Specyfikacja wymagań systemowych

Tytuł projektu:

Portal zarządzający i udostępniający informacje przestrzenne o lokalizacjach obiektów określonego rodzaju

**Spis treści**

**Spis treści 2**

**1. Wprowadzenie. 3**

1.1. Cel wykonania systemu 3

1.2. Odbiorcy docelowi 3

**2. Cele biznesowe systemu 3**

2.1. Kontekst biznesowy 3

2.2. Udziałowcy: 3

2.3. Granice funkcjonalności systemu 3

**3. Wymagania funkcjonalne 4**

3.1. Opis szczegółowy systemu 4

3.2. Tabelaryczna lista wymagań funkcjonalnych 4

**4. Wymagania niefunkcjonalne 5**

4.1. Bezpieczeństwo systemu 5

4.2. Dostępność i czas reakcji systemu, skalowalność 5

4.3. Szacowanie ilości danych i czasu realizacji funkcjonalności. Wymagania sprzętowe i środowiskowe 6

4.4. Utrzymanie systemu 6

**5. Technologie i narzędzia planowane do wykorzystania w realizacji projektu 6**

**6. Organizacja projektu. 7**

6.1. Skład osobowy 7

6.2. Harmonogram: 7

**1. Wprowadzenie.**

***1.1. Cel wykonania systemu***

Implementacja internetowego portalu, który dostarczać będzie informacji m. in. o lokalizacji i innych cechach określonego rodzaju obiektów z danego obszaru. Wybranymi przez nas obiektami są skłdy budowlane. Przykładowe dane opisujące konkretny obiekt to: nazwa punktu, ew. przynależność do sieci tego rodzaju punktów, oferta (kategorie produktów), ceny najbardziej popularnych produktów, lokalizacja geograficzna punktu usługowego. Portal będzie dodatkowo umożliwiał dodatkowe funkcjonalności.

***1.2. Odbiorcy docelowi***

|  |
| --- |
| Osoby budujące domy/mieszkania we własnym zakresie.  Handlarze/dostawcy materiałów budowlanych.  Osoby potrzebujące produktów ze składów budowlanych z innych powodów. |

**2. Cele biznesowe systemu**

***2.1. Kontekst biznesowy***

Przewidziano system jako samodzielną aplikację.

Dzięki niemu oczekuje się korzyści w postaci zmniejszenia czasu i pracy potrzebnej na odnalezienie interesującego nas produktu, znajdującego się w składach budowlanych w okolicy. Aplikacja będzie albo darmowym produktem promującym umiejętności jej twórców, albo zawierającym w sobie materiały promocyjne sponsorów. Będzie on ułatwieniem dla użytkowników, potrzebujących określonych produktów ze składów budowlanych oraz reklamą dla dostarczycieli tych produktów.

|  |
| --- |
|  |

***2.2. Udziałowcy:***

|  |
| --- |
| - bezpośredni użytkownicy aplikacji,  - administratorzy,  - twórcy oraz osoby zaangażowane w jej utrzymanie. |

***2.3. Granice funkcjonalności systemu***

|  |
| --- |
| Mapa będzie ograniczała się do terenów miasta Gdańsk. Aplikacja nie będzie służyła do komunikacji między sklepem a użytkownikiem. Użytkownik nie będzie mógł wprowadzać własnych obiektów i/lub markerów poza jednym oznaczającym jego lokalizację lub punkt startowy. Wprowadzaniem danych o obiektach będą zajmować się wyłącznie administratorzy i twórcy, osoby utrzymujące aplikacje. |

**3. Wymagania funkcjonalne**

***3.1. Opis szczegółowy systemu***

Aplikacja będzie przedstawiała mapę Gdańska, na którą zostaną naniesione obiekty w postaci markerów. Obiektami będą składy budowlane, które zawierają w swojej ofercie produkty dostępne dla przeciętnego mieszkańca Gdańska. Obiekty wraz z dodatkowymi informacjami o nich będą przechowywane w bazie danych aplikacji na serwerze. Dodawanie i modyfikowanie obiektów w bazie danych będzie dostępne dla autoryzowanego użytkownika/administratora poprzez dedykowany interfejs. Użytkownik systemu będzie mógł w prosty i przejrzysty sposób wyszukać skład budowlany znajdujący się najbliżej interesującej go lokalizacji w Gdańsku oraz wyświetlić informację o nim takie jak nazwa, adres i inne w postaci tabelarycznej. Po wprowadzeniu punktu startowego przez użytkownika będzie możliwość obliczenia odległości między wprowadzonym punktem a innym, wybranym lub najbliższym, oznaczającym skład budowlany. Mapę będzie można przesuwać oraz zmieniać jej skalę. Będzie dostępne wyszukiwanie obiektów po konkretnych parametrach, np. „znajdź skład sprzedający wiertarki”. Wizualizacja wyników będzie prowadzana przez wyświetlenie ich na mapie lub w tabeli. Dostępne będzie wyliczenie odległości i/lub trasy między punktem startowym a wybranym obiektem (markerem).

|  |
| --- |
|  |

***3.2. Tabelaryczna lista wymagań funkcjonalnych***

|  |  |  |  |
| --- | --- | --- | --- |
| Identyfikator | *MAP* | Priorytet | *wysoki* |
| Tytuł | Wyświetlenie mapy | | |
| Opis | Wyświetla obraz przedstawiający mapę ogólnogeograficzną, na której będą naniesione obiekty | | |
| Powiązanie | brak | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identyfikator | *MARK* | Priorytet | *wysoki* |
| Tytuł | Naniesione na mapę obiekty | | |
| Opis | Obiekty, który pod postacią markerów zostały naniesione na mapę przedstawioną przez aplikację. | | |
| Powiązanie | MAP | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identyfikator | *INTERFACE* | Priorytet | *wysoki* |
| Tytuł | Interfejs użytkownika/administratora | | |
| Opis | Dodawanie i modyfikowanie obiektów w bazie danych będzie dostępne dla autoryzowanego użytkownika/administratora poprzez dedykowany interfejs. | | |
| Powiązanie | MAP, MARK, INFO | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identyfikator | *INFO* | Priorytet | *średni* |
| Tytuł | Prezentacja informacji o obiekcie | | |
| Opis | Po zaznaczeniu danego obiektu zostaną wyświetlone informacje o nim. | | |
| Powiązanie | MAP, MARK | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identyfikator | *DIST* | Priorytet | *niski* |
| Tytuł | Wyliczanie odległości między obiektami | | |
| Opis | Po zaznaczeniu danego obiektu zostanie wyświetlona odległość do niego od punktu startowego | | |
| Powiązanie | MAP, MARK | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identyfikator | *ROUTE* | Priorytet | *niski* |
| Tytuł | Wyliczanie drogi między obiektami | | |
| Opis | Po zaznaczeniu danego obiektu zostanie wyświetlona droga do niego od punktu startowego | | |
| Powiązanie | MAP, MARK | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Identyfikator | *SKALE* | Priorytet | *niski* |
| Tytuł | Scalanie markerów przy zmniejszaniu skali | | |
| Opis | Przy zmniejszaniu skali mapy markery znajdujące się blisko siebie, powinny być zastąpione jednym. | | |
| Powiązanie | MAP, MARK | | |

**4. Wymagania niefunkcjonalne**

***4.1. Bezpieczeństwo systemu***

|  |
| --- |
| Aby uchronić dane o obiektach, dostęp do nich będzie dostępny tylko dla administratorów. Użytkownik będzie mógł wprowadzać tylko maksymalnie jeden punkt, który będzie aktywny wyłącznie w aktualnej sesji i nie będzie zapisywany na stałe. Administratorzy będą mieli dostęp poprzez zalogowanie się lub wcześniejsze upoważnienie systemowe. |

***4.2. Dostępność i czas reakcji systemu, skalowalność***

Aplikacja będzie dostępna poprzez adres WWW lub link do serwera.

|  |
| --- |
| Dostępna będzie zmiana skali mapy. Im niższa skala, tym większe będzie scalanie markerów w jeden (jeden marker będzie reprezentował wiele obiektów).  Czas reakcji, oczekiwania na odpowiedź i wyświetlania wyniku na mapie nie powinien przekroczyć jednej sekundy. |

***4.3. Szacowanie ilości danych i czasu realizacji funkcjonalności. Wymagania sprzętowe i środowiskowe***

|  |
| --- |
| Przewiduje się istnieje około 20 obiektów.  Czas realizacji poszczególnych funkcjonalności będzie krótszy niż 1 sekunda.  Od użytkownika wymagana będzie aktualna przeglądarka internetowa. |

***4.4. Utrzymanie systemu***

|  |
| --- |
| Bazy danych będą przechowywane na zewnętrznym serwerze opłacanym przez właściciela aplikacji. Koszt serwera to ponad 100 zł, dokładna cela zależy od dostawcy hostingu i aktualnej sytuacji na rynku. Punkty będą na bieżąco aktualizowane oraz tworzone w regularnych odstępach czasu lub w razie potrzeby.  Planowana jest rozbudowa aplikacji o większy zakres mapy, obiektów i dodatkowych informacji. |

**5. Technologie i narzędzia planowane do wykorzystania   
w realizacji projektu**

|  |  |
| --- | --- |
| Komponent mapowy – Open Layers  Nominati – geokodowanie, zwracanie informacji o punkcie/markerze  GitHub – repozytorium kodu  Języki webowe: m. in. HTML, CSS, JS  Baza danych oraz język SQL    **6. Organizacja projektu.**    ***6.1. Skład osobowy***   |  | | --- | | Przemysław Głębocki – kierownik projektu  Michał Ścibisz – frontend  Tomasz Formela - backend  Krzysztof Dąbrowski – baza danych i dokumentacja    ***6.2. Harmonogram:*** |   31 października - dokumentacja początkowa,  30 listopada - projekt systemu (frontend, wizualizacja),  15 stycznia - gotowa wstępna wersja systemu (backend),  25 stycznia - gotowa ostateczna wersja systemu oraz dokumentacji końcowa (powiązanie z bazą danych). |
|  |
|  |