystne dla producentów żywności, natomiast powoduje nierówną dystrybucję żywności, a także stanowi istotny problem dla środowiska naturalnego.

Rocznie na całym świecie produkcja żywności generuje 16.2 mld ton CO_2 -eq (szacunki Organizacji Narodów Zjednoczonych do spraw Wyżywienia i Rolnictwa z 2022 roku). Oszacowanie to obejmuje emisję zachodzącą na gospodarstwach rolnych, zmiany użytkowania gruntów oraz procesy przed i po produkcji. Według analizy Hanny Ritchie z 2021 roku (Our World in Data) stanowi to około 25-33% globalnych emisji (w zależności od definicji).

Zmniejszenie wyrzucania żywności powinno doprowadzić do zmniejszenia emisji powodowanej przez jej produkcję i transport.

2 Projekt rozwiązania

Pomysłem zespołu *Rage Against the Machine* na rozwiązanie problemu zdefiniowanego w rozdziale 1 jest stworzenie aplikacji, która umożliwiałaby sklepom spożywczym wystawianie produktów o zbliżającej się dacie przydatności w promocyjnej cenie.

Ważnym elementem systemu jest element gamifikacji. Za każdą transakcję punkty dostaje zarówno sklep, jak i użytkownik. Sklepy są klasyfikowane w rankingu na podstawie punktów. Użytkownicy mogą stale kontrolować pieniądze zaoszczędzone na kupowaniu produktów z aplikacji oraz wymieniać zebrane punkty na nagrody.

3 Wartość biznesowa

Jako interesariuszy bezpośrednich naszego rozwiązania należy wymienić konsumentów oraz sklepy spożywcze. Interesariuszami pośrednimi są producenci żywności i firmy transportowe.

Dla interesariuszy bezpośrednich produkt przynosi widoczną wartość biznesową:

- konsumentom:
 - pozwala kupować produkty taniej, kiedy chcą je spożyć w przeciągu najbliższych dni
 - pozwala wymienić zdobyte punkty na nagrody
- sklepom spożywczym
 - ułatwia sprzedaż produktów, którym zagrożone koniecznością wycofania ze sprzedaży (zysk)
 - pozwala na łatwą reklamę poprzez pozycjonowanie w rankingach - globalnym i lokalnym

Grupy społeczne, które mogą stracić na wprowadzeniu rozwiązania to producenci żywności i firmy transportowe zajmujące się jej transportem.

Widoczną korzyść rozwiązanie przynosi też dla środowiska, na czym korzystają wszyscy ludzie. Rozwiązanie może przynosić inwestorom zyski poprzez:

- wprowadzenie płatnej subskrypcji dla sklepów spożywczych, by brać udział w programie
- płatne pozycjonowanie produktów na listach przez sklepy spożywcze
- niewielki udział aplikacji w wartości transakcji (np. 5 gr. od każdej transakcji)

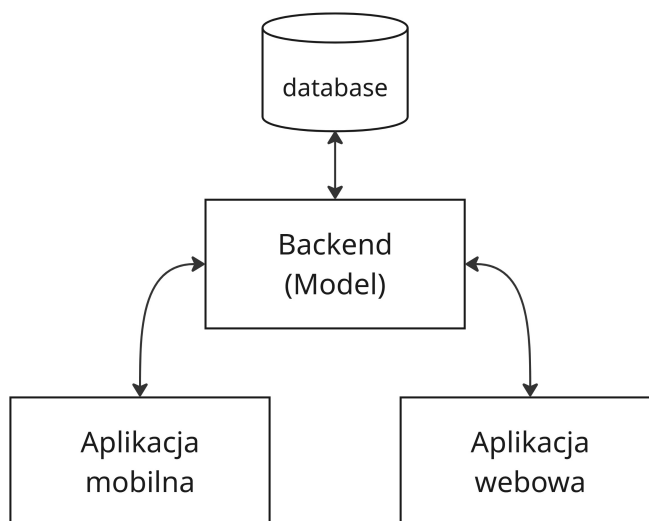
4 Implementacja

Pełny kod wersji POC aplikacji znajduje się na repozytorium:

<https://github.com/Rage-Against-the-Machine/BestHackingLeague25>.

4.1 Architektura

Rozwiązanie zostało przygotowane w architekturze 3TA (three-tier-architecture: presentation, application, data) z dwoma interfejsami prezentacyjnymi. System składa się z czterech komponentów: bazy danych, backendu, aplikacji mobilnej i aplikacji webowej. Wizualizację architektury przedstawiono na rysunku 1.



Rys. 1: Lokalizacja obserwatora

4.2 Baza danych

Uruchomiono dokumentową bazę danych (NoSQL) - MongoDB. Baza ta przechowuje w kolekcjach dokumenty złożone z par {klucz : wartość}. W bazie danych znajdują się 4 kolekcje, każda odpowiadająca innej klasie danych: użytkownicy, sklepy, produkty, kody QR.

Komunikacja z bazą danych zachodzi za pomocą języka Python i biblioteki *pymongo*.

4.3 Backend

Backend został przygotowany w języku Python, korzystając z frameworka *Flask*. Uruchomiono 15 endpointów mających realizować zadania pobierania, dodawania, usuwania, edytowania (CRUD) obiektów posiadanych klas do bazy danych. Zaimplementowano między innymi logikę dodawania punktów sklepom i użytkownikom, generowania QR kodów do identyfikacji użytkownika, przeprowadzania operacji kupowania (usunięcie liczby sztuk produktu, dodanie punktów sklepowi i użytkownikowi), uwierzytelniania kont.

Poza testami funkcjonalnymi, kod został przetestowany z użyciem testów jednostkowych. Zostały one stworzone z użyciem frameworka *pytest*.

4.4 Aplikacja mobilna

Aplikacja mobilna została stworzona dla systemów operacyjnych Android i iOS z wykorzystaniem frameworka *flutter*. Zaimplementowane funkcjonalności opisano i przedstawiono w 5.2.

4.5 Aplikacja webowa

Aplikacja webowa została stworzona z wykorzystaniem języka TypeScript, biblioteki *React* oraz narzędzia *Vite*. Zaimplementowane funkcjonalności opisano i przedstawiono w 5.3.

5 Działanie

5.1 Przykładowy scenariusz

Użytkownik przegląda w aplikacji dostępne przeceny. Potrzebuje mleka, który szybko wykorzysta. Może użyć funkcji filtrowania po kategoriach produktów oraz po odległości od sklepu. Znajduje przecenione mleko o odpowiadającej mu dacie przydatności w pobliskim sklepie.

Przychodzi do rzeczonego sklepu, bierze produkt. Generuje unikalny QR code, który jest skanowany na kasie. Sklep rejestruje zniżkę i wysyła na odpowiedni endpoint informację o sprzedaży produktów. Użytkownikowi i sklepowi dodawane są punkty rankingowe i jest aktualizowana dostępna liczba produktów w aplikacjach.

5.2 Aplikacja mobilna

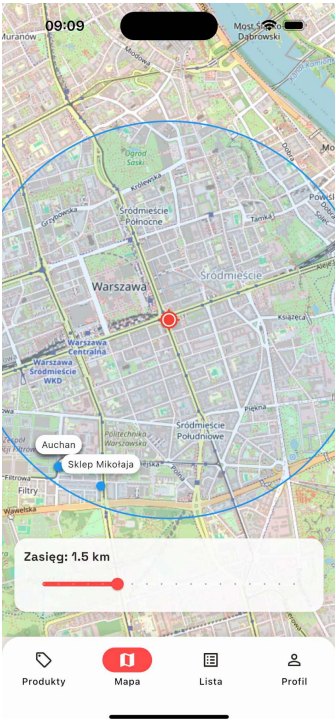
Aplikacja mobilna implementuje widok użytkownika. Umożliwia następujące czynności:

- Przeglądanie gazetki (rys. 2)
- Lokalizowanie sklepów udostępniających oferty na mapie (rys. 3)
- Przeglądanie rankingu sklepów (rys. 7)
- Oglądanie własnego profilu z generowanym kodem QR i liczbą zdobytych punktów (rys. 4)
- Generowanie listy zakupów (rys. 5)

Aplikacja umożliwia logowanie użytkownika (rys. 6)



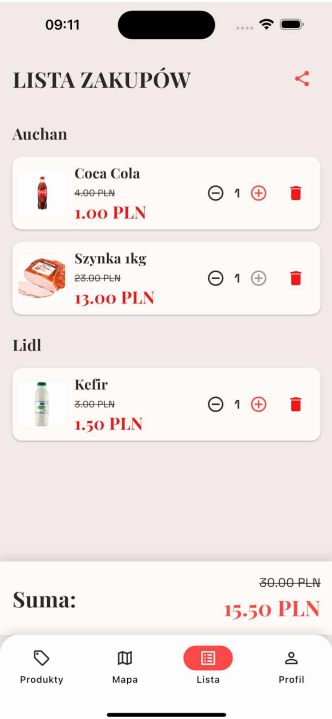
Rys. 2: Widok gazetki (prze-
gląd)



Rys. 3: Widok lokalizatora
okazji



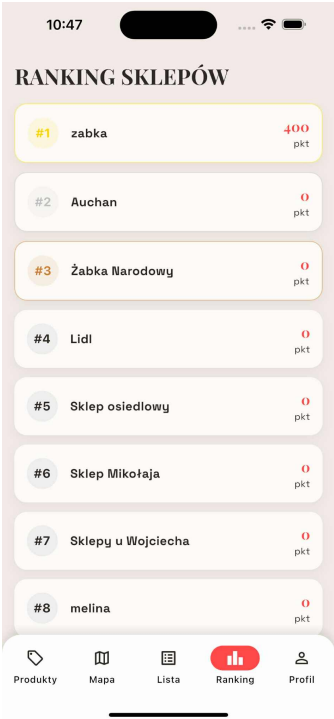
Rys. 4: Widok profilu użyt-
kownika



Rys. 5: Widok listy zakupów



Rys. 6: Widok logowania

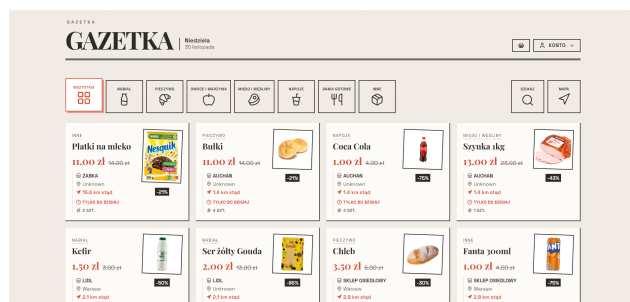


Rys. 7: Widok rankingu

5.3 Aplikacja webowa

Aplikacja webowa posiada widoki zarówno użytkownika, jak i sklepu. Umożliwia następujące czynności:

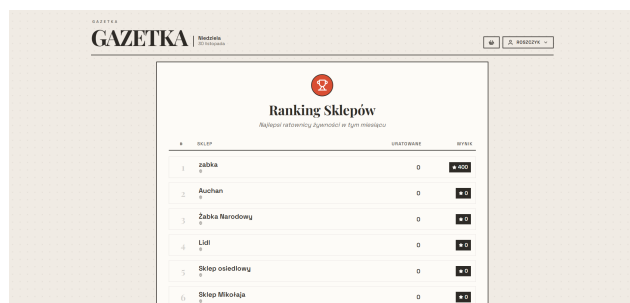
- Przeglądanie gazetki (nie wymaga zalogowania) (rys. 8)
- Lokalizowanie sklepów udostępniających oferty na mapie (rys. 9)
- Przeglądanie rankingu sklepów (nie wymaga zalogowania) (rys. 10)
- Oglądanie własnego profilu z generowanym kodem QR i liczbą zdobytych punktów (rys. 11)
- Dodawanie, usuwanie i edytowanie pozycji wystawionych do gazetki (widok sklepu)
- Generowanie listy zakupów



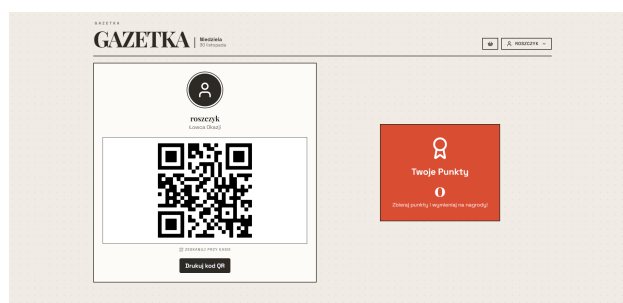
Rys. 8: Widok gazetki (przegląd)



Rys. 9: Widok lokalizatora okazji

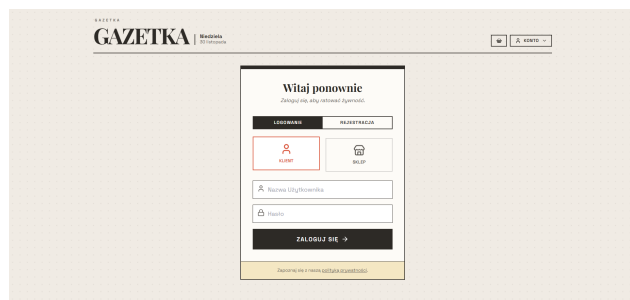


Rys. 10: Widok rankingu sklepów

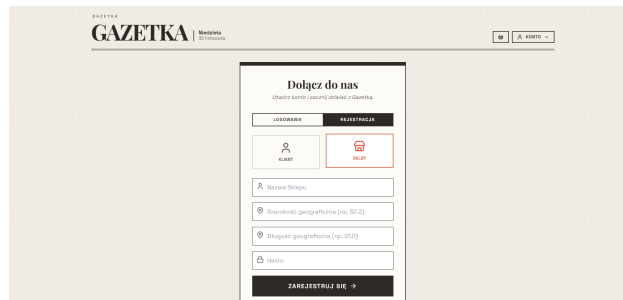


Rys. 11: Widok profilu użytkownika

Aplikacja webowa umożliwia logowanie i rejestrację klientów i sklepów (rys. 12 i 13).



Rys. 12: Widok logowania użytkownika



Rys. 13: Widok rejestracji sklepu