Inżynieria oprogramowania

Projekt Portalu Widelczyk.pl

# Dane ogólne

Właściciele: Łukasz Starzyk i Dominik Tomasiewicz

Pracownicy: Bil Gejts, Gejb Niuel

Właściciele nie wykluczają zatrudnienia większej liczby pracowników w przyszłości.

Adres: ul Zachlapana 42, 31-999 Kraków

# Ogólne założenia projektu

Firma chce zbudować portal internetowy z bazą restauracji i innych lokali gastronomicznych. Użytkownicy będą mogli oceniać każdą z restauracji oraz pisać własne opinie na ich temat. Ponadto będzie również wyszukiwarka lokali wraz z możliwością wyszukiwania restauracji w podanym mieście lub okolicy.

## Użytkownicy indywidualni

Każda osoba odwiedzająca portal powinna mieć możliwość przeglądania bazy restauracji według własnych kryteriów, gdzie ma dostęp do przeglądania aktualnych ocen oraz opinii innych użytkowników. Funkcjonalność ta nie wymaga logowania, ale nie pozwala na dodawanie nowych opinii i ocen. Każdy użytkownik ma możliwość założenia indywidualnego konta w portalu. Pozwala mu ono na dodawanie opinii, ocenianiu i przeglądaniu dodanych przez siebie wpisów i ocen.

## Użytkownicy firmowi

Każda osoba ma możliwość założenia konta które będzie pozwalać na dodawanie nowych lokali do bazy portalu. Podczas rejestracji wymagane są dodatkowe informacje takie jak imię i nazwisko, nr telefonu lokalu, telefon komórkowy oraz stanowisko. Konto takie nie jest aktywowane automatycznie. Wymaga ono weryfikacji przez pracownika firmy który sprawdza jego autentyczność. Po weryfikacji użytkownik ma możliwość dodawania informacji o własnych restauracjach.

## Pracownicy

Każdy pracownik posiada indywidualne konto w systemie. Daje mu ono uprawnienia do zarządzania listą lokali, kontami użytkowników oraz moderowania ich opinii. Pracownik może dać użytkownikowi ostrzeżenie za niestosowną opinię lub inne nieodpowiednie dane. Trzy ostrzeżenia powodują automatyczną blokadę konta.

## Właściciele

Właściciel posiada konto w systemie z najwyższymi uprawnieniami. Może on robić wszystko to co pozostali użytkownicy ponadto ma on możliwość zarządzania kontami pracowników oraz informacjami firmy dostępnymi na stronie. Kont o najwyższych uprawnieniach może być w systemie więcej niż jedno.

# Wymagania

## Podział uprawnień

**Właściciel** - Osoba będąca właścicielem firmy. Podejmuje on najważniejsze decyzje w firmie, posiada on największe możliwe uprawnienia w systemie.

**Pracownik**- Grupa pracowników posiadająca uprawnienia do modyfikowania informacji wprowadzanych przez użytkowników.

**Użytkownicy firmowi** - grupa zarejestrowanych użytkowników, ma dostęp do modyfikowania danych lokali.

**Użytkownicy indywidualni** - grupa zarejestrowanych użytkowników, ma dostęp do dodawania opinii oraz ocen lokali .

**Goście** - grupa niezarejestrowanych, niezalogowanych użytkowników, która ma prawo do przeglądania podstawowych informacji o lokalach i firmie.

## Wymagania funkcjonalne

|  |  |
| --- | --- |
| Nazwa funkcji | Wyświetlanie informacji o firmie |
| Opis funkcji | Sekcja zawiera podstawowe informacje o działalności firmy. Dane te są dostępne dla wszystkich użytkowników |
| Wejście | Brak |
| Wyjście | Dane o firmie |
| Wymagania | Brak |
| Efekty uboczne | Brak |

|  |  |
| --- | --- |
| Nazwa funkcji | Modyfikowanie informacji o firmie |
| Opis funkcji | Funkcja pozwala właścicielowi zmodyfikować informacje na temat działalności firmy |
| Wejście | Dane do zmodyfikowania |
| Wyjście | Zmiana opisu firmy na stronie |
| Wymagania | Uprawnienia na poziomie właściciela |
| Efekty uboczne | Informacja o powodzeniu lub niepowodzeniu modyfikacji |

|  |  |
| --- | --- |
| Nazwa funkcji | Formularz kontaktowy |
| Opis funkcji | W sekcji dane firmy stworzony jest formularz przy użyciu którego użytkownicy mogą skontaktować się z firmą. |
| Wejście | Treść wiadomości do firmy |
| Wyjście | Wysłanie wiadomości e-mail do firmy z treścią podaną przez użytkownika |
| Wymagania | Brak |
| Efekty uboczne | Wysłanie kopii wiadomości na podany w formularzu adres e-mail użytkownika |

|  |  |
| --- | --- |
| Nazwa funkcji | Dodanie pracownika do systemu |
| Opis funkcji | Pozwala na zapisanie do bazy danych informacji o pracowniku |
| Wejście | Dane pracownika:   * Imię i nazwisko * Adres * Telefon * Dane dotyczące umowy * Nr konta bankowego |
| Wyjście | Zapisanie informacji o pracowniku w bazie danych |
| Wymagania | Uprawnienia na poziomie właściciela |
| Efekty uboczne | Komunikat o powodzeniu lub niepowodzeniu operacji dodawania |

|  |  |
| --- | --- |
| Nazwa funkcji | Modyfikacja / usunięcie danych pracownika z systemu |
| Opis funkcji | Pozwala na modyfikację lub usunięcie danych pracownika z systemu |
| Wejście | Dane pracownika |
| Wyjście | Komunikat proszący o potwierdzenie operacji. |
| Wymagania | Uprawnienia na poziomie właściciela |
| Efekty uboczne | Po potwierdzeniu użytkownik otrzymuje komunikat o powodzeniu lub niepowodzeniu operacji. Modyfikacja informacji o pracowniku w bazie danych |

|  |  |
| --- | --- |
| Nazwa funkcji | Logowanie |
| Opis funkcji | Funkcja umożliwia użytkownikowi systemu zalogowanie się. Po zalogowaniu użytkownik może korzystać z funkcji systemu, do których ma uprawnienia |
| Wejście | Login i hasło użytkownika |
| Wyjście | Wysłanie od użytkownika pliku cookie zawierającego dane odnośnie sesji użytkownika |
| Wymagania | Brak |
| Efekty uboczne | Komunikat od powodzeniu lub niepowodzeniu operacji. Po pomyślnym zalogowaniu – przekierowanie na stronę główną |

|  |  |
| --- | --- |
| Nazwa funkcji | Zarządzanie użytkownikami |
| Opis funkcji | Funkcja pozwala na przeglądanie danych zarejestrowanych użytkowników, oraz na przeglądanie i modyfikację wprowadzonych przez nich wpisów. Pozwala również na edycję informacji o użytkowniku lub na jego usunięcie. |
| Wejście | Dane lub opinie użytkownika |
| Wyjście | Modyfikacja informacji w bazie danych |
| Wymagania | Uprawnienia na poziomie właściciela lub pracownika |
| Efekty uboczne | Wyświetlenie aktualnych danych modyfikowanego użytkownika w celu ewentualnych modyfikacji |

|  |  |
| --- | --- |
| Nazwa funkcji | Rejestracja użytkownika indywidualnego |
| Opis funkcji | Funkcja umożliwia użytkownikowi utworzyć nowe konto indywidualne w systemie. Po rejestracji użytkownik otrzymuje wiadomość z danymi które podał oraz linkiem aktywacyjnym, który musi użyć do aktywacji konta |
| Wejście | Klient podczas procesu rejestracji jest zobowiązany do podania następujących danych: login, hasło, e-mail |
| Wyjście | Dodanie wpisu w bazie danych |
| Wymagania | Brak |
| Efekty uboczne | Komunikat o powodzeniu lub niepowodzeniu operacji.  Wysłanie do klienta wiadomości e-mail z danymi oraz linkiem aktywacyjnym. |

|  |  |
| --- | --- |
| Nazwa funkcji | Rejestracja użytkownika firmowego |
| Opis funkcji | Funkcja umożliwia użytkownikowi utworzyć nowe konto firmowe w systemie. Po rejestracji użytkownik otrzymuje wiadomość z danymi które podał. Musi oczekiwać na aktywacje konta przez pracownika firmy |
| Wejście | Klient podczas procesu rejestracji jest zobowiązany do podania następujących danych: login, hasło, e-mail, imię i nazwisko, nr telefonu, nr komórkowy oraz możliwe dodatkowe informacje |
| Wyjście | Dodanie wpisu w bazie danych |
| Wymagania | Brak |
| Efekty uboczne | Wysłanie do klienta wiadomości e-mail z danymi.  Komunikat o powodzeniu lub niepowodzeniu operacji. |

|  |  |
| --- | --- |
| Nazwa funkcji | Aktywacja konta firmowego |
| Opis funkcji | Funkcja umożliwia pracownikowi aktywację lub odrzucenie oczekujących kont firmowych po wcześniejszej weryfikacji ich poprawności |
| Wejście | Panel aktywacyjny z polem na komentarz |
| Wyjście | Modyfikacja informacji w bazie danych |
| Wymagania | Uprawnienia na poziomie właściciela lub pracownika |
| Efekty uboczne | Wysłanie do klienta firmowego wiadomości o aktywacji lub jej braku wraz z komentarzem.  Komunikat o powodzeniu lub niepowodzeniu operacji. |

|  |  |
| --- | --- |
| Nazwa funkcji | Dodanie informacji o lokalu |
| Opis funkcji | Funkcja umożliwia dodanie informacji o lokalu. Informacje te będą widoczne dla wszystkich użytkowników |
| Wejście | Podanie podstawowych informacji o lokalu takich jak nazwa, adres, kontakt oraz opis |
| Wyjście | Dodanie informacji do bazy danych |
| Wymagania | Uprawnienia na poziomie klienta firmowego , pracownika lub właściciela |
| Efekty uboczne | Komunikat o powodzeniu lub niepowodzeniu operacji |

|  |  |
| --- | --- |
| Nazwa funkcji | Modyfikacja informacji o lokalu |
| Opis funkcji | Funkcja umożliwia zmianę informacji o lokalu. Pracownicy oraz właściciel może modyfikować dane wszystkich lokali. Klient firmowy ma prawo do zmiany informacji tylko dodanych przez niego wcześniej. Informacje te będą widoczne dla wszystkich użytkowników |
| Wejście | Dane do modyfikacji |
| Wyjście | Modyfikowanie informacji w bazie danych |
| Wymagania | Uprawnienia na poziomie klienta firmowego , pracownika lub właściciela |
| Efekty uboczne | Komunikat o powodzeniu lub niepowodzeniu operacji. |

|  |  |
| --- | --- |
| Nazwa funkcji | Przeglądanie opinii |
| Opis funkcji | Funkcja umożliwia zmianę przeglądanie opinii i ocen lokali które znajdują się na portalu |
| Wejście | Brak |
| Wyjście | Wyświetlenie opinii i ocen |
| Wymagania | Brak |
| Efekty uboczne | Brak |

|  |  |
| --- | --- |
| Nazwa funkcji | Dodanie opinii |
| Opis funkcji | Funkcja umożliwia dodanie przez użytkownika swojej oceny oraz opinii o danym lokalu |
| Wejście | Ocena oraz opinia |
| Wyjście | Dodanie informacji w bazie danych |
| Wymagania | Uprawnienia na poziomie użytkownika indywidualnego lub wyższe |
| Efekty uboczne | Komunikat od powodzeniu lub niepowodzeniu operacji. |

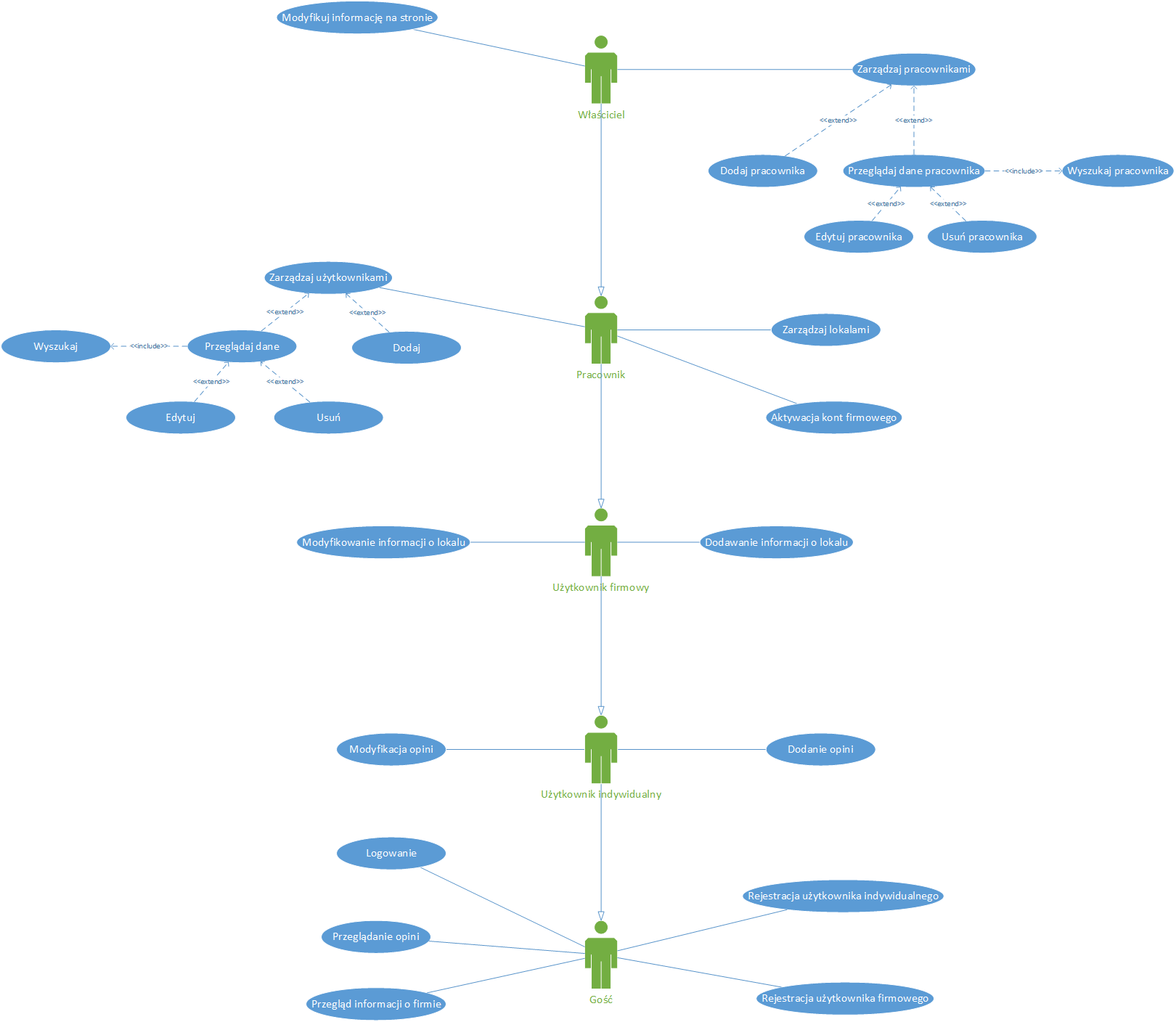
|  |  |
| --- | --- |
| Nazwa funkcji | Modyfikacja opinii |
| Opis funkcji | Funkcja umożliwia zmianę dodanych wcześniej informacji przez użytkownika. Pracownicy i właściciel ma możliwość zarządzania wszystkimi opiniami oraz ocenami. |
| Wejście | Dane do modyfikacji |
| Wyjście | Modyfikowanie informacji w bazie danych |
| Wymagania | Uprawnienia na poziomie użytkownika indywidualnego lub wyższe |
| Efekty uboczne | Komunikat od powodzeniu lub niepowodzeniu operacji |

## Wymagania niefunkcjonalne

1. Dostępność - system powinien być dostępny przez co najmniej 99% czasu w miesiącu
2. Kopie zapasowe – dane przechowywane w bazie danych (dane użytkowników, firm etc.) powinny być backupowane aby zapobiec ewentualnej utracie tych danych
3. Zgodność z przepisami – aplikacja, prezentowane dane i sposób ich przechowywania powinny być zgodne z odpowiednimi przepisami obowiązującymi w danym kraju (jeśli takowe regulacje istnieją)
4. Wydajność – aplikacja powinna wymagać jak najmniejszych zasobów dla danego obciążenia)
5. Rozszerzalność – aplikacja powinna być zaprojektowana w sposób pozwalający na późniejsze jej rozbudowywanie
6. Łatwość konserwacji – aplikacja powinna być zaprojektowana tak aby dało się relatywnie łatwo naprawiać wszelkie usterki
7. Wieloplatformowość – aplikacja powinna działać na możliwie jak największej ilości urządzeń
8. Prywatność – dane użytkowników powinny być przechowywane w bezpieczny sposób
9. Funkcjonalność
10. Praktyczność – aplikacja powinna zapewniać komfort użytkowania
11. Czas odpowiedzi – aplikacja powinna zapewniać czas odpowiedzi do użytkownika nie dłuższy niż 3s
12. Stabilność
13. Testowalność – aplikacja powinna być przystosowana do łatwego testowana
14. Skalowalność – zapewnienie coraz wydajniejszej pracy systemu w miarę zwiększania liczby elementów wykonawczych systemu

# Diagram przypadków użycia

## Schemat



## Szczegółowy opis

|  |  |
| --- | --- |
| Nazwa: | Modyfikuj informację na stronie |
| Aktorzy: | Właściciel |
| Cel: | Właściciel chce zmienić podstawowe informację o firmie lub dane kontaktowe |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Informacje* na stronie 2. Użytkownik może dowolnie modyfikować zawarte na stronie informację o firmie |
| Rozszerzenia: | * 1. Wyświetlany jest komunikat o błędzie |

|  |  |
| --- | --- |
| Nazwa: | Zarządzaj pracownikami |
| Aktorzy: | Właściciel |
| Cel: | Właściciel chce przeglądać informacje o pracowniku, dodawać, usuwać lub modyfikować jego dane |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Zarządzaj pracownikami* 2. Użytkownik może teraz dodać nowego pracownika lub wprowadzić kryteria wyszukiwania pożądanego pracownika 3. Wyświetlany jest pracownik, który spełnił kryteria wyszukiwania 4. Użytkownik może przeglądać jego dane, modyfikować je lub usuwać danego pracownika |
| Rozszerzenia: | * 1. W przypadku braku pracownika o podanych kryteriach wyszukiwania wyświetlany jest komunikat o błędzie |

|  |  |
| --- | --- |
| Nazwa: | Zarządzaj użytkownikami |
| Aktorzy: | Właściciel, Pracownik |
| Cel: | Zarządzanie kontami użytkowników |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Zarządzaj użytkownikami*  2. Użytkownik ma teraz możliwość dodania nowego użytkownika do bazy danych  3. Użytkownik może przeglądać konta użytkowników, edytować ich dane oraz usuwać konta |
| Rozszerzenia: | 3.1 W przypadku braku użytkownika o podanych kryteriach wyszukiwania wyświetlany jest komunikat o błędzie |

|  |  |
| --- | --- |
| Nazwa: | Zarządzaj lokalami |
| Aktorzy: | Właściciel, Pracownik |
| Cel: | Zarządzanie lokalami |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Zarządzaj lokalami*  2. Użytkownik może teraz swobodnie edytować istniejące lokale  3. Użytkownik ma możliwość dodania nowego lokalu |
| Rozszerzenia: |  |

|  |  |
| --- | --- |
| Nazwa: | Aktywacja konta firmowego |
| Aktorzy: | Właściciel, Pracownik |
| Cel: | Aktywacja kont firmowych |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Aktywacja konta firmowego*  2. Użytkownik wybiera konta które chce aktywować |
| Rozszerzenia: | 2.1 Użytkownik może anulować wybrane konta wraz z dodaniem komentarza |

|  |  |
| --- | --- |
| Nazwa: | Dodawanie informacji o lokalu |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy |
| Cel: | Dodawanie nowych informacji o lokalu |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Dodawanie informacji o lokalu*  2. Użytkownik uzupełnia informacje o lokalu |
| Rozszerzenia: |  |

|  |  |
| --- | --- |
| Nazwa: | Modyfikowanie informacji o lokalu |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy |
| Cel: | Zmiana informacji o danym lokalu |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Modyfikowanie informacji o lokalu*  2. Użytkownik uzupełnia lub zmienia informacje o lokalu |
| Rozszerzenia: | 2.1 Wyświetlenie komunikatu o braku uprawnień do edycji nie swojego lokalu |

|  |  |
| --- | --- |
| Nazwa: | Dodawanie opinii |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy, Użytkownik indywidualny |
| Cel: | Dodawanie nowych nowej opinii |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Dodawanie opinii*  2. Użytkownik wybiera lokal do jakiego chce dodać opinie  3. Użytkownik uzupełnia ocenę oraz opinię o wybranym lokalu |
| Rozszerzenia: |  |

|  |  |
| --- | --- |
| Nazwa: | Logowanie |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy, Użytkownik indywidualny, Gość |
| Cel: | Zalogowanie się użytkownika |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Logowanie*  2. System prosi o podanie loginu oraz hasła  3. Użytkownik podaje login i hasło  4. System weryfikuje podane dane oraz loguje klienta na konto |
| Rozszerzenia: | 4.1 W przypadku podania niewłaściwych danych system wyświetla komunikat o błędzie i odsyła z powrotem do strony logowania |

|  |  |
| --- | --- |
| Nazwa: | Rejestracja użytkownik indywidualnego |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy, Użytkownik indywidualny, Gość |
| Cel: | Zarejestrowanie nowego użytkownika indywidualnego |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Rejestracja użytkownik indywidualnego*  2. System prosi o wprowadzenie danych do rejestracji  3. System weryfikuje wprowadzone dane  4. System wysyła użytkownikowi e-mail oraz przekierowuję go na stronę logowania |
| Rozszerzenia: | 3.1 W przypadku podania błędnych danych system wyświetla komunikat o błędach |

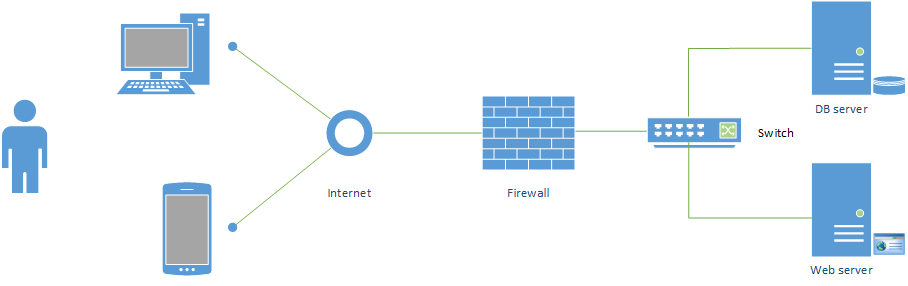
|  |  |
| --- | --- |
| Nazwa: | Rejestracja użytkownika firmowego |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy, Użytkownik indywidualny, Gość |
| Cel: | Zarejestrowanie nowego użytkownika firmowego |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Rejestracja*  2. System prosi o wprowadzenie danych do rejestracji wraz z dodatkowymi danymi na temat firmy  3. System weryfikuje wprowadzone dane  4. System wyświetla komunikat o oczekiwaniu na aktywację przez pracownika |
| Rozszerzenia: | 3.1 W przypadku podania błędnych danych system wyświetla komunikat o błędach |

|  |  |
| --- | --- |
| Nazwa: | Przegląd informacji o firmie |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy, Użytkownik indywidualny, Gość |
| Cel: | Wyświetlenie informacji o firmie |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Przegląd informacji o firmie*  2. System wyświetla aktualne dane firmy |
| Rozszerzenia: |  |

|  |  |
| --- | --- |
| Nazwa: | Przegląd opinii |
| Aktorzy: | Właściciel, Pracownik, Użytkownik firmowy, Użytkownik indywidualny, Gość |
| Cel: | Wyświetlenie cennika |
| Główny scenariusz: | 1. Użytkownik otwiera rozszerzenie *Przegląd cen*  2. Użytkownik wybiera jeden lub więcej lokali wg określonych kryteriów  2. System wyświetla opinie dla lokali |
| Rozszerzenia: |  |

# Architektura systemu

## Diagram

****

## Klient

Do korzystania z aplikacji wymagane jest przeglądarka internetowa obsługująca aktualne standardy (HTML , CSS, JavaScript) oraz połączenie z Internetem. System operacyjny nie powinien mieć wpływu na działanie strony internetowej. Witryna powinna być przystosowana do działania na urządzeniach mobilnych (smartfony i tablety). W przyszłości możliwa jest budowa aplikacji mobilnej na najpopularniejsze systemy mobilne.

## Serwer

W celu zapewnienia wysokiej jakości usługi aplikacja powinna działać na dedykowanych serwerach. Zadaniem serwerów będzie obsługa strony internetowej oraz bazy danych. System musi być podłączony do Internetu przy użyciu łącza o wysokiej przepustowości. System powinien być dostępny bez przerw, niedostępność strony nie powinna być większa niż 7h w ciągu miesiąca (dostępność co najmniej 99% czasu). W przypadku dużego obciążenia serwerów można zwrócić się do firmy hostingowej w celu rozbudowy systemu, zwiększenia mocy serwerów lub łącza.

System powinien obsługiwać do 500 użytkowników jednocześnie a czas odpowiedzi z serwera nie powinien przekraczać 3 sekund.

## Ocena architektury systemu, w kontekście sprzętu.

Decyzja o umieszczeniu serwisu na serwerach hostingowych innej firmy pozwala na zrzucenie odpowiedzialności za konserwacje i utrzymanie maszyn fizycznych na inną firmę. Firmy takie posiadają duże doświadczenie w dziedzinie hostingu stron WWW czy innego rodzaju serwerów. Ponadto mają one najczęściej kilka różnych specyfikacji zależnych od potrzeb użytkowników. Pozwala to na zmianę parametrów systemu w zależności od zapotrzebowania.

## Ocena architektury systemu, w kontekście wykorzystywanych technologii.

System zostanie zbudowany w technologiach MySQL oraz PHP z wykorzystaniem Framework'u Zend. Pozwoli to na wykorzystanie znanych i darmowych technologii które ułatwiają tworzenie systemów, ich rozbudowę i konserwacje. MySQL jest darmową relacyjną bazą danych która jest bardzo często stosowana w budowie stron internetowych. Cieszy się ona dużą popularnością w małych i średnich serwisach internetowych oraz posiada duże wsparcie społeczności.

# Plan testów

## Zgodność z definicją wymagań

Zaplanowane testy mają na celu weryfikację wszystkich określonych funkcjonalności serwisu. Sprawdzenie czy określone funkcje wykonują to co zostało określone w specyfikacji, czy nie powodują one błędów w działaniu strony oraz sprawdzają odporność na dane wprowadzane przez użytkowników.

## Testy funkcjonalne

W tym etapie będą sprawdzane wszystkie funkcje systemu, ich zgodność z wymaganiami, ich pracę w różnych przypadkach testowych.

Sprawdzane moduły to między innymi:

* logowanie
* rejestracja
* zarządzanie użytkownikami
* zarządzanie pracownikami
* dodawanie lokali
* dodawanie opinii

Dane testowe będą podzielone na:

* nietypowe poprawne
* typowe poprawne i błędne
* nietypowe błędne

## Testy bazy danych

Testowana tutaj będzie poprawność bazy danych pod względem technicznym oraz zgodność z postawionymi przed nią zadaniami. Testy te będą trwały przez cały okres produkcji gdyż baza danych będzie wykorzystywana przy pracy nad większością funkcji w serwisie.

## Testy sprawności serwisu

Testy te mają za zadanie sprawdzić jak szybko działa portal. Jak będzie się zachowywać przy zwiększonym obciążeniu, czy czasy odpowiedzi nie wydłużą się znacząco. Sprawdzona zostanie również szybkość i stabilność działania przy dużej ilości wygenerowanych danych. Duża ilość informacji w bazie danych czy na stronie internetowej nie powinna znacząco spowalniać działania systemu. Część tych testów może zostać wykonana w celu sprawdzenia sprawności poszczególnych funkcji ale ostateczne testy wydajnościowe zostaną przeprowadzone na gotowej wersji serwisu.

## Testy akceptacyjne

Test ten przeprowadza klient. Pomyślne przejście tego testu oraz akceptacja klienta oznacza zakończenie prac nad pełną wersją portalu.

# Plan Kosztów

Obliczenie koszu sprzętu sprowadza się do:

**koszt sprzętu = koszt narzędzi + koszt licencji**

Do wykonania projektu używamy darmowych narzędzi programistycznych. Wykorzystując język PHP oraz bazę danych MySQL nie ponosimy kosztów licencji.

**koszty licencji = 0 zł**

Koszty sprzętu to koszty 2 stanowisk komputerowych dla projektantów oraz koszt wynajęcia testowego serwera WWW.

**koszt jednego stanowiska: 3000 zł**

**koszt wynajęcia serwera: 500 zł**

Koszty sprzętu dla projektu wynoszą:

**koszty sprzętu = 2 x 3000 + 500 = 6500 zł**

Koszty opracowania serwisu

koszt opracowania = koszt podstawowy x niezawodność x czas x zasoby x narzędzia x ekspertyza x pensja

koszt podstawowy = Złożoność Systemu x KDSI x Multiplikator wyrażone w osobomiesiącach

(Złożoność Systemu- od 2.4 dla systemów jednowarstwowych do 3.6 dla wielowarstwowych, KDSI Kilo Delivered Source Instructions -tysiące dostarczonych linii kodu, Multiplikator- złożoność danych od 0.7 dla alfanumerycznych do 1.7 dla obiektowych -medialnych)

**koszt podstawowy = 3.0 x 3 x 1.1 = 9.9**

niezawodność -współczynnik niezawodności i bezpieczeństwa systemu od 0.5 dla nieistotnego poziomu do 2.0 dla w pełni zabezpieczonego systemu

czas -wydłużenie lub skrócenie czasu realizacji od 0.5 dla długiego cyklu do 2.0 dla wykonania przed terminem

zasoby - wielkość zasobów danych/sieci od 0.5 dla MB do 2.0 dla TB

narzędzia - współczynnik dostępności narzędzi, od 0.5 dla w pełni dostępnych CASE do 2.0 dla unikatowych rozwiązań bez wspomagania

ekspertyza - współczynnik zależny od zewnętrznych audytów i ekspertyzy, od 0.7 dla braku ekspertyzy do 2.0 dla wymagających certyfikatów

pensja - pensja programisty

**koszt opracowania = 9.9 x 1.5 x 1.2 x 1 x 0.5 x 0.7 x 2500 = 15 592.50 zł**

Koszt całościowy

**koszt = koszt sprzętu + koszt opracowania = 6500 + 15 692.50 = 22 092.50 zł**

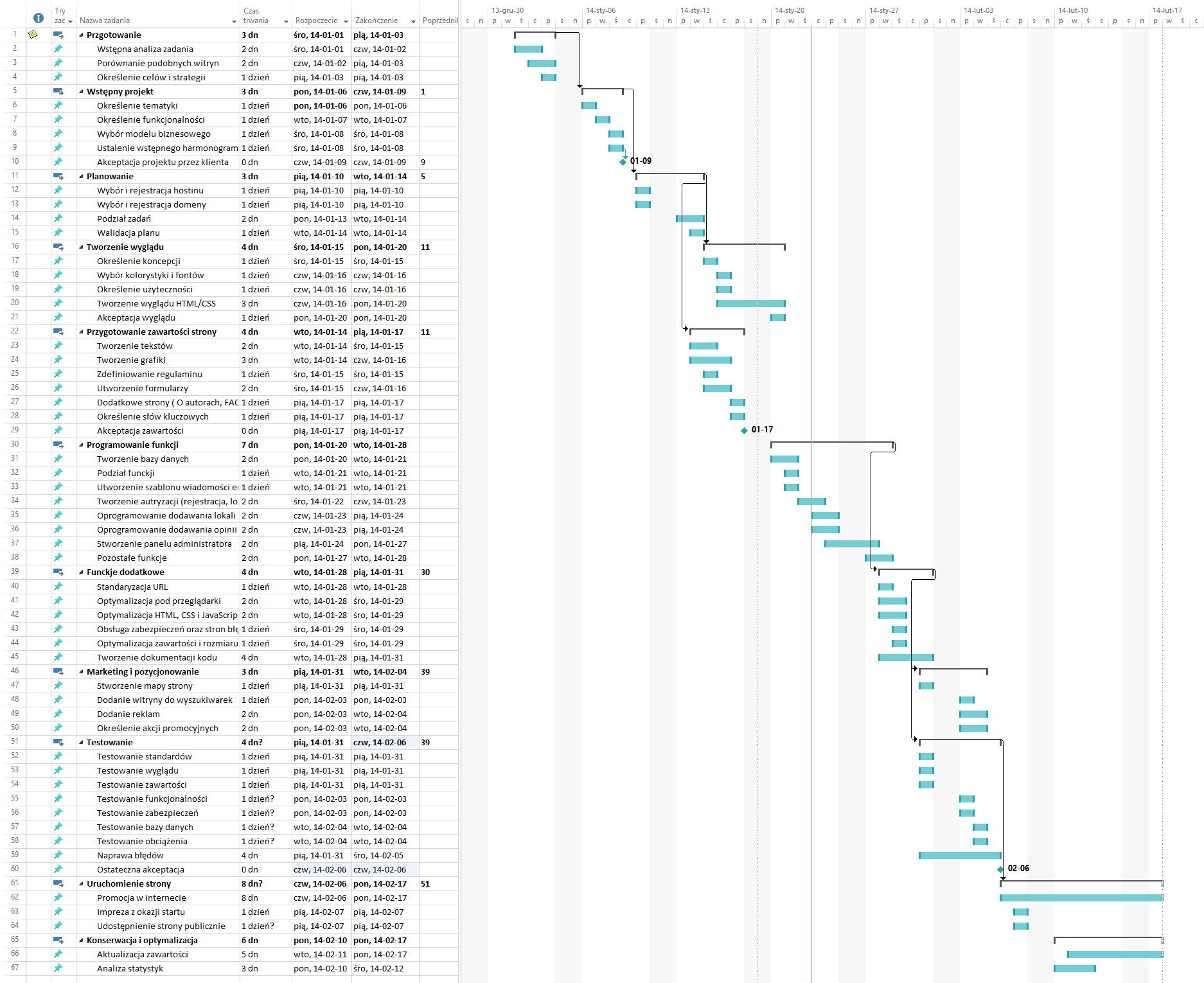
# Plan konserwacji i aktualizacji

Konserwacja i aktualizacja serwisu będzie obejmować pomoc techniczną, konserwację serwerów, aktualizację portalu w celu usunięcia znalezionych błędów oraz dodawania nowych funkcjonalności.

Serwis będzie objęty pomocą techniczną która będzie polegać na naprawach awarii serwisu, zbieraniu informacji o usterkach i błędach systemu w celu wypuszczenia poprawek. Analizowaniu działania strony w celu optymalizacji jej działania. Kolejnym elementem konserwacji będzie kontrola serwerów bazodanowych, sprawdzanie spójności bazy oraz tworzenie fizycznych kopii bazy danych.

# Harmonogram projektu

Przygotowany harmonogram opisuje cały proces od tworzenia projektu poprzez jego implementację, testowanie do wdrożenia. Projekt składa się z głównych etapów które zostały podzielone na zadania. Każde zadanie zostało precyzyjnie zdefiniowane i rozplanowane w czasie. Czas ich trwania podano w dniach roboczych. Na harmonogramie zaznaczono też kamienie milowe, czyli istotne punkty kontrolne realizacji projektu.



# Interfejs

Projekt interfejsu użytkownika zakłada stworzenie strony internetowej portalu. Powinna być ona prosta w obsłudze dla wszystkich użytkowników. Strona powinna działać na najpopularniejszych przeglądarkach oraz na urządzeniach mobilnych. Wszystkie funkcje powinny być łatwo dostępne oraz odpowiednio zabezpieczone przed nieautoryzowanym dostępem.

Przykład implementacji strony głównej portalu.



