

Aplikacja webowa do losowania jedzenia

Jira: <https://projektzespolygr17.atlassian.net/jira/software/projects/PROJ/boards/1>

Github: https://github.com/miKopciowski/Projekt_Zespolywy

Główny opis projektu

Celem projektu jest stworzenie aplikacji webowej polegającej na wylosowaniu jednej pozycji z całego menu lub paru wybranych dań danej restauracji. Losowanie będzie można również przeprowadzać wpisując kwotę, którą chcemy przeznaczyć oraz czy chcemy napój również będzie można zaznaczyć jakich składników nie lubimy bądź mamy alergie żeby nie było możliwości wylosowania dania które posiada dane składniki. Aplikacja będzie posiadać w bazie danych parę najpopularniejszych restauracji z możliwością ręcznego dodania menu lokalu którego nie ma w bazie.

Zaletami używania aplikacji mogą być szybszy wybór zamówienia, pewnego rodzaju zabawa polegająca na testowaniu posiłków, których normalnie byśmy nie zamówili. Dla osób lubiących nutkę hazardu będzie to aplikacja idealna.

Motywacja

Pomysł na ten projekt i jego motywacja pojawiły się podczas burzy mózgów jaką wykonaliśmy podczas pierwszego sprintu. Stwierdziliśmy, że każdy z nas natrafia na problem z wyborem jedzenia w restauracji i często tracimy 5 lub 10 min nawet w fast foodach na wybór jedzenia między opcją A i opcją B. Tak wpadliśmy na pomysł aby ten problem za nas rozwiązała aplikacja, i żeby to na niej spoczywał obowiązek wybrania nam jedzenia.

Cel projektu

Celem projektu jest stworzenie aplikacji webowej, która będzie pomagała niezdecydowanemu użytkownikowi wybrać produkty z restauracji. Ma być ona przejrzysta, prosta w obsłudze, tak aby mógł się nią posłużyć każdy potencjalny użytkownik. Aplikacja będzie posiadała dostęp do menu wielu restauracji, które będzie wyświetlane w przystępny sposób w witrynie internetowej umożliwiając wybór poszczególnych produktów, spośród których będzie losowane finalne gotowe do zamówienia danie. Projekt ma być zrealizowany do końca semestru letniego 2021/2022.

Zespół

Mikołaj Kopciowski - Lider

- Planowanie sprintów oraz przestrzeganie terminów
- Zarządzanie taskami oraz przydziałem zadań
- Prezentacja finalnego projektu
- Projektowanie UI/UX

Wojciech Bilski - Frontend

- Wybranie technologii
- Odpowiedzialny za stworzenie Front-endu
- Odpowiedzialny za hostowanie Front-endu

Bartek Grzywaczewski - Backend

- Stworzenie schematów bazy oraz UML
- Stworzenie bazy danych
- Odpowiedzialny za stworzenie back-endu
- Odpowiedzialny za hosting back-endu

Michał Muciek - Tester

- Odpowiedzialny za stworzenie testów aplikacji
- Odpowiedzialny za sprawdzenie wszystkich zadań w projekcie
- Odpowiedzialny za przetestowanie aplikacji

Czas wykonania projektu

Czas trwania projektu: 8.10.2021 - 22-06-2022

Długość sprintów: 2 tygodnie

Daily: co trzy dni o 19:00

Sprint planning: Poniedziałek 18:00

Sprint review: Niedziela 20:00

Zadania

- PROJ-18 Stworzenie zadań dokumentacji
- PROJ-78 Prezentacja swojej części projektu na spotkaniu
- PROJ-80 Prezentacja swojej części projektu na spotkaniu
- PROJ-79 Prezentacja swojej części projektu na spotkaniu
- PROJ-81 Prezentacja swojej części projektu na spotkaniu
- PROJ-69 Ponowne przeczytanie dokumentacji oraz dodanie komentarzy w przypadku uwag
- PROJ-70 Ponowne przeczytanie dokumentacji oraz dodanie komentarzy w przypadku uwag
- PROJ-71 Ponowne przeczytanie dokumentacji oraz dodanie komentarzy w przypadku uwag
- PROJ-72 Ponowne przeczytanie dokumentacji oraz dodanie komentarzy w przypadku uwag
- PROJ-73 Dostarczenie opisów wcześniej wykonanych zadań w celu określenia punktacji

- PROJ-74 Dostarczenie opisów wcześniej wykonanych zadań w celu określenia punktacji
- PROJ-75 Dostarczenie opisów wcześniej wykonanych zadań w celu określenia punktacji
- PROJ-76 Dostarczenie opisów wcześniej wykonanych zadań w celu określenia punktacji
- PROJ-77 Określenie punktacji wszystkich dotychczas wykonanych zadań
- PROJ-60 Stworzenie diagramu aktywności
- PROJ-61 Stworzenie diagramu klas
- PROJ-62 Stworzenie diagramu sekwencji
- PROJ-63 Uzupełnienie: Czas wykonania projektu
- PROJ-64 Napisanie motywacji dlaczego wybieramy Jenkinsa
- PROJ-65 Opracowanie scenariuszy integracyjnych
- PROJ-66 Opracowanie scenariuszy regresyjnych
- PROJ-67 Uzupełnienie: nasz zespół
- PROJ-68 Spis kroków milowych
- PROJ-37 Główne cele projektu [DOKUMENT]
- PROJ-29 Kolory
- PROJ-45 Stworzenie projektu w figmie
- PROJ-47 Zaprojektowanie całej aplikacji
- PROJ-48 Zaprojektowanie ruchu po aplikacji
- PROJ-53 Stworzenie diagramu USE-CASE
- PROJ-54 Stworzenie diagramu bazy
- PROJ-55 Opracowanie scenariuszy testów manualnych
- PROJ-56 Zapoznanie się z CI CD Jenkins i przedstawienie go grupie
- PROJ-39 Stworzenie wstępnej dokumentacji
- PROJ-40 Główny Opis projektu
- PROJ-41 Motywacja projektu
- PROJ-42 Cel Projektu
- PROJ-43 Motywacja wyboru technologii
- PROJ-27 Potrzebne funkcjonalności
- PROJ-28 Rozmieszczenie na stronie
- PROJ-22 Konfiguracja Wojtek
- PROJ-30 Znalezienie terminala z Linuxa na windowsa i przeprowadzenie nas przez instalacje
- PROJ-23 Konfiguracja Mikołaj
- PROJ-24 Konfiguracja Bartek
- PROJ-25 Konfiguracja Michał
- PROJ-31 Podstawy NodeJS
- PROJ-33 Stworzenie GitHuba
- PROJ-34 Zapoznanie się z tematem testowania aplikacji webowych
- PROJ-35 Sposób działania
- PROJ-14 Wybór technologii
- PROJ-13 Plan na aplikacje
- PROJ-5 Stworzenie JIRy i zapoznanie się z jej strukturą
- PROJ-6 Przydzielenie początkowych tasków

Milestones

Podczas całego projektu wykonamy 6 milestone. Długość milestone jest indywidualna i zależna od naszej wiedzy w danym milestone.

Semestr 1:

[M1] 1.10.2021 - 22.11.2021

Stworzenie Jiry, Wybranie ról w projekcie, Wstępny zarys dokumentacji oraz zarys aplikacji

[M2] 22.11.2021 - 19.12.2021

Workflow, Testy oraz UML

[M3] 19.12.2021 - XX.01.2022

Dokończenie dokumentacji i przygotowanie dokumentu na przyszły semestr

Semestr 2:

[M4] 25.01.2022 - 21.02.2022

Przygotowanie wszystkich środowisk. Poprawnie działające CI/CD Jenkins

[M5] 22.02.2022 - 30.04.2022

MVP Aplikacji

[M6] 01.05.2022 - 22.06.2022

Poprawienie błędów aplikacji oraz zahostowanie

Epiki

- Przygotowanie zadań na pierwszy semestr
- Przygotowanie JIRY
- Konfiguracja ReactJS + NodeJS
- UX/UI Wizja i pomysły
- Dokumentacja
- Design UX UI
- UML
- Testy
- Dokończenie Dokumentacji

Testowanie

Testy dostępne są pod poniższym adresem:

https://github.com/miKopciowski/Projekt_Zespolowy/blob/main/testy_manualne.txt

Motywacja wybranej technologii

Zdecydowaliśmy się na zrobienie aplikacji webowej. Głównym powodem naszego wyboru jest możliwość korzystania z aplikacji na każdym urządzeniu z przeglądarką internetową i dostępem do Internetu. Aplikacja będzie działała zarówno na komputerach, jak i urządzeniach mobilnych, bez konieczności instalowania dodatkowego oprogramowania.

Wybraliśmy **ReactJS** na front-end, ponieważ nadaje się do tworzenia dobrze wyglądających i działających aplikacji internetowych. React jest używany na stronach internetowych znanych firm np. Facebook, Dropbox, Khan Academy, Netflix, Airbnb, PayPal... Dodatkowo React cały czas zyskuje na popularności, dzięki czemu łatwo znaleźć materiały do nauki.

Wybraliśmy **NodeJS** na back-end, ponieważ dobrze współpracuje z ReactJS. Pisanie frontendu i backendu w Javascript pozwala na lepszą komunikację w zespole i łatwiejsze łączenie ze sobą funkcji. NodeJS nadaje się do tworzenia mikroserwisów, jest dobry wydajnościowo i w prosty sposób łączy się z bazami danych.

Jenkins

Ciągła integracja (ang. Continuous integration), czyli ciągle szukanie błędów.

Inaczej Continuous Delivery (CD). Po fazie integracji i przetestowaniu programu, może nastąpić automatyczna faza dostarczenia. Polega ona na przygotowaniu aplikacji do instalacji, spakowaniu i umieszczeniu w centralnym repozytorium.

Jenkins to wieloplatformowa aplikacja-serwer open source do kontrolowania błędów oparty na pluginach.

Jednym z popularniejszych narzędzi CI/CD jest Jenkins, który pozwala na zautomatyzowanie procesu budowania np. uruchamia test, buduje projekt albo zbiera informacje o przeprowadzonych operacjach poprzez skonfigurowanie zadań. Zadania można ustawić w różny, odpowiedni dla nas sposób. W przypadku aplikacji opartych o nowoczesne frameworki jak React, pierwszym zadaniem będzie odpalenie Composera w celu zaktualizowania paczek, uruchomienie webpacka w celu kompilacji plików JS i CSS. Następnie Jenkins uruchomi niezbędne testy jednostkowe, które pozwalają przetestować poszczególne atomowe części aplikacji i wykryć błędy we wczesnej fazie.

Jenkins dobrze współpracuje z githubem i po odpowiednim ustawieniu Jenkins może po każdym nowym commicie automatycznie pobierać kod, skompilować i testować program po czym zgłaszać błędy poprzez np. wiadomość e-mail, lub w aplikacji mobilnej.

więcej informacji:

<https://www.jenkins.io/>

<https://szkolajenkinsa.pl/2021/02/07/czym-jest-jenkins/>

Motywacja Jenkins

Jenkins to profesjonalne narzędzie, które w ciągły i automatyczny sposób sprawdza integralność wszystkich elementów systemu i natychmiast informuje o znalezionych błędach.

Wybieramy właśnie to narzędzie, ponieważ Jenkins doskonale nadaje się do testowania aplikacji webowych.

Jenkins poprawnie wspiera i testuje aplikacje tworzone w ReactJS w połączeniu z NodeJS.

Ciągłe sprawdzanie występowania błędów pozwala na optymalną i szybką pracę zespołu.

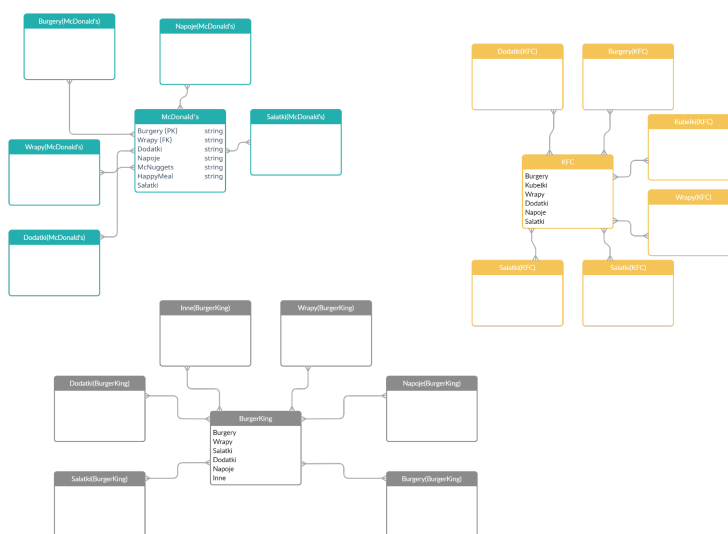
Ważną cechą jest dla nas to, że Jenkins może być uruchomiony na serwerze w kontenerze Docker.

Oznacza to, że tester nie musi uruchamiać narzędzia przy każdej zmianie w kodzie.

Jenkins posiada dużo dokumentacji, poradników i jest używany przez bardzo wiele firm.

Dokumentacja techniczna

Projekt bazy danych



Baza danych została zaprojektowana w sposób ułatwiający podział menu restauracji na kategorie oraz umożliwia łatwe dodanie pozycji do danej kategorii. Baza danych utworzona w MySQL zbliżona jest do Oracle SQL. MySQL stanowi dogodniejszy dla naszych potrzeb wybór ze względu na brak dodatkowych opłat dla administratora bazy a także ze względu na płynność tak istotną dla baz typu read-only.

Diagram aktywności

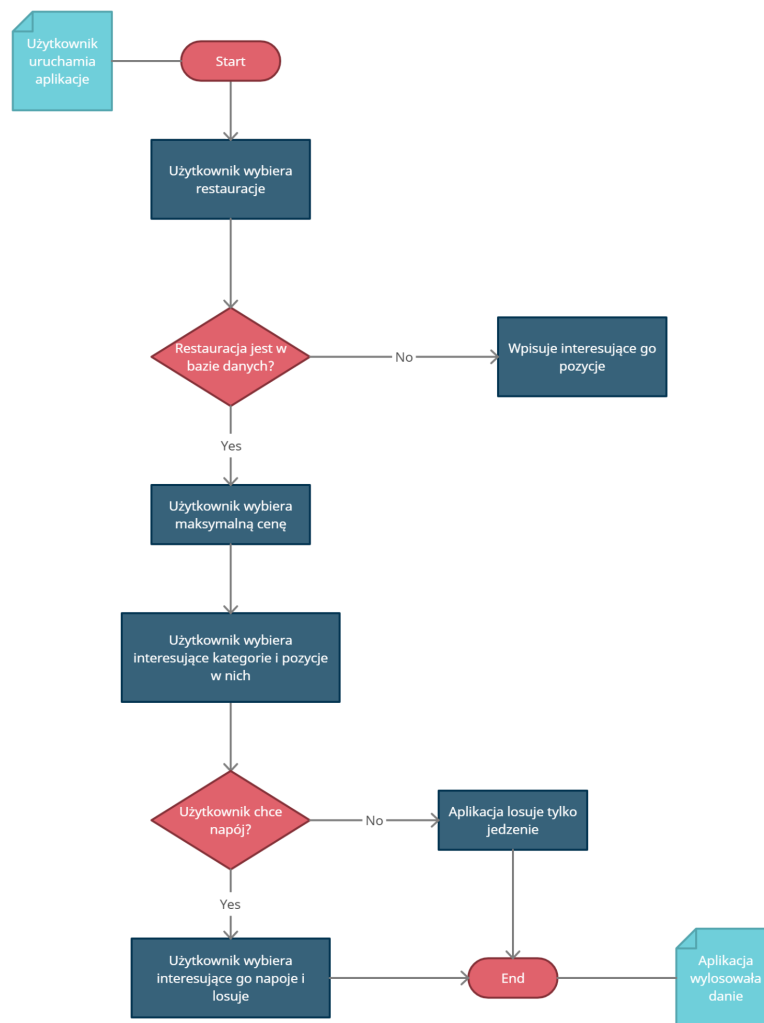


Diagram Klas

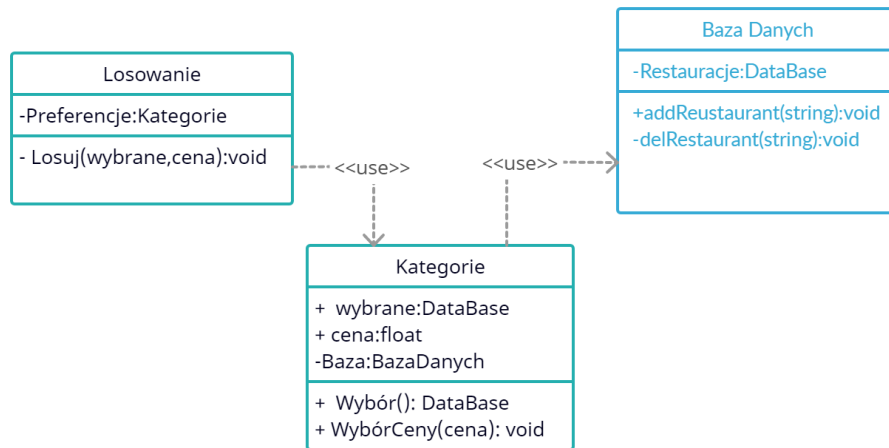
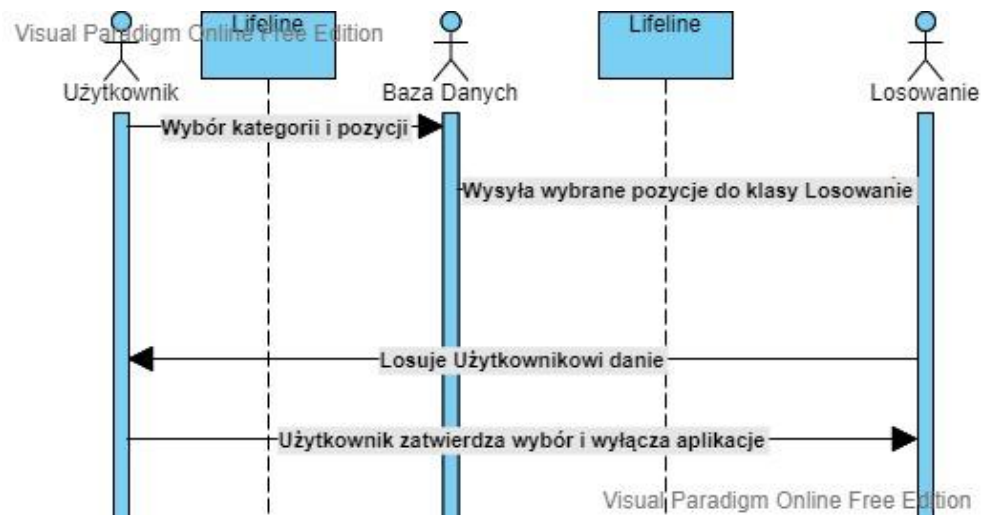
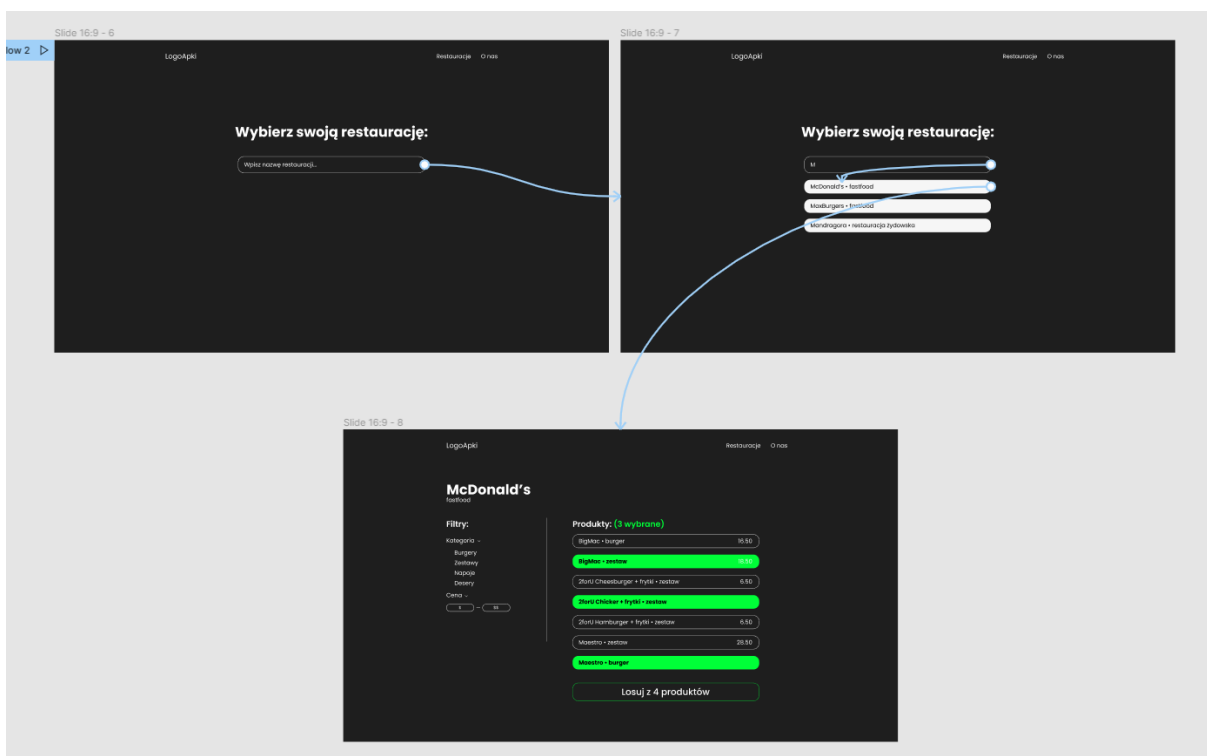
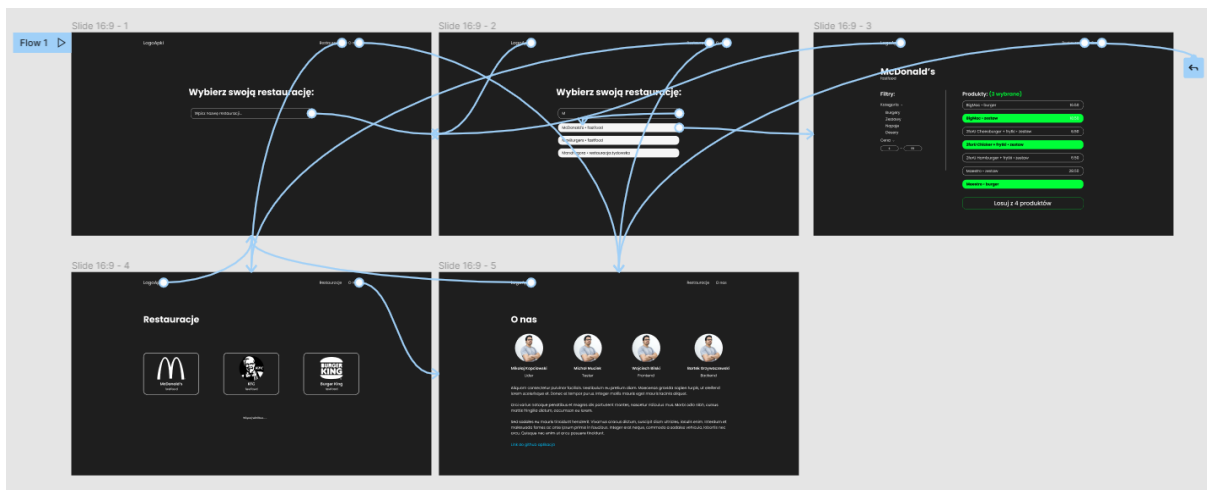


Diagram sekwencji



GUI

Interface całej aplikacji wraz z poruszaniem się po niej



Poszczególne ekrany projektu:

Wybierz swoją restaurację:

Wybierz swoją restaurację:

McDonald's • fastfood

MaxBurgers • fastfood

Mandragora • restauracja żydowska

McDonald's

fastfood

Filtry:

Kategoria ▾

Burgery

Zestawy

Napoje

Desery

Cena ▾

\$ — \$

Produkty: (3 wybrane)

BigMac • burger	16.50
BigMac • zestaw	18.50
2forU Cheesburger + frytki • zestaw	6.50
2forU Chicker + frytki • zestaw	
2forU Hamburger + frytki • zestaw	6.50
Maestro • zestaw	28.50
Maestro • burger	

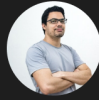
Losuj z 4 produktów

Restauracje

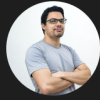


Więcej wkrótce.....

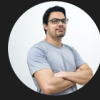
O nas



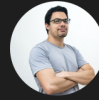
Mikołaj Kopciowski
Lider



Michał Muciek
Tester



Wojciech Biłski
Frontend



Bartek Grzywaczewski
Backend

Aliquam consectetur pulvinar facilisis. Vestibulum eu pretium diam. Maecenas gravida sapien turpis, ut eleifend lorem scelerisque et. Donec et tempor purus. Integer mollis mauris eget mauris lacinia aliquet.

Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Morbi odio nibh, cursus mattis fringilla dictum, accumsan eu lorem.

Sed sodales eu mauris tincidunt hendrerit. Vivamus a lacus dictum, suscipit diam ultricies, iaculis enim. Interdum et malesuada fames ac ante ipsum primis in faucibus. Integer erat neque, commodo a sodales vehicula, lobortis nec arcu. Quisque nec enim ut arcu posuere tincidunt.

[Link do github aplikacji](#)

Hosting

Nasza aplikacja zostanie hostowana na Microsoft Azure