

UNIwersytet WarMińsko-Mazurski
W OLSZTYNIE
WYDZIAŁ MATEMATYKI I INFORMATYKI

Kierunek: Informatyka

Przemysław Dzienisiewicz

Aplikacja do robienia sprawnych zakupów

Praca inżynierska wykonana
w Katedrze Metod Matematycznych
Informatyki
pod kierunkiem
dr Pawła Drozdy

Olsztyn 2021

**UNIVERSITY OF WARMIA AND MAZURY
IN OLSZTYN
FACULTY OF MATHEMATICS AND COMPUTER
SCIENCE**

Field of study: Computer Science

Przemysław Dzienisiewicz

Efficient shopping application

Engineering Thesis written in
Department of Mathematical Methods
in Computer Science
under supervision of
dr Paweł Drozda

Olsztyn 2021

Spis treści

Spis treści

| | |
|--|----|
| 1. Wstęp..... | 4 |
| 2. Opis technologii | 6 |
| 3. Struktura pracy..... | 10 |
| 3.1. Schemat bazy danych | 10 |
| 3.2. Diagram przypadków użycia | 11 |
| 3.2.1. Uogólniony diagram przypadków użycia | 11 |
| 3.2.2. Szczegółowy diagram przypadków użycia | 12 |
| 4. Dokumentacja użytkownika | 13 |
| Ekran rejestracji/logowania do aplikacji | 13 |
| Ekran aktualnej listy zakupów..... | 13 |
| Ekran bazy produktów użytkownika..... | 13 |
| Ekran wyszukiwania restauracji..... | 13 |
| Ekran zarządzania kontem | 13 |
| 5. Podsumowanie | 14 |
| Streszczenie | 14 |
| Abstract | 14 |
| Bibliografia..... | 15 |
| Spis Tabel..... | 15 |
| Spis zdjęć | 15 |
| Spis diagramów | 15 |

1. Wstęp

Tematem pracy inżynierskiej jest aplikacja mobilna o nazwie „Shopping App”, która powstała w celu usprawnienia zakupów. Użytkownik aplikacji jest w stanie znacząco przyspieszyć swoje zakupy dzięki zaplanowaniu kupna potrzebnych mu produktów poprzez wprowadzenie ich do aplikacji na listę przed wyruszeniem do sklepu. Aby zakupy przebiegały jeszcze sprawniej, w trakcie może odznaczać zakupione już artykuły. Posiadacz aplikacji ma bazę swoich produktów zawsze pod ręką - i jeśli chce – może swobodnie wracać do wcześniej dodanych artykułów, nawet gdy w aktualnej liście nie będzie ich potrzebował.

Aplikacja została stworzona przez Przemysława Dzienisiewicza, który był odpowiedzialny za całokształt. Motywacją do jej powstania była istota poszerzenia wiedzy z zakresu javascriptowej biblioteki ‘React Native’ służącej do tworzenia interfejsów użytkownika, która współdzieli zawartość samego ‘React’, umożliwiając programistom korzystanie z frameworka wraz z natywną platformą (tzn. konkretnym systemem operacyjnym tj. android, iOS). Oprócz samej nauki, wyszła potrzeba posiadania aplikacji na własny użytek, która pozwoli na sprawniejszą realizację zakupów - uporządkowania samej listy produktów oraz przede wszystkim skrócenia czasu potrzebnego na jej realizację, szczególnie w aktualnym czasie pandemii, przez co użytkownik może zminimalizować potencjalny kontakt z ludźmi przebywającymi w danym sklepie.

Tworzenie aplikacji nie byłoby możliwe bez możliwości podglądu aktualnego stanu wizualnego oraz fizycznego sprawdzenia jej działania. Za tę kwestię odpowiadał prywatny smartphone (Samsung S8) oraz środowisko ‘Expo’ odpalane na komputerze osobistym, które działało jako emulator urządzenia mobilnego

W aplikacji do stworzenia różnych ekranów użytkownika zostały użyte komponenty wielokrotnego użytku, które zapewnia zestaw z samej paczki biblioteki 'React Native' oraz 'React Native Elements'. Zbudowane w całości w javascript są spójnym zestawem dla wielu platform tj. android, iOS oraz przeglądarek. Zawierają predefiniowane style wielu komponentów (np. przyciski), toteż zdecydowanie ułatwia pracę nad tworzeniem aplikacji.

Na całym etapie tworzenia aplikacji używany był system kontroli wersji jakim jest 'GitHub', dzięki któremu możliwe było ciągłe śledzenie zmian w kodzie źródłowym, co usprawniło pracę z uwagi na potrzeby zmian, bądź powrotu do poprzednich wersji aplikacji, odtworzenia sprawnego stanu projektu lub porównania wprowadzonych zmian.

Pomocna okazała się również internetowa aplikacja 'Trello', czyli aplikacja internetowa do tworzenia list w stylu Kanban (tzn. metoda sterowania produkcją), która pozwalała na zwinne zarządzanie ustalonymi zadaniami oraz poszczególnymi etapami powstawania aplikacji od samego początku.

2. Opis technologii

React Native – główny filar na którym opiera się aplikacja. Jest to technologia stworzona przez firmę „Facebook” i służy do tworzenia wieloplatformowych aplikacji mobilnych (wspólny kod dla wszystkich platform - android oraz iOS), co zarazem jest jedną z jej największych zalet. Drugą jest natomiast to, iż całą aplikację piszemy w języku JavaScript – od logiki, aż po widoki.

Expo – środowisko ułatwiające pracę z React Native. W tym projekcie używane było jak emulator urządzenia mobilnego, pozwalającego na pokazywanie efektów pracy w czasie rzeczywistym podczas programowania. W jej szereg wchodzi kilka ważnych rzeczy, tj.:

- Aplikacja desktopowa, która umożliwia budowanie aplikacji, debugowanie
- Aplikacja mobilna, którą można zainstalować na dowolnym urządzeniu mobilnym z androidem lub iOS i odpalić przez nią tworzoną aplikację.
- Biblioteki expo, dzięki którym zapewnia wiele pomocniczych rozwiązań ułatwiających tworzenie aplikacji (np. obsługa powiadomień)

JSX – (JavaScript XML) to format zapisu kodu HTML oraz XML wewnątrz języka JavaScript. Pierwotnie został zaproponowany przez Facebook’a i użyty w ich bibliotece ‘React.js’. Aktualnie korzystają z niego również inne biblioteki. Zalicza się jako ‘lukier składniowy’ (ang. ‘syntactic sugar’), czyli cecha składni języka, którą można wyeliminować przez proste przekształcenia składniowe, istniejąca jedynie dla wygody programisty podczas pracy.

CSS3 – (ang. Cascading Style Sheets) tłumaczone jako kaskadowe arkusze stylów – język używany do opisu prezentacji dokumentu napisanego w języku znaczników, takim jak HTML (nadawanie stylów). Służy do kontrolowania wyglądu poszczególnych elementów na stronie lub aplikacji (tj. położenie obrazów, tekstu). Jedna z trzech podstawowych technologii WWW – obok JavaScript oraz HTML.

HTML5 – język znaczników, którego używa się do strukturyzowania i nadawania znaczenia naszej zawartości internetowej, na przykład definiowania akapitów, nagłówków oraz tabel danych.

JavaScript – (JS) język programowania zgodny ze specyfikacją ECMAScript, który umożliwia implementację złożonych funkcji na stronach internetowych – odpowiada min.in. za dynamiczne aktualizowanie treści, sterowanie multimediami oraz interaktywne funkcje. Jest językiem wysokopoziomowym, często kompilowanym na czas oraz wieloparadygmatycznym.

MongoDB – jest otwartym, nierelacyjnym systemem zarządzania bazą danych. Jego głównymi zaletami są duża skalowalność, wydajność oraz brak ściśle zdefiniowanej struktury obsługiwanych baz danych. Wykorzystuje dokumenty podobne do JSON z opcjonalnymi schematami.

Visual Studio Code – darmowy, usprawniony edytor kodu z obsługą operacji programistycznych tj. debugowanie, uruchamianie zadań i kontrola wersji stworzony przez firmę Microsoft dla Windows, Linux oraz macOS. Jest w pełni konfigurowalny pod potrzeby użytkownika.

Android Studio – program zapewniający ujednolicone środowisko, w którym można tworzyć aplikacje na telefony i tablety z systemem android.

Pozwala na uruchomienie urządzenia wirtualnego systemu android (emulator) do testowania i debugowania aplikacji.

Git Bash - aplikacja terminalowa dla środowisk Microsoft Windows, która zapewnia warstwę emulacji dla obsługi poleceń Git. Pozwalała na łączenie się z systemem operacyjnym za pomocą pisemnych poleceń.

GitHub – platforma hostingu kodu do tworzenia oprogramowania oraz kontroli wersji za pomocą Git. Oferuje funkcje rozproszonej kontroli wersji i zarządzania kodem źródłowym Git, a także własne funkcje – tj. śledzenie błędów, zarządzanie zadaniami, ciągłą integrację. Github ułatwił szczególnie pracę nad aplikacją w przypadku programowania na dwóch różnych urządzeniach – wymiana aktualnym kodem źródłowym była bardzo sprawna.

Yelp – serwis społecznościowy umożliwiający użytkownikom publikowanie recenzji i ocenianie firm (tj. restauracje, butiki, salony). Przedsiębiorstwa mogą aktualizować swoje dane kontaktowe, ofertę oraz zdjęcia usług i towarów. W Polsce głównymi miejscami ocenianymi w serwisie są Warszawa i Kraków. Umożliwił zaimplementowanie wyszukiwania oraz uzyskiwania informacji o restauracji w aplikacji.

Nodemon – narzędzie, które pomaga w tworzeniu aplikacji opartych na node.js poprzez automatyczne ponowne uruchamianie aplikacji 'node' po wykryciu zmian plików w katalogu. Umożliwia płynną pracę podczas działania z emulatorem środowiska aplikacji mobilnej.

Ngrok – prosta w obsłudze, wieloplatformowa aplikacja, która umożliwia udostępnienie lokalnego serwera programistycznego w Internecie. Oprogramowanie sprawia, że lokalnie hostowany serwer WWW jest

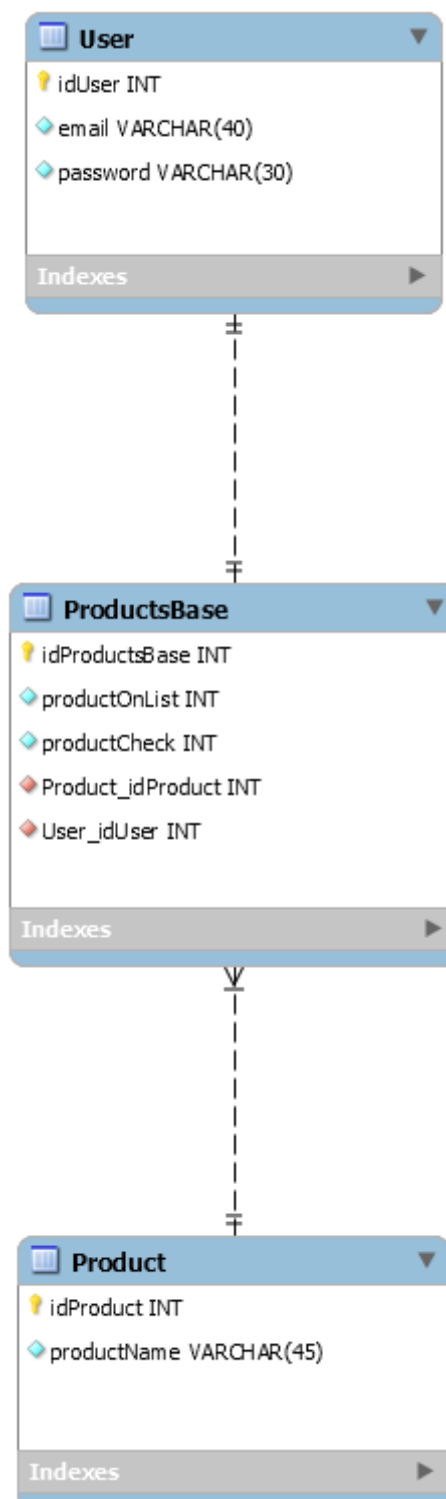
hostowany w poddomenie ngrok.com, co oznacza, że nie jest potrzebny publiczny adres IP ani nazwa domeny na komputerze lokalnym. Pozwolił na darmową pracę z aplikacją mobilną i połączenie do bazy danych w MongoDB.

Trello – internetowa aplikacja do tworzenia list w stylu Kanban, pozwalająca na zwinne zarządzanie zadaniami.

Windows PowerShell – wieloplatformowy, rozbudowany interpreter poleceń do automatyzacji zadań i zarządzania konfiguracją, składający się z powłoki wiersza poleceń i języka skryptowego.

3. Struktura pracy

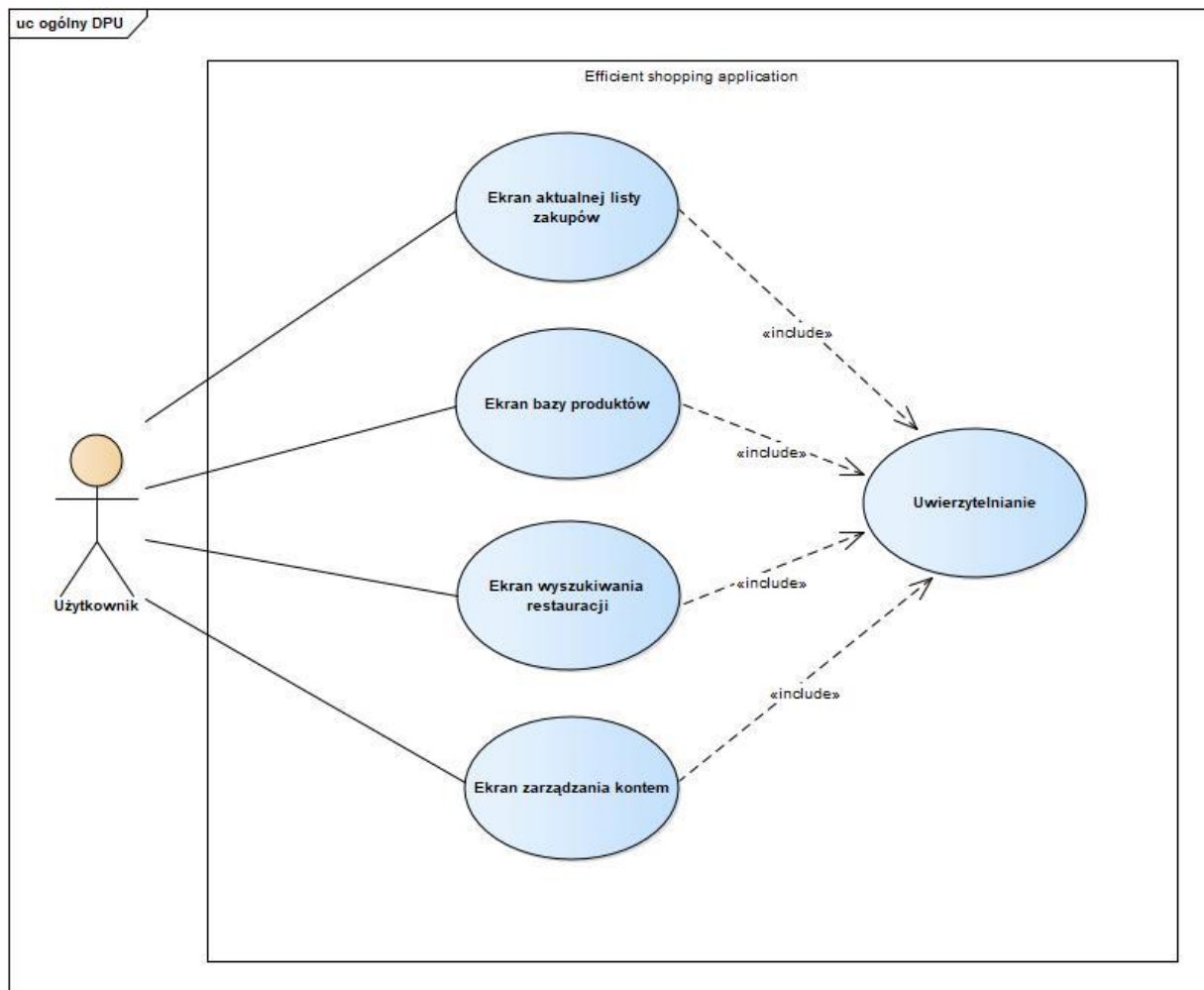
3.1. Schemat bazy danych



3.2. Diagram przypadków użycia

3.2.1. Uogólniony diagram przypadków użycia

Diagram przypadków użycia



3.2.2. Szczegółowy diagram przypadków użycia

Diagram przypadków użycia

4. Dokumentacja użytkownika

Ekran rejestracji/logowania do aplikacji

Ekran aktualnej listy zakupów

Ekran bazy produktów użytkownika

Ekran wyszukiwania restauracji

Ekran zarządzania kontem

5. Podsumowanie

Celem stworzenia aplikacji, która jest tematem tejże pracy inżynierskiej, było usprawnienie robienia zakupów. Użytkownik aplikacji jest w stanie znacząco przyspieszyć swoje zakupy dzięki zaplanowaniu potrzebnych mu produktów i wprowadzeniu ich na listę w aplikacji przed wyruszeniem do sklepu. Aby zakupy przebiegały jeszcze sprawniej, w trakcie może odznaczać zakupione już artykuły. Osobiście dla mnie było to dużym wyzwaniem, ponieważ był to pierwszy samodzielny projekt - od początku do końca - z wykorzystaniem biblioteki 'React Native'. Ostatecznie jestem zadowolony z wyniku pracy oraz tego jak dużo doświadczenia wniosła w moją przyszłą karierę developerską.

Streszczenie

Celem pracy inżynierskiej jest stworzenie aplikacji mobilnej do sprawniejszego robienia zakupów, która będzie działać na obu systemach operacyjnych – android oraz iOS. Ma spełniać podstawowe zadanie, czyli umożliwić użytkownikowi skrócenie robienia zakupów - poprzez zaplanowanie jakie produkty chce kupić, wprowadzeniem ich do aplikacji na listę przed udaniem się do sklepu oraz odznaczanie zakupionych już artykułów.

Abstract

The aim of the engineering work is to create a mobile application for more efficient shopping, which will work on both operating systems - Android and iOS. It is supposed to fulfill the basic task, in other words to enable the user to shorten the shopping process - by planning what products he wants to buy, entering them into the application on the list before going to the store and checking off already purchased items.

Bibliografia

- [1] <https://reactnative.dev/docs/getting-started>
- [2] <https://docs.expo.io>
- [3] <https://developer.mozilla.org/en-US/docs/Glossary/JavaScript>
- [4] <https://developer.mozilla.org/en-US/docs/Glossary/HTML>
- [5] <https://www.npmjs.com/package/nodemon>
- [6] <https://pl.wikipedia.org/wiki/MongoDB>
- [7] https://en.wikipedia.org/wiki/Visual_Studio_Code
- [8] <https://ngrok.com/docs>
- [9] <https://www.atlassian.com/git/tutorials/git-bash>
- [10] <https://www.yelp.pl/about>
- [11] <https://guides.github.com/activities/hello-world/>
- [12] <https://reactjs.org/docs/introducing-jsx.html>
- [13] <https://help.trello.com/article/708-what-is-trello>
- [14] <https://docs.microsoft.com/en-us/powershell/scripting/overview?view=powershell-7.1>
- [15] <https://developer.android.com/studio/intro>
- [16] https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics

Spis Tabel

Spis zdjęć

Spis diagramów