

Dokumentacja Technicznej

Figuya Onrine

Dla Akademia Humanistyczno-Ekonomiczna w Warszawie

Kierunek: Informatyka

Sporządził:

Aleksander Michał Dmowski Numer albumu: 52662

Wraz z:

Paweł Aleksiejuk Numer albumu: 53847

Aleksandra Plewińska Numer albumu: 55262

Marcin Chamczyk Numer albumu: 53848

Karol Chwaciński Numer albumu: 55252

Rozpoczęta dnia 2024.03.08 r.

Miejsce ostatniego wpisu:

DEVLOG SPIS..... 9

Spis treści

WSTĘP	3
1. PRZEZNACZENIE I CEL	4
1.1 KONTEKST OGÓLNY	4
1.2 CEL	4
2. OPIS TECHNICZNY	5
2.1 TECHNOLOGIE	5
2.2 FRONTEND	5
2.3 BACKEND	6
2.4 BAZA DANYCH	6
2.5 WYMAGANIA	6
2.6 TESTY I WALIDACJA	6
3. PODSUMOWANIE	8
3.1 UJĘCIE KOŃCOWE	8
3.2 DAJSZY ROZWÓJ PROJEKTU	8
4. DEVLOG SPIS	9
4.1 DEVLOG	9

Wstęp

Niniejsza dokumentacja przedstawia techniczne aspekty w odniesieniu do utworzenia strony internetowej „Figuya-Onrine”, jak i czym „Figuya-Onrine” jest co jest powodowe powstania sklepu oraz jakie są jego cele. Dokumentacja ta jest tylko ujęciem ogólnym której celem jest wsparcie w kontroli jak i rozwoju projektu.

1. Przeznaczenie i cel

1.1 Kontekst ogólny

Projekt „Figuya-Onrine” to internetowy sklep dla handlarza limitowanymi figurkami kolekcjonerskimi. Dokumentacja przedstawia zarys technologiczny użyty przy tworzeniu trzonu strony internetowej.

1.2 Cel

Bazowym modelem sprzedażowym jest informowanie o tak zwanym dropie¹ (*pojawieniu się zbioru nowych figurek możliwych do sprzedania*) za pomocą discorda². Celem zaistnienia sklepu internetowego jest przede wszystkim umożliwienie sprzedaży produktów lub usług przez Internet. Daje to możliwość zwiększenia zasięgu i rynku zbytu na sprzedawanie figurek przez dotarcie do nowych klientów nieograniczających się do internetowych społeczności wspólnych zainteresowań, ale też do klientów szukających tego typu produktów w sieci. Sklep internetowy jest dostępny dla klientów przez całą dobę, co umożliwia im elastyczne zakupy w dowolnym czasie, niezależnie od godzin pracy właściciela sklepu i zawierania transakcji handlowych poprzez korespondencje na platformie discord. Utworzenie sklepu internetowego jest ważne w aspekcie skalowalności projektu dzięki, łatwości w dodawaniu nowych produktów lub opcji, bez konieczności ponoszenia dużych inwestycji. Pozwoli też analizować dane sprzedażowe w celu ustalenia potrzeb zakupowych klientów. Aspekty te w szerszej perspektywie przełożą się na zwiększenie konkurencyjności właściciela sklepu.

¹ Limitowana edycja produktów wprowadzanych na rynek w określonym momencie.

² Discord, multiplatformowe narzędzie do komunikacji ze społecznościami z danego grona zainteresowań lub powiązań,

2. Opis techniczny

2.1 Technologie

Stos technologiczny zastosowany w projekcie głównie z:

- **JavaScript** to język programowania używany głównie do tworzenia interaktywnych i dynamicznych elementów na stronach internetowych. Działa w przeglądarkach internetowych, pozwalając na manipulację HTML i CSS, reagowanie na zdarzenia użytkownika oraz komunikację z serwerami w tle. Dzięki swojej wszechstronności i szerokiemu wsparciu, jest kluczowym narzędziem dla web developerów.
- **Ruby** to wszechstronny, dynamiczny język programowania, który jest znany ze swojej prostoty. Jest używany do tworzenia aplikacji internetowych dzięki frameworkowi *Ruby on Rails*. Ruby kładzie duży nacisk na czytelność i łatwość pisanie kodu.
- **Docker Compose** to narzędzie umożliwiające definiowanie i uruchamianie wielokontenerowych aplikacji Dockerowych za pomocą pliku YAML³. Ułatwia zarządzanie zależnościami między usługami, automatyzując uruchamianie, konfigurację i skalowanie kontenerów w sposób zorganizowany i powtarzalny.
- **TypeScript** to statycznie typowany nadzbiór JavaScriptu, który kompiluje się do czystego JavaScriptu. Jego funkcjonalność to dodanie typów do języka, co pomaga w wykrywaniu błędów podczas kompilacji oraz ułatwia utrzymanie i rozwój dużych projektów. Dzięki temu można korzystać z zaawansowanych funkcji edytorów kodu, takich jak autouzupełnianie i refaktoryzacja, co zwiększa produktywność i jakość kodu. TypeScript jest w pełni kompatybilny z istniejącym kodem JavaScript i bibliotekami, co umożliwia stopniową migrację.
- **PostgreSQL** to zaawansowane, otwartoźródłowe środowisko do zarządzania bazami danych (DBMS⁴), znany ze swojej skalowalności i zgodności ze standardem SQL⁵. Oferuje zestaw funkcji, takich jak transakcje ACID, obsługa zaawansowanych typów danych, replikacja i rozszerzalność przez moduły. PostgreSQL jest szeroko stosowany w aplikacjach wymagających niezawodności, integralności danych i wydajności.

2.2 Frontend

³ Kim jest YAMAL

⁴

⁵

Składa się z TypeScripta. Został zastosowany ze względu na...

2.3 Backend

Gotowy Backend

2.4 Baza danych

Baza danych działa na bazie PostgreSQL, wybór padł z powodu

2.5 Wymagania

WSL

Docker

Repository cloned at WSL

IDE with WSL support (fe: vscode with extensions)

2.6 Testy i validacja

Podczas manualnego testowania strony metodą eksploracji wykryto następujące błędy:

Wyszukiwanie

Testy funkcjonalne:

- **Dodawanie produktów:** Dodano różne ilości produktów do koszyka i sprawdzono, czy liczba przedmiotów w koszyku aktualizuje się poprawnie.
- **Zmiana ilości:** Zmieniano ilość produktów w koszyku i sprawdzono, czy licznik przedmiotów aktualizuje się odpowiednio.
- **Odświeżanie strony:** Odświeżano stronę i sprawdzono, czy liczba przedmiotów w koszyku pozostaje poprawna.

Menu drop-down

Testy funkcjonalne:

- **Sprawdzanie widoczności:** Kliknięto na menu drop-down i upewniono się, że wszystkie pozycje są wyświetlane poprawnie.
- **Nawigacja:** Klikano na różne pozycje menu i sprawdzano, czy przekierowują do odpowiednich stron.

- **Zamykanie menu:** Klikano poza menu drop-down i sprawdzano, czy zamyka się poprawnie.

Koszyk nie pokazuje ilości przedmiotów

Testy funkcjonalne:

- **Dodawanie produktów:** Dodano różne ilości produktów do koszyka i sprawdzono, czy liczba przedmiotów w koszyku aktualizuje się poprawnie.
- **Zmiana ilości:** Zmieniano ilość produktów w koszyku i sprawdzano, czy licznik przedmiotów aktualizuje się odpowiednio.
- **Odświeżanie strony:** Odświeżano stronę i sprawdzano, czy liczba przedmiotów w koszyku pozostaje poprawna.

Usuwanie z koszyka

Testy funkcjonalne:

- **Usuwanie pojedynczych produktów:** Usuwano pojedyncze produkty z koszyka i sprawdzano, czy liczba przedmiotów oraz wartość zamówienia aktualizują się poprawnie.
- **Usuwanie wszystkich produktów:** Usuwano wszystkie produkty z koszyka i sprawdzano, czy koszyk jest pusty i liczba przedmiotów wynosi 0.
- **Potwierdzenie usunięcia:** Sprawdzano, czy użytkownik otrzymuje komunikat potwierdzający usunięcie produktu.

Sumowanie produktów w koszyku

Testy funkcjonalne:

- **Dodawanie różnych produktów:** Dodano różne produkty do koszyka i sprawdzano, czy suma wartości produktów jest poprawnie obliczana.
- **Zmiana ilości:** Zmieniano ilość poszczególnych produktów w koszyku i sprawdzano, czy suma wartości zamówienia aktualizuje się odpowiednio.
- **Usuwanie produktów:** Usuwano produkty z koszyka i sprawdzano, czy suma wartości zamówienia jest aktualizowana poprawnie.

2.7 Rzut diagramu klas

3. Podsumowanie

3.1 Ujęcie końcowe

Dokumentacja techniczna stanowi kluczowy element ewidencji pracy nad projektem, zapewniając szczegółowe informacje niezbędne do zrozumienia, użytkowania, utrzymania i rozwijania produktu lub systemu. Istnienie tego dokumentu ma być pomocą w utrzymaniu infrastruktury sklepu internetowego „Figuya-Onrine”.

3.2 Dalszy Rozwój projektu

Dalszy rozwój projektu przewiduje nadzorowanie pojawiających się błędów i eliminację ich. Dodawanie potrzebnych rozwiązań. Uzupełnianie dokumentacji do naniesionych zmian.

4. Devlog Spis

Ostania aktualizacja: **31.05.2024**

4.1 Devlog

Edycja 0.1.0 31.05.2024

Zakończenie wstępnego etapu prowadzenia dokumentacji

Edycja 0.0.1 28.05.2024

Update

Edycja 0.0.0 08.03.2024

Rozpoczęcie dokumentacji