

Wprowadzenie do zarządzania projektami deweloperskimi

laboratorium



Karpiński Maciej
Krysa Marcin
Kuczma Łukasz
Mertuszka Adam

Prowadzący mgr inż. Marcin Tracz

Spis treści

1	Opis projektu	3
2	Role w grupie	4
3	Wymagania i założenia projektowe	5
4	Opis działania	6
5	Wykorzystana technologia i narzędzia	7
6	Event storming	8
7	Backlog	9
8	Estymata	10
9	Sprinty	11
10	Interfejs aplikacji	12
11	Uruchomienie w środowisku developerskim	13
12	Podsumowanie	14

1 Opis projektu

Projekt nazwaProjektu jest aplikacją internetową przeznaczoną dla pasjonatów warzących piwo w domu, którzy chcieliby posiadać listę posiadanych składników wraz z terminami ważności.

2 Role w grupie

3 Wymagania i założenia projektowe

Wymagania projektowe:

- Uwierzytelnianie (w tym poprzez social media);
- Przechowywanie różnych typów produktów;
- Powiadomienia;
- Poręczny interfejs;
- Separacja backendu i frontendu;
- Logi.

Założenia:

- Użytkownik może tworzyć swoje własne schowki;
- Użytkownik wybiera składnik z listy składników i dodaje go do swojego schowka;
- Użytkownik wprowadza termin ważności składnika;
- Użytkownik dostaje powiadomienie o zbliżającym się terminie ważności;
- Przyjazny interfejs użytkownika;

4 Opis działania

5 Wykorzystana technologia i narzędzia

skopiować finalny

6 Event storming

Event Storming został przeprowadzony na portalu „Miro” <https://miro.com/>, w dniu 05.12.2020 było to nowe i ciekawe doświadczenie dla każdego z członków zespołu. Projekt w momencie przeprowadzenia event stormingu był już w zaawansowanej formie, wraz z rozpisanyimi zadaniami więc wykorzystaliśmy spotkanie do stworzenia modelu pracy aplikacji. Podczas tworzenia wizualizacji okazało się, że rozumienie aspektów działania i reagowania aplikacji na niektóre zdarzenia różnią się między poszczególnymi członkami zespołu, ujawnione różnice w wyobrażeniu na temat funkcjonowania poszczególnych komponentów aplikacji udało się wybrać najlepsze rozwiązania. Podczas spotkania wyraźnie było widać brak doświadczenia w tego typu zadaniu, na początku często przerywaliśmy sobie wypowiedzi ale dość szybko doszliśmy do porozumienia i zapanował porządek. Wspólnie doszliśmy do wniosku że było to ciekawe doświadczenie, chodź lepszym byłoby spotkanie „twarzą w twarz” wraz z osobą doświadczoną, która mogłaby poprowadzić takie spotkanie. W miarę rozwoju aplikacji do modelu aplikacji były dodawane kolejne szczegóły. Wynikiem event stormingu jest spójny model działania aplikacji, dzięki czemu każdy członek zespołu może prześledzić przepływ informacji.

7 Backlog

8 Estymata

9 Sprinty

10 Interfejs aplikacji

11 Uruchomienie w środowisku developerskim

Repozytorium kodu projektu i jego dokumentacja znajdują się w serwisie „GitHub” pod adresem:

<https://github.com/MacKarp/Homebrewing-storage>

Do uruchomienia systemu wymagana jest działająca platforma konteneryzacji „Docker” oraz narzędzie „Docker Compose” szczegóły instalacji i konfiguracji wymaganych komponentów, można sprawdzić na stronie internetowej: <https://docs.docker.com/>. Po ściągnięciu repozytorium w katalogu głównym projektu znajdują się pliki wsadowe, umożliwiające kompilację i uruchomienie „frontendu” i „backendu” wraz z bazą danych. Pliki wsadowe dla systemu Windows:

- Backend-Start.bat
- Backend-Start-DEV.bat
- Backend-Stop.bat
- Backend-Stop-DEV.bat

Pliki dla systemu Linux:

- Backend-Start.sh
- Backend-Start-DEV.sh
- Backend-Stop.sh
- Backend-Stop-DEV.sh

Pliki „Backend-Start.bat”, „Backend-Stop.bat” oraz „Backend-Start.sh” i „Backend-Stop.sh” służą do uruchamiania i zatrzymywania „backendu” w wersji produkcyjnej, która pobierze i uruchomi najnowszą stabilną wersję „backendu” aplikacji i bazę danych z serwisu „Docker Hub”, w przypadku pliku z przyrostkiem „DEV” zostanie uruchomiona kompilacja wersji deweloperskiej znajdującej się w lokalnym katalogu „Backend” a następnie zostanie utworzony kontener z działającą aplikacją. W przypadku chęci zmiany ustawień „backendu” aplikacji lub kontenera należy edytować - zachowując odpowiednie formatowanie plików yaml - plik w wersji produkcyjnej lub deweloperskiej:

- docker-compose-backend.yaml
- docker-compose-backend-dev.yaml

12 Podsumowanie