

UNIwersytet w Białymstoku  
Wydział Matematyki i Informatyki  
Instytut Informatyki

Dawid Syczewski

STRONA INTERNETOWA  
RESTAURACJI Z SYSTEMEM  
ZARZĄDZANIA TREŚCIĄ  
I ZAMÓWIENIAMI ONLINE

*Praca dyplomowa napisana  
pod kierunkiem  
dr. Mariusza Żynela*

Białystok 2018

# Spis treści

Wstęp	1
<b>1 Analiza wymagań</b>	<b>3</b>
1.1 Cel i założenia pracy	3
1.2 Założenia	4
1.3 Rozwiązania podobne i znaczące różnice	4
1.4 Wymagania funkcjonalne i przypadki użycia	6
1.4.1 Przypadki użycia	6
1.4.2 Przypadek użycia (dokonanie zamówienia)	9
1.4.3 Przypadek użycia (dodanie pozycji w menu)	10
1.4.4 Wymagania dotyczące aplikacji dla administratora	11
1.4.5 Wymagania dotyczące aplikacji dla klientów	11
1.4.6 Wymagania niefunkcjonalne	11
<b>2 Wybrane technologie</b>	<b>12</b>
2.1 Technologie bazowe	12
2.1.1 Java	12
2.1.2 HTML, CSS, JavaScript, TypeScript	13
2.1.3 Biblioteki i rozwiązania	13
2.2 Warstwa wizualna	17
2.2.1 UX – User Experience	19
2.3 Narzędzia	20
2.3.1 IntelliJ IDEA	20
2.3.2 GIMP	20
2.3.3 Redis	20
<b>3 Realizacja projektu</b>	<b>21</b>
3.1 Architektura warstwy klienta	21
3.2 Architektura warstwy administratora	21
3.3 Baza danych	21
3.4 Wdrożenie systemu	22

---

<b>4 Aplikacja</b>	<b>23</b>
4.1 Frontend . . . . .	23
4.1.1 Strona główna . . . . .	24
4.1.2 Rejestracja, logowanie, odzyskiwanie hasła . . . . .	25
4.1.3 Profil . . . . .	26
4.1.4 Oferta . . . . .	34
4.1.5 Koszyk . . . . .	38
4.1.6 Zamówienie . . . . .	41
4.2 Portal kontrolny administratora . . . . .	46
<b>5 Podsumowanie</b>	<b>51</b>
<b>Spis rysunków</b>	<b>52</b>
<b>Bibliografia</b>	<b>53</b>

# Wstęp

Nie od dziś wiadomo, że placówki takie jak bary i restauracje korzystają z możliwości promowania się i realizowania zamówień poprzez Internet. Żyjemy w czasach gdzie ciężko jest prowadzić firmę nie istniejąc w sieci. Co za tym idzie każdy przedsiębiorca stara się zapewnić jak najwygodniejsze i najszybsze rozwiązania swoim klientom. W tym miejscu pojawia się wiele możliwości programistom, którzy tworzą coraz to bardziej zaawansowane systemy. Jeszcze kilkanaście lat temu nikt nie spodziewał się tak szybkiego rozwoju technologii i tego, że będzie można robić zakupy spożywcze bez wychodzenia z domu. Oczywiście teraz nikogo to już nie dziwi, że obiad zamawiamy poprzez stronę internetową. Wybieramy co chcemy zjeść, klikamy przycisk „zapłać” i możemy się spodziewać dostawcy, który pod wskazany adres dowiezie nam nasze zamówienie. Właśnie taki system w dobie Internetu jest potrzebny każdemu kto, prowadzi działalność gastronomiczną i nie ma tutaj wyjątków bo obecny konsument, który jest wiecznie w pośpiechu tego wymaga.

Celem pracy jest implementacja systemu internetowego do zarządzania funkcjonalnością strony restauracji. System dzieli się na widoczną część dla klienta, gdzie można dokonywać zamówień oraz warstwę administratora, który może dodawać produkty i je edytować. Drugą częścią będzie system zarządzania treścią, który będzie obsługiwany przez osobę upoważnioną do dodawania, edytowania i usuwania produktów. Obie te części tworzą spójną całość. System realizowany jest z zamiarem wykorzystania go w wielu restauracjach, więc z założenia musi być łatwy do przeniesienia do innych placówek. Przykładami miejsc, w których może być wykorzystywany są bary, restauracje, pizzerie, cukiernie i wszelkiego rodzaju obiekty gastronomiczne, które mogą czerpać korzyść dzięki promowaniu w internecie. Cechą wspólną dla wszystkich tych placówek jest oferowanie produktów aktualnie dostępnych w menu i chęć sprzedania ich potencjalnym klientom.

Strona udostępniac będzie funkcjonalności znane z wielu podobnych aplikacji znajdujących się w internecie. Jeżeli chodzi o klienta otrzyma on funkcjonalność założenia konta z możliwością edytowania swoich danych w panelu klienta. Osoba odwiedzająca będzie mogła przeglądać dostępne produkty i przeczytać krótkie opisy każdego dania. Klient w prosty sposób będzie mógł dodać do koszyka oferowane dania, a następnie dokonać płatności. Część systemu, do której ma wyłącznie dostęp administrator, będzie pozwalała na do-

dawania, edytowanie oraz usuwanie produktów na stronie internetowej. Panel obsługi danych elementów zrealizowany w jak najprostszy sposób aby osoba z podstawowymi umiejętnościami z zakresu obsługi komputera była w stanie swobodnie korzystać z dostępnych narzędzi. Interfejs zostanie stworzony z możliwością dalszego rozwoju. Co korzystnie wpłynie na sytuację firmy, która się rozwija i mogą pojawiać się kolejne zapotrzebowania na funkcje strony internetowej.

Praca podzielona jest na pięć rozdziałów. Rozdział pierwszy to wprowadzenie do ogólnego opisu założeń co do projektu czyli wstęp. Kolejny to analiza wymagań – zawiera dokładny zakres działania oraz cel i funkcjonalność strony internetowej. Poruszony również zostaje temat kwestii User Experience (UX z ang. doświadczenie użytkownika, por. [12]). Krótko przeanalizowana relacja w odniesieniu do interakcji człowiek-komputer na przedstawionym przykładzie strony internetowej. Z założenia projekt ma zostać zrealizowany w jak najbardziej przystępny sposób dla odbiorcy. Produkt powinien pozostawiać jak najlepsze wrażenie, prezentować dobrą funkcjonalność. Korzystanie z serwisu powinno być przyjemne i niemęczące. W rozdziale trzecim opisane zostały wybrane technologie wykorzystane do realizacji aplikacji internetowej. Przedstawione zostały aspekty użytych bibliotek dostępnych w wybranych technologiach oraz narzędzia programistyczne wykorzystane do tworzenia systemu. W kolejnej części opisana została realizacja strony internetowej. Przedstawiony został model realizacji najważniejszych części systemu zarządzania oraz omówienie plusów i minusów obranego kierunku tworzenia aplikacji. W ostatnim dziale znajdują się zrzuty ekranu ze zrealizowanego projektu strony internetowej restauracji oraz aspekty funkcjonalności po stronie panelu administratora.

# Rozdział 1

## Analiza wymagań

### 1.1 Cel i założenia pracy

Celem pracy jest stworzenie niezawodnego systemu obsługującego placówki gastronomiczne w najnowszych technologiach. Aplikacja ma wspomagać bary, restauracje i inne obiekty, które mają w swojej ofercie produkty. Jedną z najważniejszych kwestii jest stworzenie panelu zarządzania produktami. Natomiast użytkownik musi mieć możliwość przeglądania wszelkich dostępnych pozycji w ofercie firmy bez możliwości edycji poszczególnych elementów. Kolejną funkcjonalnością systemu ma być realizowanie zamówienia w prosty i przystępny sposób. Odwiedzający po wybraniu zestawu, który go interesuje musi go dodać do koszyka i dokończyć zamówienie. Kwestia realizowania zamówienia została przedstawiona za pomocą prostej funkcjonalności koszyka. Obsługa ma być intuicyjna bez zbędnych funkcjonalności, które komplikowałyby właścicielowi jak i klientowi obsługę systemu. Administracja stroną ma odbywać się bez konieczności udzielania pomocy przez osoby z branży informatycznej.

System jest kierowany do osób o małym kapitale finansowym. Na starcie małe firmy nie są w stanie przeznaczyć dużych pieniędzy na stronę internetową i jej zarządzanie. Stworzona oferta ma być bardzo przystępna co również nie będzie wymagało dodatkowego przeszkolenia z racji, że system będzie posiadał jak najbardziej podstawowe opcje obsługi. Mała lub średnia firma gastronomiczna, która dopiero decyduje się na sprzedaż i reklamowanie swojego biznesu w Internecie jest właśnie klientem docelowym tego systemu. Również w przypadku firm, które potrzebują czegoś bardziej rozwiniętego ten system jest dostosowany do rozwoju i powiększenia go o funkcjonalność np. prowadzenia danych dotyczących obrotów firmy. Można określić, że przedstawiany system jest pakietem podstawowym skierowanym do odbiorców o małym kapitale lub osób potrzebujących aplikacji „od ręki” lub na czas oczekiwania do momentu implementacji systemu docelowego o większych możliwościach.

## 1.2 Założenia

- Wdrożenie systemu musi być szybkie. Jednak doświadczona osoba jest wymagana do uruchomienia aplikacji. Jediną kwestią, która powinna być rozważona to edycja warstwy wyglądu strony internetowej i produkcja grafiki.
- Aplikacja powinna być obsługiwana bez konieczności instalowania aplikacji na telefonie lub komputerze.
- Odbiorcą docelowym ma być osoba rozpoczynająca działalność z małym kapitałem finansowym dlatego też system musi być jak najłatwiejszy w obsłudze bez konieczności udziału w szkoleniach.
- System ma w pełni obsługiwać realizację zamówień poprzez stronę internetową. Natomiast nie wyklucza to klienta, który po przejrzaniu menu decyduje się na zamówienie telefoniczne.

Jeżeli chodzi o zakres pracy przy danym projekcie obejmuje on:

- Przegląd dostępnych nowoczesnych technologii i narzędzi programistycznych odpowiednich do zrealizowania tematu pracy.
- Analiza ogólnych wymagań aplikacji jeżeli chodzi o branżę gastronomiczną typu restauracje i bary.
- Przygotowanie architektury.
- Testowanie stworzonego systemu i ewentualne poprawki.

## 1.3 Rozwiązania podobne i znaczące różnice

Na pierwszy rzut oka po wejściu klient nie odczuje żadnej różnicy pomiędzy wszelkimi innymi dostępnymi systemami na rynku. Istotna kwestia jest jednak po stronie administratora gdzie ten w panelu może edytować wszelkie dostępne funkcjonalności na stronie. Wizualnie dla odbiorcy to będzie zwykła strona restauracji o przyjemnym wyglądzie, która oferuje dokonanie zamówienia online bez zbędnych utrudnień. Większość tego typu placówek rezygnuje z zamawiania poprzez Internet ze względu na kosztowną produkcję i utrzymanie tego typu funkcjonalności. Wpisując w wyszukiwarkę nazwę lokalnej restauracji lub baru nie powinniśmy się spodziewać tego typu możliwości ze strony małego przedsiębiorcy. Ten gotowy system jest w stanie to zmienić i być łatwo dostępny dla każdego.

Bardzo często gdy już natrafimy na stronę, która oferuje realizację zamówień możemy spotkać się z problemem niezgodzących się cen lub brakiem

danego dania w menu. Jest to spowodowane brakiem systemu zarządzania produktami danej strony co powoduje utrudnienie w jej edycji. Mały przedsiębiorca byłby zmuszony zatrudniać osobę do wykonania tego prostego zadania co generowało by niepotrzebne koszty. Dzięki danemu systemowi właściciel bez problemu może wejść w panel administratora i dokonać wymaganych zmian.

Na rysunku 1.1 (strona 5) znajduje się zrzut ekranu podobnej strony internetowej popularnej pizzerii Savona, która oferuje zamówienia online.



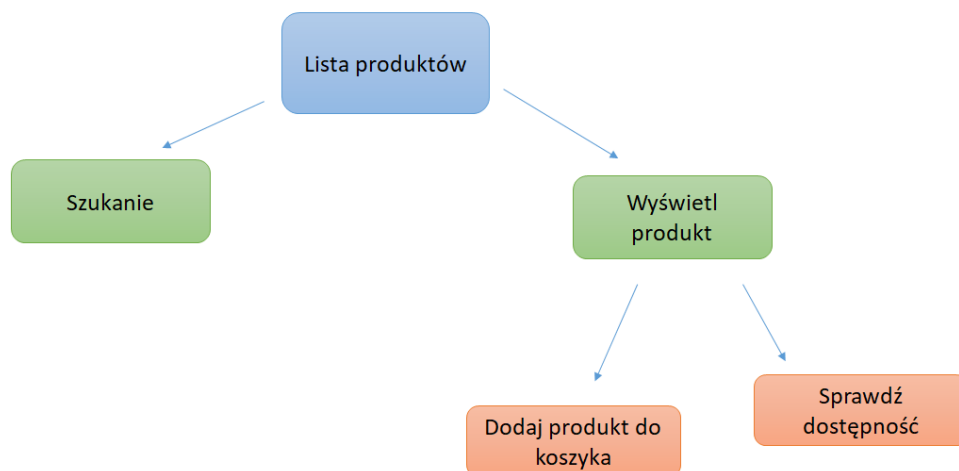
Rysunek 1.1: Strona internetowa Pizzerii Savona.

## 1.4 Wymagania funkcjonalne i przypadki użycia

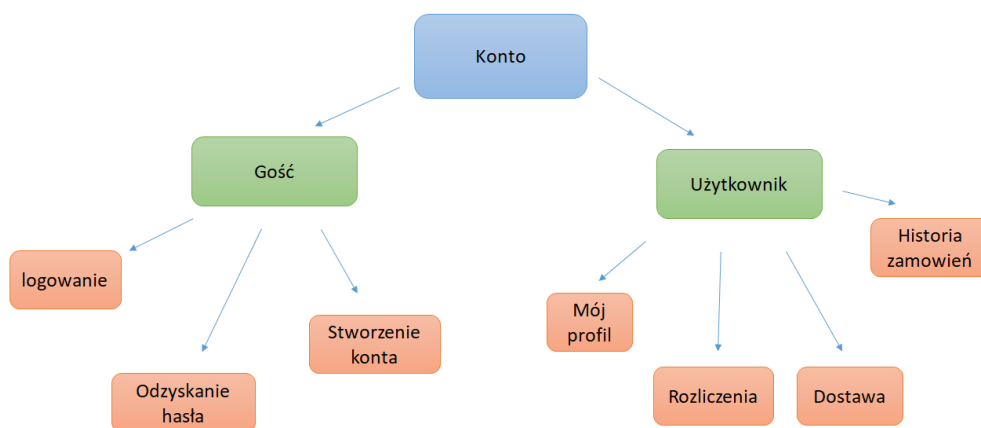
Wymagania funkcjonalne pozwolą nam opisać oczekiwane działanie aplikacji i określić jakie usługi i cechy są oferowane. Pozwolą nam zilustrować zachowanie systemu w różnych sytuacjach.



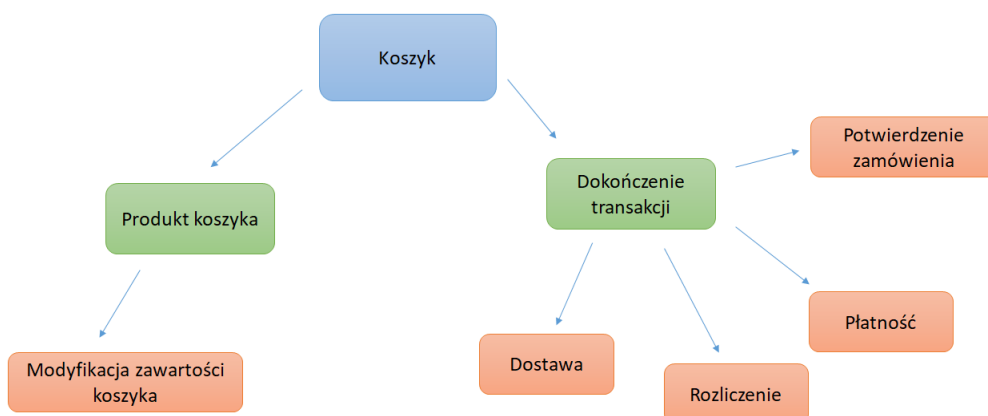
### 1.4.1 Przypadki użycia



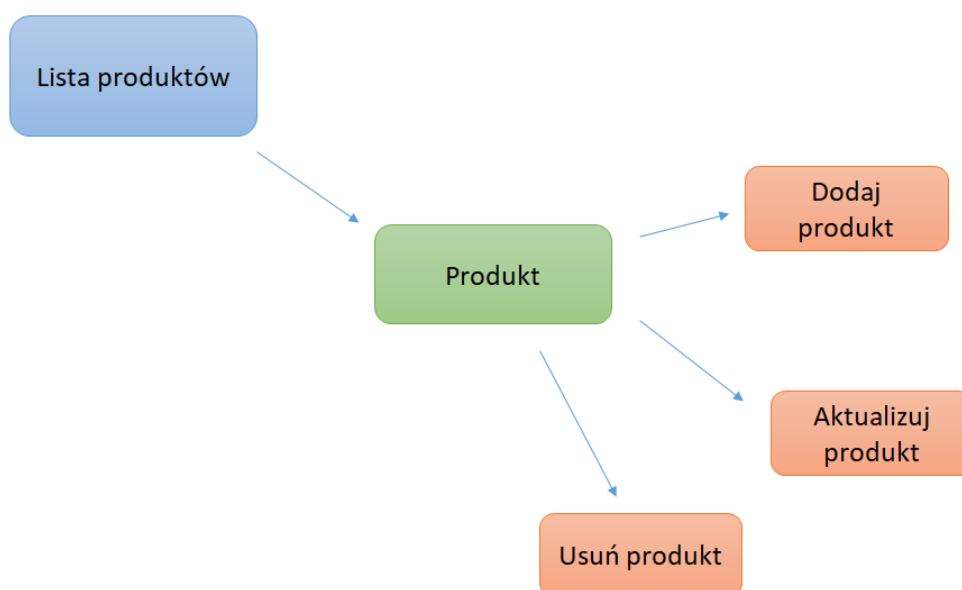
Rysunek 1.2: Schemat listy produktów.



Rysunek 1.3: Schemat konta użytkownika.



Rysunek 1.4: Schemat działania koszyka.



Rysunek 1.5: Schemat listy produktów po stronie administratora.

Poniżej znajdują się najważniejsze przypadki użycia oraz opis.

### 1.4.2 Przypadek użycia (dokonanie zamówienia)

#### 1. Uczestniczący aktorzy:

- Użytkownik (zalogowany).

## 2. Podstawowy ciąg zdarzeń:

- Użytkownik wyszukuje interesujące go danie.
- Użytkownik dodaje interesujące go danie.
- Użytkownik określa ilość.
- Użytkownik wypełnia podstawowe dane potrzebne do zrealizowania zamówienia.
- Po zatwierdzeniu system przenosi użytkownika do podsumowania.

## 3. Alternatywny ciąg zdarzeń:

- Po dodaniu produktów do koszyka użytkownik nie finalizuje zamówienia.
- Po dodaniu produktów do koszyka użytkownik usuwa dodane zamówienie.

## 4. Zależność czasowa:

- Częstotliwości wykonywania: nieokreślony.
- Przewidywane spiętrzenie: w godzinach obiadowych.
- Zakładany czas realizacji: do godziny.
- Maksymalny czas realizacji: nieokreślony.

### **1.4.3 Przypadek użycia (dodanie pozycji w menu)**

#### 1. Uczestniczący aktorzy:

- Administrator.

#### 2. Podstawowy ciąg zdarzeń:

- Administrator loguje się.
- Administrator otwiera zakładkę dodawania kolejnego produktu i wypełnia podstawowe informacje w formularzu i dodaje zdjęcie.
- Administrator zatwierdza dane.

#### 3. Alternatywny ciąg zdarzeń:

- Administrator rezygnuje z dodania produktu i go nie zatwierdza.

#### 4. Zależność czasowa:

- Częstotliwości wykonywania: nieokreślony.
- Przewidywane spiętrzenie: nieokreślony.
- Zakładany czas realizacji: do 5 minut.
- Maksymalny czas realizacji: nieokreślony.

#### 1.4.4 Wymagania dotyczące aplikacji dla administratora

- Wejście do panelu administracyjnego wymaga podania loginu i hasła. Konto musi mieć uprawnienia i role administratora.
- Konto umożliwiające obsługę panelu administratora jest tylko jedno i niedostępne poprzez zwykłą rejestrację, natomiast istnieje możliwość stworzenia dodatkowych kont poprzez bazę danych.
- Możliwość edytowania, dodawania, usuwania produktów.

#### 1.4.5 Wymagania dotyczące aplikacji dla klientów

- Możliwość rejestracji konta i logowanie do swojego profilu za pomocą hasła i loginu.
- Odzyskanie hasła.
- Użytkowanie strony głównej i dostęp do wszelkich funkcjonalności z jej poziomu.
- Możliwość przeglądania wszystkich dostępnych produktów.
- Dodawanie wszystkich dostępnych dań do koszyka z zakładki produktów.
- Możliwość przeglądania dotychczasowych zamówień z zakładki swojego profilu.
- Możliwość zmiany danych swojego konta.

#### 1.4.6 Wymagania niefunkcjonalne

Cechy i funkcjonalności systemu, które sprawiają, że jest on wartościowy dla potencjalnego klienta:

- System jest dostępny z przeglądarki na komputerze.
- Jednocześnie system może być użytkowany przez wiele osób zalogowanych.
- System jest bardzo prosty w obsłudze i nie wymaga dodatkowego szkolenia. Właściciel już po kilku minutach jest w stanie poruszać się swobodnie i dodawać produkty. Nie jest wymagane żadne dodatkowe szkolenie z zakresu obsługi systemu.
- Konfiguracja systemu trwa nie dłużej niż 14 dni.

Cechy te mogą również w szczególny sposób ułatwić i zwiększyć sprzedaż.

# Rozdział 2

## Wybrane technologie

Rozdział zawiera opisy wszelkich narzędzi, które zostały użyte podczas tworzenia aplikacji. Podzielony zostanie na trzy części: „Technologie bazowe”, „Biblioteki i rozwiązania” oraz „Warstwa wizualna”. Pierwszy podrozdział zawierać będzie opisy podstawowych technologii, które zostały użyte do realizacji projektu. Kolejny zaś zawiera opisy bibliotek czyli popularnych gotowych rozwiązań, które znacząco wspomagają podczas produkcji aplikacji. Najwięcej czasu jednak zostanie poświęcone opisowi podstaw funkcjonalności strony, która została wykonana za pomocą języka Java. Aplikacja użytkowana będzie za pomocą przeglądarki także bazą technologiczną wyglądu i działania strony będzie standardowo HTML, CSS oraz dodatkowe biblioteki w podrozdziale 2.1.3.

### 2.1 Technologie bazowe

Projekt jest aplikacją internetową, więc zostanie podzielona na technologie działające po stronie serwera i po stronie przeglądarki czyli naszego komputera. Po stronie serwera działać będzie aplikacja napisana w języku Java z frameworkiem Spring [10]. Natomiast z racji tego, że system jest stroną internetową zostanie użyty standardowo popularny HTML, CSS, Angular 6 i dodatkowe biblioteki, które zostaną wspomniane w podrozdziale 2.1.3.

#### 2.1.1 Java

Java [5] jest obiektowym językiem programowania. Została stworzona przez Jamesa Goslinga i jego grupę w firmie Sun Microsystems. Jest bardzo silnie typowana. Jest językiem kompilowanym przez maszynę wirtualną do kodu binarnego. Ze względu na to, że Java pojawiła się już w roku 1995 jest bardzo rozwinięta. Daje nam nieograniczone możliwości tworzenia oprogramowania. Wirtualna maszyna daje nam możliwość uruchomienia aplikacji na popularnych systemach. Dodatkowo od wielu lat powstają frameworki, które rozsze-

rzają możliwości Javy. Ograniczeniem w pewnym stopniu jest fakt, że Java wymaga wirtualnej maszyny na serwerze hostingu.

### 2.1.2 HTML, CSS, JavaScript, TypeScript

**HTML (HyperText Markup Language).** HTML [4] wraz ze stylami CSS [6] oraz językiem JavaScript jest praktycznie podstawą każdej strony internetowej w sieci. Co oznacza, że ciężko go zastąpić. Jest to język wykorzystujący znaczniki do tworzenia dokumentów hipertekstowych. Działa to dzięki wbudowanym mechanizmom w przeglądarki, które po otrzymaniu kodu interpretują go i wyświetla zamierzony efekt na ekranie użytkownika.

**CSS (Cascading Style Sheets).** CSS to język opisujący kwestie prezentacji strony WWW. W rozwinięciu kaskadowe arkusze stylów to tak naprawdę reguły, które określają jak dany element na stronie ma wyglądać. Za pomocą CSS możemy określać kolory, odstępy, rozmiary, dodawać ramki, obracać elementy i decydować ogólnie o wszelkich efektach wizualnych. Możemy również pozycjonować wszelkie elementy naszej strony. Dużą zaletą jest również to, że arkusze mogą zostać użyte do wielu wspólnych dokumentów.

**JS (JavaScript).** JavaScript to język skryptowy, który jest wykonywany po stronie przeglądarki. Najczęściej zostaje on wykorzystany do produkcji aplikacji internetowych WWW. Jest on zazwyczaj odpowiedzialny za interakcje serwisu. Wszelkie reagowanie na kliknięcia, najechania myszą lub wypełnianie błędnie formularza z wyświetlaniem komunikatów możemy właśnie obsługiwać za pomocą języka JavaScript [1]. Język ten przeszedł bardzo długą drogę, ponieważ został zapoczątkowany w 1996 roku i jest rozwijany do dzisiaj. Jego osadzenie w plikach HTML jest banalnie proste i nie wymaga dodatkowych instalacji.

**TS (TypeScript).** TypeScript [3] został stworzony przez firmę Microsoft. Jest pewnego rodzaju nakładką na język JavaScript i rozszerza jego możliwości o programowanie zorientowane obiektowo. Możemy pisać w oparciu o klasy, funkcje lub interfejsy. TypeScript może zostać skompilowany do JavaScript, dzięki czemu może zostać uruchomiony w przeglądarce.

### 2.1.3 Biblioteki i rozwiązania

**Bootstrap.** Bootstrap [2] to bez wątplenia jedna z najpopularniejszych bibliotek HTML, CSS oraz JS. Korzystają z niej jedne z największych portali w Internecie takie jak np. Twitter. Jest gotowym rozwiązaniem do tworzenia nowoczesnych stron i aplikacji. Obecnie popularniejsza wersja to 3.3.7 natomiast już kolejna z oznaczeniem v4 jest dostępna do użytku. Za pomocą bootstrapa możemy stylizować formularze, przyciski, nawigacje i wszelkie inne

elementy, które zawsze znajdują się na portalach internetowych. Największą zaletą jest również to, że dzięki użyciu tej biblioteki uzyskujemy pełną responsywność. Wykorzystujemy do tego strukturę dwunastu kolumn, a każdy element z odpowiednią klasą reaguje na zmniejszanie rozdzielczości ekranu. Dzieje się to automatycznie wystarczy dodać odpowiednią klasę.

**Angular 6.** Angular [7] to platforma, która pomaga budować aplikacje w Internecie. Łączy w sobie szablony, wstrzykiwanie zależności, najlepsze praktyki oraz narzędzia do rozwiązywania problemów związanych z wytwarzaniem aplikacji. Pozwala na tworzenie systemów działających w Internecie, komputerze oraz telefonie. Jest wspierany oraz rozwijany przez firmę Google.

Nowa wersja Angulara wprowadza interfejs wiersza poleceń. Najpierw musimy zainstalować node package manager. Jeżeli wszystko przebiegło pomyślnie komenda `npm -V` wyświetli nam wersję, którą posiadamy.

Angular CLI (interfejs wiersza poleceń) pozwala używać komend do budowania aplikacji. Jest to metoda znana z innych języków programowania takich jak Ruby on Rails. Kilka przykładowych poleceń, które są niezbędne i ułatwią nam prace z tym frameworkiem:

`ng new [nazwa]` – za pomocą tego polecenia tworzymy nowy czysty projekt

`ng generate [komponent] [nazwa]` – tej komendy stosujemy w momencie kiedy chcemy wygenerować nowy komponent, model lub interfejs.

`ng serve` – uruchomia aplikację na domyślnym porcie 4200. Jeżeli chcemy zmienić port na inny wystarczy dodać port w poleceniu.

Większość podstawowych ustawień znajduje się w głównym katalogu projektu. Główną zaletą, która przemawia na korzyść nowej wersji tego frameworku jest jeszcze lepiej przystosowane środowisko. Nie musimy zajmować się konfiguracją systemów automatyzacji pracy takich jak Gulp czy też Grunt. Po uruchomieniu aplikacji za pomocą polecenia `ng serve` możemy skupić się na programowaniu. Każda zmiana w edytorze po zapisaniu spowoduje przeładowanie aplikacji. W razie małych problemów takich jak brak modelu danych czy też serwisu zostaniemy poinformowani.

Aplikacje budowane za pomocą Angulara składają się z drzew komponentów. Są to podstawowe elementy, które zawierają poszczególne widoki strony. Przeglądając strukturę folderów i plików aplikacji Angular 6, powinna ona wyglądać następująco:

`e2e`

`node_modules`

`src`

app (najważniejsze pliki aplikacji)

Większość pracy wykonuje się w folderze src/app. Właśnie tutaj przechodzimy komponenty, serwisy oraz modele. W momencie tworzenia kolejnego elementu aplikacji za pomocą interfejsu wiersza poleceń plik app.module.ts automatycznie zostaje zaktualizowany. Rozszerzenie .ts dotyczy oczywiście języka TypeScript, który jest zalecany do tworzenia aplikacji za pomocą Angulara. W momencie tworzenia kolejnego komponentu konsola zwróci nam następujące informacje:

```
ng generate component example
```

- CREATE src/app/example/example.component.html
- CREATE src/app/example/example.component.spec.ts
- CREATE src/app/example/example.component.ts
- CREATE src/app/example/example.component.css
- UPADATE src/app/app.module.ts

W pliku example.component.html możemy korzystać z wszystkich możliwości HTML. W projekcie dodatkowo używamy możliwości bootstapa. W każdym znaczniku możemy umieszczać style CSS lub umieszczać wszystko w oddzielnym pliku example.component.css.

**Angular 6 Material.** Angular Material to zbiór komponentów, które pozwalają na szybkie budowanie warstwy wizualnej. Posiada wszystkie potrzebne elementy, które mogą wystąpić na stronie internetowej takie jak przyciski, formularze, nawigacje i wiele innych. Jest to biblioteka dedykowana do Angulara i aby z niej skorzystać wystarczy zainstalować ją za pomocą komendy:

```
npm install -save @angular/material
```

Angular wraz z biblioteką Material został wykorzystany jako podstawa do wytworzenia frontendu aplikacji, czyli warstwy wizualnej. Zawiera w sobie wszystkie metody kontrolujące zdarzenia takie jak akcje przesłania treści z formularza, wyświetlenie produktów lub zawartości koszyka.

**Spring.** Pierwsza wersja tego frameworka pojawiła się w 2002 roku i od tamtej pory jest dalej rozwijana. Spring został stworzony przez Roda Johnsona wraz z Juergenem Hoellerem i Yannem Caroffem. Już w 2005 roku posiadał bardzo dużą popularność wśród programistów Javy. Spring oferuje dużą swobodę w projektowaniu aplikacji. Powstał jako konkurencja Javy EE i we współczesnych wersjach widać, że oba rozwiązania korzystają z podobnego podejścia. Zawiera w sobie wiele projektów, które są dynamicznie rozwijane. Kluczowym elementem jest kontener wstrzykiwania zależności. z początku



największą trudnością sprawiała konfiguracja, która nie należała do najprostszych. z biegiem czasu wprowadzono jednak wiele uproszczeń, które pomagają w tworzeniu aplikacji za pomocą Javy. Framework Spring zawiera w sobie wiele pomniejszych projektów, które odpowiadają na przykład za wstrzykiwanie zależności, autoryzacje, bezpieczeństwo czy też zarządzanie bazą danych. Od niedawna również prężnie rozwijana funkcja związana z działaniem chmurowym. Obecnie jedynymi najważniejszymi projektami wchodzącymi w skład Springa to:

**Spring Boot** – ułatwia tworzenie samodzielnych aplikacji Spring opartych na wersjach produkcyjnych, które można uruchomić bez skomplikowanej konfiguracji.

**Spring Data** – dostarcza nam rozwiązania dostępne w modelu programowania za pomocą Springa, zachowując przy tym specjalne cechy bazowego magazynu danych. Ułatwia korzystanie z technologii dostępu do danych, relacyjnych i nierelacyjnych baz danych oraz chmurowych usług danych. Zawiera w sobie wiele podprojektów, które są specyficzne dla danej bazy danych. Projekty są opracowywane we współpracy z wieloma firmami i programistami.

**Spring Security** – to potężne i wysoce konfigurowalne środowisko uwierzytelniania i kontroli dostępu. Jest to standard zabezpieczania aplikacji Spring. Koncentruje się na zapewnieniu uwierzytelniania i autoryzacji aplikacji Java. Podobnie jak wszystkie projekty Spring, prawdziwą zaletą są jego możliwości.

Oczywiście są to przykładowe projekty. Jest ich o wiele więcej i wszystkie cieszą się dużą popularnością wśród programistów Java. Dodatkową zaletą jest również obszerna dokumentacja i wiele poradników, które ułatwiają rozpoczęcie przygody w pisaniu aplikacji webowych i nie tylko.

Spring został wykorzystany do wytworzenia warstwy backendu aplikacji z podejściem architektonicznym REST (Representational State Transfer), czyli zmiana stanu aplikacji poprzez reprezentacje. To podejście polega na serwowaniu danych w formacie JSON. Każdy adres jest zawsze taki sam dla danego zasobu. Powinien być również unikalny. Wprawdzie rozwiązania REST nie wymusza użycia HTTP, ale prawie zawsze idzie z nim w parze. To coś więcej niż znane wszystkim metody GET oraz POST. Mamy nagłówki żądania i odpowiedzi. Napisany w ten sposób backend możemy testować za pomocą serowania danych za pomocą narzędzia cURL, które umożliwia wysyłanie zapytań do API. Popularnym narzędziem jest również postman, który posiada graficzny interfejs i jest bardziej przystępny dla osób z mniejszym doświadczeniem programistycznym. Takie podejście daje nam możliwość testowania aplikacji bez wytworzenia warstwy wizualnej czyli frontendu aplikacji.

## 2.2 Warstwa wizualna

W tym podrozdziale zostaną omówione kwestie graficzne strony. Poruszymy temat doświadczenia użytkownika na stronie i jak zadbać o interfejs aby był zrozumiały dla każdego.

**Grafika i fotografie na stronie.** Wszelkie fotografie zostały wykonane przeze mnie. Postanowiłem poświęcić temu nieco uwagi aby strona wyglądała atrakcyjniej i nie była tylko wersją prezentacyjną.

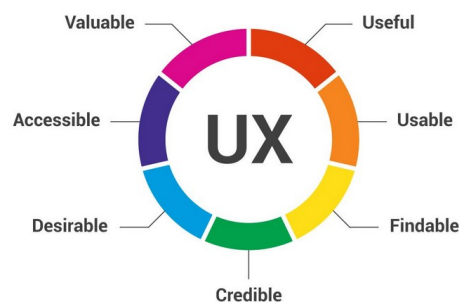


Rysunek 2.1: Przykładowa fotografia użyta w pracy.



Rysunek 2.2: Logo wykonane na potrzeby projektu.

### 2.2.1 UX – User Experience



Rysunek 2.3: Podział doświadczeń użytkownika.

User experience, czyli doświadczenie użytkownika odnosi się do oceny odczuć na temat danego projektu względem jego wrażeń. Najważniejsze obecnie na rynku jest zaprojektowanie aplikacji nie tylko aby działała, lecz również aby była użyteczna. Bardzo często w większych firmach zatrudnia się osoby odpowiedzialne za projektowanie UX i jest to bardzo dobre podejście. Obecny

rynek wymaga jak najlepiej przemyślanych aplikacji ze względu na to, że użytkownicy Internetu są w bardzo różnych przedziałach wiekowych. Coś co może wydawać się oczywiste osobie w wieku 20 lat nie koniecznie musi być zrozumiałe seniorom.

## 2.3 Narzędzia

W tym rozdziale znajdują się podstawowe narzędzia, które zostały wykorzystane podczas tworzenia aplikacji. Wszystkie są darmowe lub zostały użyte w wersjach edukacyjnych.

### 2.3.1 IntelliJ IDEA

IntelliJ [9] to zintegrowane środowisko programistyczne od firmy JetBrains. Obsługuje większość popularnych języków. Wspiera język Java oraz popularne frameworki takie jak Play czy też Spring.

### 2.3.2 GIMP

Gimp [8] to bardzo lekki program do edycji grafiki rastrowej. Jest na darmowej licencji i pozwala w niewielkim stopniu zastąpić narzędzia takie jak Photoshop. Posiada wszelkie niezbędne narzędzia do stworzenia podstawowej grafiki strony i obróbki zdjęć.

### 2.3.3 Redis

Redis to darmowy magazyn przechowywania struktur danych, używany jako baza danych lub pamięć podręczna.

# Rozdział 3

## Realizacja projektu

Realizacja projektu polega na rozdzieleniu warstw aplikacji na frontend i backend. Jest to zabieg ułatwiający dalszy rozwój aplikacji. Pliki zostają rozdzielone na te, które odpowiadają za funkcjonowanie aplikacji w kwestii strony internetowej i te, które kierują jej działaniem. W kolejnych podrozdziałach znajdują się opisy poszczególnych funkcjonalności obu warstw.

### 3.1 Architektura warstwy klienta

Warstwa kliencka została zawarta w folderze `frontend-restauracji`. Następnie została ona podzielona na podkatalogi, które wskazują na dane elementy strony. Każdy z nich odpowiada za poszczególne funkcjonowanie strony i zawiera odpowiednie komponenty, modele oraz serwisy, które realizują założone zadania.

### 3.2 Architektura warstwy administratora

Warstwa administratora odpowiada za kwestie funkcjonalności systemu zarządzania produktami i jest dostępna po zalogowaniu się konta z rolą administratora. System posiada tylko jedno takie konto ze względów bezpieczeństwa. Kontroler sprawdza za każdym razem jakie prawa ma przypisane dane konto aby uniknąć komplikacji.

### 3.3 Baza danych

Baza danych została oparta na MySQL. Jest tworzona za pomocą mapowania obiektowo-relacyjnego (ORM, Object Relational Mapping). Wykorzystany został do tego framework Hibernate, który zapewnia translację pomiędzy warstwą obiektów, a relacyjną bazą danych. Autorem tego frameworka jest firma Red Hat. Takie rozwiązanie jest świetne w aplikacjach z technologią Spring. Do

skorzystania z tej metody musimy wykorzystać Spring Data, który udostępnia nam interfejs JDBC. Możemy w wygodny sposób definiować mapowanie z wyników zapytań na obiekty Javy lub uzupełniać parametry automatycznie na podstawie nazw pól w klasach. Minusem tego podejścia jest powtarzalność kodu. Nadal nie jest to jednak rozwiązanie idealne, ponieważ musimy posiadać wiedzę związaną z SQL. Przy tym podejściu podczas kompilacji zostaje utworzona baza danych. Jeżeli środowisko programistyczne wykryje zmiany, dokona aktualizacji tabel.

### 3.4 Wdrożenie systemu

Aplikacja została przygotowana do prezentacji lokalnie. Docelowo system wymaga serwera oraz hostingu z wirtualną maszyną Javy i bazy MySQL. Zbudowany projekt należy umieścić na hostingu oraz podłączyć bazę danych. Ze względu na wysokie koszty jeżeli chodzi o wykupienie miejsca na serwerze, system zostanie zaprezentowany lokalnie uruchomiony za pomocą kontenera Tomcat [11] wbudowanego w Spring Boot.

# Rozdział 4

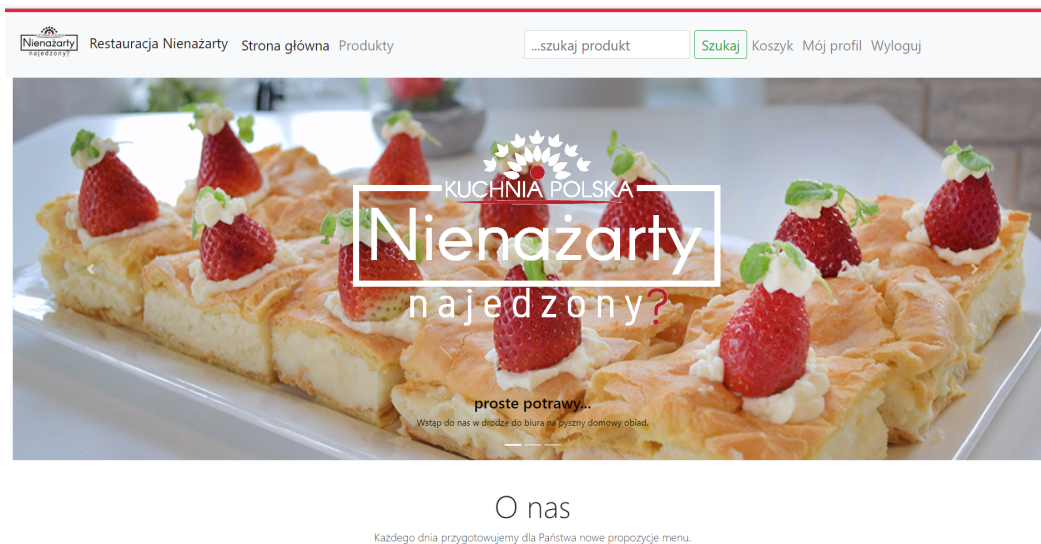
## Aplikacja

Aplikacja przeszła szereg testów z udziałem użytkownika systemu przez człowieka. Jak wspomniano w początkowej części dokumentacji system docelowo ma trafić do małych przedsiębiorców z niewielkim budżetem. Aplikacja pomyślnie przeszła zakładaną funkcjonalność i bez przeszkód może być wykorzystywana przez firmy gastronomiczne.

### 4.1 Frontend

Realizacja strony głównej jest bardzo prosta i intuicyjna. Użytkownik poruszając się po stronie ma dostęp do podstawowych informacji, a po wejściu w zakładkę produkty może dokonać zamówienia. Szablon strony jest podstawowy i powinien zostać wykonany indywidualnie do każdego klienta. Natomiast prawdziwa funkcjonalność zostaje ukazana dopiero po założeniu konta i zalogowania się na nie. Na poniższych zrzutach ekranu znajduje się widok elementów strony.

### 4.1.1 Strona główna



Rysunek 4.1: Strona główna restauracji.

### 4.1.2 Rejestracja, logowanie, odzyskiwanie hasła

Logowanie   **Nowe konto**   Odzyskanie hasła

---

Nazwa użytkownika  
Podaj nazwę użytkownika

E-mail  
Wszystkie wiadomości będą wysłane na podany e-mail.

**Utwórz nowe konto**

Rysunek 4.2: Formularz tworzenia nowego konta.



### 4.1.3 Profil

Logowanie      Nowe konto      **Odzyskanie hasła**

Twój adres e-mail \*

Podaj adres e-mail podany przy reejstracji konta.

**Potwierdź**

Rysunek 4.3: Formularz odzyskiwania hasła.

## Mój profil

Edycja	Zamówienia	Płatność	Dostawa
--------	------------	----------	---------

---

Imię

Nazwisko

Login \*

Aktualne Hasło

Prosimy o podanie aktualnego hasła do zmiany hasła lub e-maila.

Adres Email \*

Wszelkie informacje zostaną przesłane na podany email.

Nowe Hasło \*

Potwierdź hasło \*

Rysunek 4.4: Formularz profilu użytkownika.

## Mój profil

Edycja   **Zamówienia**   Płatność   Dostawa

---

Kod zamówienia	Numer zamówienia	Suma	Status
1531847737000	1	16	utworzone
1531852750000	2	16	utworzone

Rysunek 4.5: Zakładka z listą zamówień użytkownika.





## Mój profil

Edycja   Zamówienia   **Płatność**   Dostawa

---

Lista kart kredytowych   Dodaj/Zaktualizuj kartę

---

Domyślna	Karta	Akcja	
<input checked="" type="radio"/> domyślna	Moja karta nr1		
<input type="radio"/> domyślna	Moja karta nr2		

**Zapisz**

Rysunek 4.6: Zakładka płatności z dostępną listą kart płatniczych.

## Mój profil

Edycja	Zamówienia	<b>Płatność</b>	Dostawa
--------	------------	-----------------	---------

---

[Lista kart kredytowych](#)   [Dodaj/Zaktualizuj kartę](#)

---

**\* Nazwa karty:**

---

**\* Nazwa**

---

**\* Adres**

---

<b>* Miasto</b>	<b>* Województwo</b>	<b>* Kod pocztowy</b>
<input type="text" value="Białystok"/>	<input type="text" value="podlaskie"/>	<input type="text" value="15-178"/>

---

Rysunek 4.7: Formularz dodawania/aktualizacji danych karty.

## Informacje karty



\* Wybierz typ karty

\* Imię Właściciela

\* Numer karty

\* Data wygaśnięcia

CV kod





Zapisz

Rysunek 4.8: Informacje o karcie.

## Mój profil

Edycja      Zamówienia      Płatność      **Dostawa**

Lista adresów dostawy      Dodaj/Aktualizuj adres dostawy

Domyślna	Adres Dostawy	Akcja	
<input type="radio"/> domyślna	Monte Cassino, Białystok, podlaskie		
<input checked="" type="radio"/> domyślna	Simony Kossak, Białystok, podlaskie		

**Zapisz**

Rysunek 4.9: Zakładka dostaw z listą dostępnych adresów.

## Mój profil

Edycja	Zamówienia	Płatność	<b>Dostawa</b>
--------	------------	----------	----------------

---

Lista adresów dostawy	<b>Dodaj/Aktualizuj adres dostawy</b>
-----------------------	---------------------------------------

---

\* Nazwa

\* Adres

\* Miasto                      \* Województwo                      \* Kod pocztowy

<input type="text" value="Białystok"/>	<input style="border-bottom: none; border-right: none; border-top: none; border-left: none; padding: 0 5px; text-align: center; font-size: 0.8em;" type="text" value="podlaskie"/> ▼	<input type="text" value="15-352"/>
--	--	-------------------------------------

---

**Zapisz**

Rysunek 4.10: Formularz dodania/aktualizacji adresu dostawy.

## 4.1.4 Oferta


### Lista produktów

---


Filtruj po opisie produktu:

Pozycji na stronie


---



**Łabędzi puch**  
Data dodania: 2018-07-12  
Producent: Nienażarty  
Ilość kalorii: 525  
**Cena: 5.00 PLN** ~~7.00 PLN~~  
Opis: Kruche ciasto pokryte dżemem jagodowym, z masą serową. Na wierzchu puszysta śmietana pokryta wiórkami kokosowymi.



**Naleśniki**  
Data dodania: 2018-07-26  
Producent: Nienażarty  
Ilość kalorii: 345  
**Cena: 11.00 PLN** ~~15.00 PLN~~  
Opis: Do wyboru naleśniki z mascarpone z miętą, jabłkami lub z serem i rodzynkami



**Kotlet mielony**  
Data dodania: 2018-07-13  
Producent: Nienażarty  
Ilość kalorii: 497  
**Cena: 12.00 PLN** ~~15.00 PLN~~

Rysunek 4.11: Widok listy produktów.





### Kotlet mielony

Data dodania: 2018-07-13

Producent: Nienażarty

Ilość kalorii: 497

**Cena: 12.00 PLN** ~~15.00 PLN~~

Opis: kotlet mielony z prażoną cebulką, z ziemniakami/kaszą/ryżem i surówką z białej kapusty



### Kacze udko

Data dodania: 2018-07-05

Producent: Nienażarty

Ilość kalorii: 377

**Cena: 16.00 PLN** ~~18.00 PLN~~

Opis: Udko z kaczki z konfiturą i gruszkami, kasza gryczana, surówka z kapusty



### Karpotka

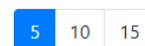
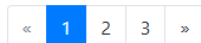
Data dodania: 2018-07-11

Producent: Nienażarty

Ilość kalorii: 299

**Cena: 7.00 PLN** ~~9.00 PLN~~

Opis: Ciasto ptysiowe przełożone kremem budyniowym



Rysunek 4.12: Lista produktów z dodatkowymi przyciskami stronicowania.

## Informacje o produkcie

[Powrót do produktów](#)



### Łabędzi puch

**Producent:** Nienazarty

**Kraj pochodzenia:** Polska

**Data dodania:** 2018-07-12

**Kategoria:** deser

**Ilość kalorii:** 525 kalorii

**Kod produktu:** 3673

**Waga:** 100 g

**Cena promocyjna:** 5.00PLN

List Price: 7.00 PLN

Ilość:

**Produkt dostępny**

[Dodaj do koszyka](#)

Krucze ciasto pokryte dżemem jagodowym, z masą serową. Na wierzchu puszysta śmietana pokryta wiórkami kokosowymi.

Rysunek 4.13: Podgląd produktu.

## Informacje o produkcie

[Powrót do produktów](#)



### Omlet

**Producent:** Nienażarty

**Kraj pochodzenia:** Polska

**Data dodania:** 2018-07-13

**Kategoria:** deser

**Ilość kalorii:** 415 kalorii

**Kod produktu:** 6373

**Waga:** 180 g

**Cena promocyjna:** **15.00PLN**

List Price: ~~17.00 PLN~~

Ilość:

**Niedostępny**

[Dodaj do koszyka](#)

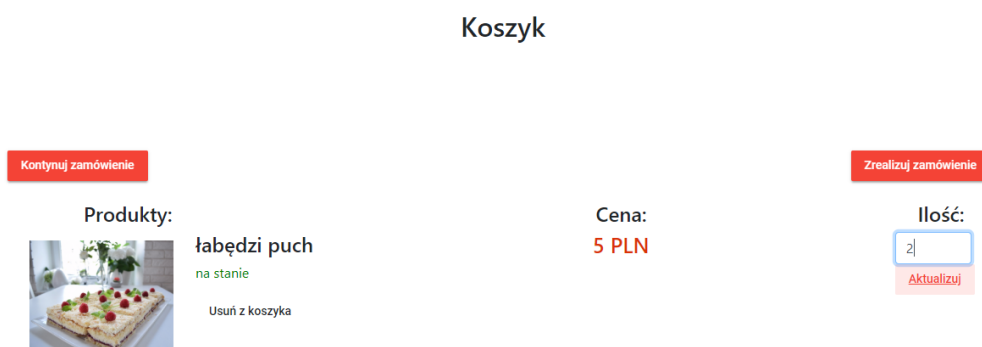
Do wyboru: omlet z szynką i szpinakiem, omlet na słodko z dżemem, omlet vege z papryką i pomidorami

Rysunek 4.14: Podgląd produktu ze statusem niedostępny.

### 4.1.5 Koszyk



Rysunek 4.15: Widok koszyka z dodanymi produktami.




Rysunek 4.16: Aktualizacja ilości produktu.

## Koszyk

[Kontynuuj zamówienie](#) [Zrealizuj zamówienie](#)

Koszyk został zaktualizowany!

Produkty:	Cena:	Ilość:
 łabędzi puch na stanie <a href="#">Usuń z koszyka</a>	5 PLN	<input type="text" value="2"/> <a href="#">Aktualizuj</a>

Rysunek 4.17: Komunikat potwierdzający aktualizację produktu.

#### 4.1.6 Zamówienie

**Złóż zamówienie**

---

## Podsumowanie zamówienia

Cena bez opakowań :37 PLN  
Podatek VAT :8.51 PLN

---

## Całkowite zamówienie:45.51 PLN

1. Adres dostawy      2. Informacje płatności      3. Przegląd

---

### 1. Adres dostawy

Dostępny adres dostawy	Akcja
Monte Cassino, 21 / 16, Białystok, podlaskie	zastosuj
Simony Kossak, 4, Białystok, podlaskie	zastosuj

\* Nazwa

\* Adres

\* Miasto      \* Województwo      \* Kod pocztowy

**Dalej**

Rysunek 4.18: Formularz podsumowania zamówienia.

[Złóż zamówienie](#)

## Podsumowanie zamówienia

Cena bez opakowań :37 PLN

Podatek VAT :8.51 PLN

**Całkowite zamówienie:45.51 PLN**

[1. Adres dostawy](#)[2. Informacje płatności](#)[3. Przegląd](#)

### 2. Informacje płatności

Dostępne karty kredytowe

Akcja


Moja karta nr1

zastosuj

Moja karta nr2

zastosuj

Rysunek 4.19: Zakładka z informacjami płatności w podsumowaniu zamówienia.




\* Wybierz rodzaj karty:

Visa ▼

\* Właściciel

Dawid

\* Numer karty

6748549474 

\* Data ważności                      Kod CV

Maj ▼    2020 ▼    123

Taki sam jak adres dostawy

\* Adres

Monte Cassino

21 / 18

\* Miasto                      \* Województwo                      \* Kod pocztowy

Białystok    podlaskie ▼    15-178

**Dalej**

Rysunek 4.20: Formularz zmiany danych do płatności.



[Złóż zamówienie](#)

## Podsumowanie zamówienia

Cena bez opakowań :37 PLN

Podatek VAT :8.51 PLN

**Całkowite zamówienie:45.51 PLN**

1. Adres dostawy

2. Informacje płatności

3. Przegląd

### 3. Przejrzyj produkty i dostawy

Wybierz metodę dostawy:

 Dostawa na dzisiaj Dostawa na jutro

#### Produkt

Cena:

Ilość:



Łabędzi puch

5 PLN

2

Produkt dostępny



Naleśniki

11 PLN

1

Produkt dostępny



Kacze udko

16 PLN

1




Produkt dostępny

Cena za wszystko: (3 sztuk/a zamówienia): **37 PLN**[Potwierdź zamówienie](#)

Rysunek 4.21: Podgląd podsumowania i listy produktów w zamówieniu.

✓ Dziękujemy, Twoje zamówienie zostało przyjęte.  
Przediwdywana data dostawy Środa, 2018/6/17

Lista produktów zamówienia:

- |   |                     |                  |
|---|---------------------|------------------|
|  | <b>łabędzi puch</b> | <b>5 PLNx 2</b>  |
|   | Na stanie           |                  |
|  | <b>Naleśniki</b>    | <b>11 PLNx 1</b> |
|   | Na stanie           |                  |
|  | <b>Kacze udko</b>   | <b>16 PLNx 1</b> |
|   | Na stanie           |                  |

Rysunek 4.22: Komunikat potwierdzający dokonania zamówienia.

## Zamowienie - 4



Odebrane x



projektrestauracja2018@gmail.com

do mnie ▾

Dzień dobry Dawid,

Dziękujemy za dokonania zamówienia na naszej stronie.

Numer zamówienia: #4. Oto szczegóły twojego zamówienia:

**Szczegóły zamówienia #4**

Rozliczenie	Płatność	Adres dostawy
Mój adres nr2 <a href="#">Simony Kossak 4</a> <a href="#">Białystok</a> <a href="#">podlaskie</a> <a href="#">15-352</a>	Dawid Visa 6748549474 Data wygaśnięcia: 5/20	Mój adres nr2 <a href="#">Simony Kossak 4</a> <a href="#">Białystok</a> <a href="#">podlaskie</a> <a href="#">15-352</a>

**Podsumowanie zamówienia**

Nazwa	Cena produktu	Ilość	Płatność
Kacze udko	18.0	1	16.00
Karpatka	9.0	1	7.00
		<b>Razem</b>	23.00
		<b>podatek</b>	5.29
		<b>Suma</b>	28.29

Pozdrawiamy, restauracja.

Rysunek 4.23: Potwierdzenie zamówienia wysłane na adres e-mail użytkownika.

## 4.2 Portal kontrolny administratora

Poniżej znajdują się zrzuty ekranu warstwy portalu administratora. Jest to centrum zarządzania produktami widocznymi na stronie głównej.

PORTAL ADMINISTRATORA							
Wyświetl Produkty   Dodaj Produkt   Wyloguj							
	Nazwa	Producent	Kategoria	Cena standardowa	Cena promocyjna	Aktywne	Akcja
<input type="checkbox"/>	łabętzki puch	Nienazarty	deser	7	5	true	x Usun
<input type="checkbox"/>	Naleśniki	Nienazarty	deser	15	11	true	x Usun
<input type="checkbox"/>	Kollet mielony	Nienazarty	danie	15	12	true	x Usun
<input type="checkbox"/>	Kacze udko	Nienazarty	danie	18	16	true	x Usun
<input type="checkbox"/>	Karpaka	Nienazarty	danie	9	7	true	x Usun
<input type="checkbox"/>	Omllet	Nienazarty	deser	17	15	true	x Usun
<input type="checkbox"/>	Chłodnik	Nienazarty	danie	10	7	true	x Usun
<input type="checkbox"/>	Sernik makowy	Nienazarty	ciasto	11	8	true	x Usun
<input type="checkbox"/>	Ryba	Bar Okraglak	danie	19	14	true	x Usun
<input type="checkbox"/>	Kurczak serowy	Bar Okraglak	danie	17	14	true	x Usun
<input type="checkbox"/>	SBIT	Bar Okraglak	ciasto	11	5	true	x Usun
<input type="checkbox"/>	Lasagne	Pociągódzeszenia	danie	19	17	true	x Usun
<input type="checkbox"/>	Piaciki ziemniaczane	Pociągódzeszenia	danie	17	13	true	x Usun

Rysunek 4.24: Portal administratora z listą produktów.

[←Pwrót](#) / [Edytuj](#)

**Nazwa produktu:** Naleśniki



**Producent:** Nienazarty

**Kraj:** Polska

**Data dodania:** 2018-07-26

**Język:** polski

**Kategoria:** deser

**Kalorie:** 345

**Alergeny:** nabiał

**Kod produktu:** 3473

**Waga:** 120

**Cena standardowa:** 15

**Cena promocyjna:** 11

**Dostępna ilość:** 50

**Opis:** Do wyboru naleśniki z mascarpone z miętą, jabłkami lub z serem i rodzynkami

Rysunek 4.25: Podgląd produktu w panelu.

**Informacje nowego produktu \* pole wymagane**

Nazwa \*

Producent \* Kraj \*

Data dodania \*  
dd----yyyy  Aktywny

Danie główne ▾ Orzechy ▾ Angielski ▾

Kalorie Kod produktu \*

Cena standardowa Cena promocyjna

Waga g Dostępna ilość

Opis \*

Obrazek  No file chosen

Rysunek 4.26: Formularz dodawania nowego produktu.

**Aktualizacja informacji produktu \* pole wymagane**

Nazwa \*  
łabędzi puch

---

Producent \*  
Nienazarty

Kraj \*  
Polska

---

Data dodania \*  
12-Jul-2018

Aktywny

Deser      alergeny      język

---

Kalorie  
525

Kod produktu \*  
3673

---

Cena standardowa  
7

Cena promocyjna  
5

---

Waga  
100

Dostępna ilość  
g 21

---

opis \*  
Krucze ciasto pokryte dżemem jagodowym, z masą serową. Na wierzchu puszysta śmietana pokryta wiórkami kokosowymi.

---

Obrazek  No file chosen

Rysunek 4.27: Formularz edytowania istniejącego produktu.

# Rozdział 5

## Podsumowanie

Celem pracy było stworzenie uniwersalnego systemu do obsługi placówek oferujących klientom swoje produkty. Głównym założeniem miały być zamówienia poprzez Internet i stworzenie systemu zarządzania treścią do kontroli menu aplikacji. Nie oznacza to jednak, że to tylko jedyna branża docelowa. System z powodzeniem może zostać przerobiony pod wszelkiego rodzaju sklepy. Podczas prac udało się zrealizować wszystkie aspekty teoretyczne jak i praktyczne. Projekt posiada warstwę widoku, bazę danych oraz interfejs. Został stworzony uniwersalny system, który w większości przypadków umożliwia tworzenie, odczytanie, aktualizację i usuwanie obiektów przechowywanych w bazie danych.

System jest w pełni edytowalny i wszystkie elementy mogą zostać skonfigurowane pod oczekiwania danego klienta. Oczywiście ta kwestia wymaga skorzystania z pomocy programisty, który byłby w stanie dokonać odpowiednich zmian w kodzie. Również kwestia panelu kontroli daje wielkie możliwości jeżeli chodzi o rozwój jej funkcjonalności. Wystarczy dodać kolejną zakładkę w menu lub nawigację z boku i stworzyć nowy kontroler funkcjonalności. Projekt można rozbudowywać w przyszłości. W przypadku zainteresowania rozwojem projektu, przykładową funkcjonalnością, którą warto dodać, może być wspomniana w pracy możliwość rezerwacji miejsc w lokalu lub czat służący do kontaktowania się z pracownikami baru lub restauracji.

# Spis rysunków

1.1	Strona internetowa Pizzeri Savona. . . . .	5
1.2	Schemat listy produktów. . . . .	6
1.3	Schemat konta użytkownika. . . . .	7
1.4	Schemat działania koszyka. . . . .	8
1.5	Schemat listy produktów po stronie administratora. . . . .	9
2.1	Przykładowa fotografia użyta w pracy. . . . .	17
2.2	Logo wykonane na potrzeby projektu. . . . .	18
2.3	Podział doświadczeń użytkownika. . . . .	19
4.1	Strona główna restauracji. . . . .	24
4.2	Formularz tworzenia nowego konta. . . . .	25
4.3	Formularz odzyskiwania hasła. . . . .	26
4.4	Formularz profilu użytkownika. . . . .	27
4.5	Zakładka z listą zamówień użytkownika. . . . .	28
4.6	Zakładka płatności z dostępną listą kart płatniczych. . . . .	29
4.7	Formularz dodawania/aktualizacji danych karty. . . . .	30
4.8	Informacje o karcie. . . . .	31
4.9	Zakładka dostaw z listą dostępnych adresów. . . . .	32
4.10	Formularz dodania/aktualizacji adresu dostawy. . . . .	33
4.11	Widok listy produktów. . . . .	34
4.12	Lista produktów z dodatkowymi przyciskami stronicowania. . . . .	35
4.13	Podgląd produktu. . . . .	36
4.14	Podgląd produktu ze statusem niedostępny. . . . .	37
4.15	Widok koszyka z dodanymi produktami. . . . .	38
4.16	Aktualizacja ilości produktu. . . . .	39
4.17	Komunikat potwierdzający aktualizację produktu. . . . .	40
4.18	Formularz podsumowania zamówienia. . . . .	41
4.19	Zakładka z informacjami płatności w podsumowaniu zamówienia. . . . .	42
4.20	Formularz zmiany danych do płatności. . . . .	43
4.21	Podgląd podsumowania i listy produktów w zamówieniu. . . . .	44
4.22	Komunikat potwierdzający dokonania zamówienia. . . . .	45
4.23	Potwierdzenie zamówienia wysyłane na adres e-mail użytkownika. . . . .	46
4.24	Portal administratora z listą produktów. . . . .	47
4.25	Podgląd produktu w panelu. . . . .	48



---

4.26 Formularz dodawania nowego produktu. . . . .	49
4.27 Formularz edytowania istniejącego produktu. . . . .	50

# Bibliografia

- [1] Tomasz Jakut, *JavaScript: programowanie zaawansowane*, Wydawnictwo Helion, 2016.
- [2] Syed Fazle Rahman, *Bootstrap: tworzenie interfejsów stron WWW*, Wydawnictwo Helion, 2015.
- [3] Nathan Rozentals, *Język TypeScript. Tajniki kodu*, Wydawnictwo Helion, 2017.
- [4] Steven M. Schafer, *HTML, XHTML i CSS: Biblia*, Wydawnictwo Helion, 2011.
- [5] Herbert Schildt, *Java. Kompendium programisty. Wydanie IX*, Wydawnictwo Helion, 2015.
- [6] Witold Wrotek, *CSS3: zaawansowane projekty*, Wydawnictwo Helion, 2015.
- [7] Angular, *Witryna internetowa*, <https://angular.io/>, stan na 25 czerwca 2018.
- [8] GIMP, *Witryna internetowa*, <https://www.gimp.org/>, stan na 12 czerwca 2018.
- [9] IntelliJ, *Witryna internetowa*, <https://www.jetbrains.com/phpstorm/>, stan na 12 czerwca 2018.
- [10] Spring dokumentacja oraz przykłady, *Witryna internetowa*, <https://spring.io/>, stan na 25 czerwca 2018.
- [11] Tomcat, *Witryna internetowa*, <http://tomcat.apache.org/>, stan na 12 czerwca 2018.
- [12] User experience, *Witryna internetowa*, [https://pl.wikipedia.org/wiki/User\\_experience](https://pl.wikipedia.org/wiki/User_experience), stan na 12 czerwca 2018.