

SpeedCheck. Faza konstrukcji

Marko Golovko, Yurii Purdenko

31 grudnia 2020

Historia zmian

Numer wersji	Data	Autor	Opis
0.1	14.12.20	YP	Utworzenie dokumentu

Spis treści

1 Testy funkcjonalne

1.1 Wyszukiwanie samochodów

1.1.1 User Story

- Nazwa: Wyszukiwanie samochodu
- Historyjka użytkownika: Jako użytkownik, chce mieć możliwość wyszukiwania samochodu, aby widzieć go ostatnią lokalizację.
- Funkcjonalne kryteria akceptacji:
 - Strona wyświetla lokalizacje samochodu.
 - Jeżeli lokalizacja nie była znaleziona, wyświetla komunikat, że nie udało się znaleźć.
 - Jeżeli użytkownik nie ma prawa do wyszukiwania, wyświetla komunikat, że nie ma prawa do wyszukiwania.

1.1.2 Use case

- Tytuł: Wyszukiwanie samochodu
- Aktor: Użytkownik
- Warunki początkowe:
 - Użytkownik musi być zalogowany.
 - Użytkownik musi mieć prawo do wyszukiwania samochodu.
 - Strona musi mieć pole do wyszukiwania.
- Zdarzenie inicjujące: Użytkownik wpisuje numer samochodu w odpowiednie pole i klika przycisk wyszukiwania.
- Przebieg w krokach:

- Użytkownik loguje się na stronie.
- Użytkownik wpisuje numer samochodu w odpowiednie pole i klika przycisk wyszukiwania
- Strona wyświetla lokalizację samochodu.
- Przebiegi alternatywne:
 - Wyszukiwany samochód nie był znaleziony.
 - Użytkownik nie ma prawa do wyszukiwania samochodu.
- Sytuacji wyjątkowe: Użytkownik jest pracownikiem policji i ma prawo do wyszukiwania dowolnego samochodu.
- Warunki końcowe:
 - Strona wyświetla lokalizację samochodu.
 - Albo strona wyświetla komunikat o nieudanej próbie.

1.2 Przegląd kamer monitorujących

1.2.1 User Story

- Nazwa: Przegląd kamer monitorujących
- Historyjka użytkownika: Jako użytkownik, chce mieć możliwość przeglądu kamer monitorujących, by wiedzieć o sytuacji na drogach
- Funkcjonalne kryteria akceptacji:
 - Strona wyświetla dostępne transmisje.
 - Jest możliwość filtracji za lokalizacją.
 - Jest możliwość oglądania kilku kamer(do 9) jednocześnie.

1.2.2 Use case

- Tytuł: Przegląd kamer monitorujących
- Aktor: Użytkownik
- Warunki początkowe:
 - Użytkownik musi być zalogowany.
 - Strona ma dostępne transmisje
 - Użytkownik znajduje się w sekcji „Przegląd kamer online”
- Zdarzenie inicjujące: Użytkownik klika na zakładkę „Przegląd kamer online”

- Przebieg w krokach:
 - Użytkownik loguje się na stronie.
 - Użytkownik zaznacza lokalizacje kamer z transmisją.
 - Strona wyświetla transmisji.
- Przebiegi alterantywne:
 - Transmisji nie były znalezione.
 - Użytkownik dodaje kolejną transmisję do oglądania.
- Sytuacji wyjątkowe: Użytkownik chce dodać więcej niż dziewięć transmisji na raz.
- Warunki końcowe:
 - Strona wyświetla dostępne transmisji.
 - Albo strona wyświetla komunikat o nieudanej próbie.

1.3 Sprawdzenia przekroczeń limitu prędkości

1.3.1 User Story

- Nazwa: Sprawdzenia przekroczeń limitu prędkości
- Historyjka użytkownika: Jako użytkownik, chce mieć możliwość sprawdzenia przekroczeń limitu prędkości, by mieć możliwość zabezpieczyć bezpieczną sytuację na drogach.
- Funkcjonalne kryteria akceptacji:
 - Strona wyświetla dostępne transmisji.
 - Jest możliwość filtracji za lokalizacją.
 - Jest możliwość oglądania kilku kamer(do 9) jednocześnie.
 - Na transmisji jest pokazane przekroczenie prędkości.

1.3.2 Use case

- Tytuł: Sprawdzenia przekroczeń limitu prędkości
- Aktor: Użytkownik
- Warunki początkowe:
 - Użytkownik musi być zalogowany.
 - Musi mieć prawo do sprawdzenia prędkości wszystkich samochodów (np. być pracownikiem policji).

- Strona ma dostępne transmisje.
- Użytkownik znajduje się w sekcji „Sprawdzenia przekroczeń limitu”.
- Zdarzenie inicjujące: Użytkownik klika na zakładkę „Sprawdzenia przekroczeń limitu”
- Przebieg w krokach:
 - Użytkownik loguje się na stronie.
 - Użytkownik zaznacza lokalizacje kamer z transmisją.
 - Strona wyświetla transmisji i pokazuje przekroczenia prędkości.
- Przebiegi alterantywne:
 - Transmisji nie były znalezione.
 - Użytkownik dodaje kolejną transmisję do oglądania.
- Sytuacji wyjątkowe: użytkownik nie jest pracownikiem policji i ma możliwość do sprawdzenia prędkości tylko własnego samochodu.
- Warunki końcowe:
 - Strona wyświetla dostępne transmisji.
 - Strona pokazuje przekroczenia prędkości.
 - Albo strona wyświetla komunikat o nieudanej próbie.

2 Pomiary spełnienia wymagań niefunkcjonalnych

2.1 Wymaganie niefunkcjonalne

• 2.2 Szybkość aplikacji. aplikacja musi być szybka

- Pomiar: $X = A/N$
- $A =$
- Wykorzystanie pamięci operacyjnej. Sprawdzanie zajętości w procentach pamięci RAM
- Łatwość użycia. Test UX.
- Niezawodność aplikacji. Sprawdzanie liczby błędów na liczbę godzin użytkowania.
- Działanie aplikacji przy dużej liczbie użytkowników. Stress test.

3 Plan beta testowania

3.1 Planowanie

Zdefiniowanie takich elementów:

- Zakres testów i cel testów
- Elementu testowego
- Funkcjonalności, które zostaną przetestowane
- Poziomy i typy testów
- Kryteria akceptacji
- Rzeczy które są dostarczane razem z testami
- Środowisko testowe
- Harmonogram
- Skład teamu
- Ryzyka i zagrożenia

3.2 Analiza

- Określamy przedmiot testowy
- Określamy cel testowania
- Robimy spis funkcjonalności i możliwości, które będą testowane.
- Z otrzymanej informacji definiujemy ankiety testowe.
- Określamy elementy testowe
- Dokonujemy analizy ryzyka

3.3 Projektowanie testów

- Wyznaczamy poziomy testów dla konkretnych testowanych obszarów
- Wybieramy techniki projektowania
- Tworzymy ankiety testowe
- Priorytetyzacja ankiet testowych

3.4 Implementacja

- Przygotowujemy środowisko testowe
- Testują ochotnicy manualne
- Team testerów kieruje proces beta testowania

3.5 Wykonanie testów

- Użytkownicy wykonują testy na naszym elemencie testowym
- Po wykonaniu testów otrzymane wrażenie zapisują w zawartości aktualnej ankiety
- Ocenę użytkowników team testerów przetwarza w raport
- Zgodnie z raportem część przetestowanej funkcjonalności przechodzi z powrotem do fazy projektowania
- Zgodnie z uwzględnieniem pewnych odstępstw, plan może ulec zmianie.

3.6 Kontrola

- Monitorowanie
- Raportowanie
- Nadzór

3.7 Czynności zamykające testowanie

- Sprawdzić, czy wszystko zostało dostarczone zgodnie z planem i dokumentacją
- Zarządzić błędami, czyli zamykanie notek, tworzenie nowych, edycja istniejących, jeśli jest taka potrzeba
- Utrzymywanie dokumentacji
- Prace niezbędne do utrzymania środowiska testowego i ponownego użycia w przyszłości
- Zorganizować spotkanie retrospektywne
- Zachować artefakty procesu testowego

- 4 Plan zarządzania ryzykiem
- 5 Plan zarządzania jakością wytwarzania oprogramowania
- 6 Ocena zgodności wykonanych prac z wizją systemu i specyfikacją wymagań.

Rezultaty są zgodne z wizją systemu i specyfikacją wymagań.