Uniwersytet Zielonogórski

Wydział Informatyki, Elektrotechniki i Automatyki

Praca dyplomowa

Kierunek: Informatyka

Aplikacja wspomagająca dietoterapię

Rafał Kulus

Promotor:

Dr Inż. Marek Wróblewski

Zielona Góra, miesiąc rok

Streszczenie

Streszczenie pracy dyplomowej powinno zawierać kilkuzdaniowe omówienie zagadnień poruszanych w pracy. W tej części należy pokrótce scharakteryzować cel oraz podstawowe założenia pracy na tle stanu nauki i techniki w danej dziedzinie, jak również zamieścić od 2 do 6 słów kluczowych.

Słowa kluczowe: praca dyplomowa, skład komputerowy, formatowanie dokumentu.

Spis treści

Spis rysunków

Spis tabel

Rozdział 1

Wstęp

1.1. Wprowadzenie

1.2. Cel i zakres pracy

W tym miejscu należy jasno sformułować cel pracy. Podrozdział ten pisze się tak, aby czytelnik mógł zrozumieć motywację podjęcia badań bez konieczności zapoznania się z dodatkową literaturą. Należy uprzednio dokonać przeglądu literaturowego, który ma pokazać, że autor ma świadomość historii badań nad rozwiniętym problemem oraz aktualnego ich stanu. Powinno się tu wskazać na korelacje z istniejącymi podejściami oraz brakiem istniejącej wiedzy motywujące podjęcie badań objętych pracą dyplomową. Przykład: Celem pracy było opracowanie i realizacja projektu prezentacji dydaktycznej, przeznaczonej dla osób pragnących uzupełnić swoją wiedzę w zakresie poznania języka LaTeX oraz zasad składu tekstu.

Praca swym zakresem obejmowała:

- zgromadzenie i zapoznanie się z literaturą tematu,
- opracowanie założeń projektu,
- zaprojektowanie struktury logicznej pracy,
- realizację projektu oraz usunięcie błędów.

Cel pracy: Celem pracy jest stworzenie aplikacji webowej typu pro-health w

technologii Java Jakarta. Głównym zadaniem aplikacji będzie analiza stosowanej przez użytkownika diety.

Zakres pracy:

- 1. Określenie funkcjonalności aplikacji.
- 2. Implementacja aplikacji webowej typu pro-health w technologii Java Jakarta.
- 3. Opracowanie architektury relacyjnej bazy danych.
- 4. Implementacja mechanizmów niezbędnych w procesie analizy stosowanej przez użytkownika diety polegającej na przeprowadzeniu kalkulacji wartości energetycznej aktualnie stosowanej diety oraz wskazanie w niej niedoborów, a także rekomendacja produktów uzupełniających zalecany bilans energetyczny w zależności od charakteru stosowanej diety i aktywności (redukcja masy, kuracja lecznicza, trening siłowy).
- 5. Projekt i implementacja graficznego interfejsu użytkownika aplikacji.

1.3. Struktura pracy

Tu powinno znaleźć się omówienie treści pracy w postaci opisu zawartości poszczególnych rozdziałów.

Uwaga: Treść tego rozdziału najrozsądniej napisać po napisaniu pozostałej części pracy.

Rozdział 2

Projektowanie aplikacji internetowych

2.1. Przegląd dostępnych rozwiązań

Projektowanie aplikacji internetowych jest bardzo rozległym tematem, którego wątek można podzielić na kilka podsekcji. Mogą to być min. interfejs użytkownika, architektura systemu, bezpieczeństwo, wydajność itd. Istnieje również wiele technologii oraz frameworków, które przydadzą się w tworzeniu aplikacji internetowych. Jeśli chodzi o tworzenie interfejsu użytkownika można skorzystać z następujących frameworków:

- React.js
- Angular
- Vue.js

Frameworki wykorzystywane w budowie backendu:

- Django (Python)
- Ruby on Rails (Ruby)
- Express.js

Bazy danych:

- MySql
- PostgreSQL
- MariaDb

Bezpieczeństwo:

- SSL/TLS
- OAuth, JWT

Testowanie:

- Jest
- Mocha
- Selenium

2.2. Java Jakarta

Jakarta EE, wcześniej znana jako Java Enterprise Edition, jest najnowszą wersją platformy, której zadaniem jest standaryzowanie najpopularniejszych technologii wykorzystywanych do tworzenia aplikacji biznesowych w języku progoramowania Java. Wcześniej odpowiedzialna była za nią firma Oracle, lecz w 2019 została ona przekazana fundacji Eclipse. Zmiana wynikająca z przekazania znaku towarowego wymusiła zmianę nazwy na Jakarta EE. Nie wnosi ona żadnych nowych konstrukcji do języka java, lecz oferuje zestaw specyfikacji i standardów odpowiedzialnych za dostęp do bazy danych, bezpieczeństwo czy też komunikację sieciową. Składa się ona z kilkudziesięciu specyfikacji m.in.:

- Jakarta Servlets - odpowiedzialne za komunikacje sieciowa (obsługa HTTP) w aplikacjach java,
- Jakarta SeverPages (JSP) - silnik szablonów, przeznaczony do tworzenia dynamicznych stron internetowych w języku java,

- JPA (Java Persistence API): Interfejs programowania aplikacji do zarządzania relacyjnymi danymi w aplikacjach,
- JMS (Java Message Service): API do obsługi komunikacji asynchronicznej przez systemy rozproszone,
- Jakarta Bean Validation (walidacja danych).

Serwery aplikacji, które są kompatybilne z Jakarta jest wiele, lecz nie wszystkie wykorzystują jej pełną liste specyfikacji. Takie jak Glassfish, Wildfly czy też Open-Liberty. Aktualnie wiekszość stosowanych rozwiązań to impelementacje, które posiadają jedynie funkcjonalność servletów. Można wyróżnić tutaj Apache Tomcat, którego użyłem do tego projektu, czy też Jetty, które jest propowane przez fundacje Eclipse. To samo rozwiązanie jest stosowane min. w frameworku Spring, stąd rosnąca popularność tego rozwiązania.

2.3. Mechanizmy przechowywania danych w systemach webowych

Mechanizmy do przechowywania danych w systemach webowych, dzielą się na kilka kategorii:

- Bazy danych - jest to najczęstszy rodzaj przechowywania danych w systemach sieciowych.
- Cookies,
- Sesje.

2.4. Model aplikacji internetowej

2.5. Projektowanie responsywnego interfejsu użytkownika

.

2.6. Skalowalność aplikacji internetowych

Rozdział 3

Mechanizmy rekomendacji

3.1. Sposoby pozyskiwania danych w systemach rekomendacyjnych

3.1.1. Profilowanie jawne

3.1.2. Profilowanie domniemane

3.2. Algorytmy rekomendacji

3.2.1. Content-based Filtering

3.2.2. Memory-based

3.2.3. Model-based

3.2.4. User-based

3.2.5. Algorytmy hybrydowe

Rozdział 4

Projekt i implementacja aplikacji internetowej

4.1. Analiza wymagań i funkcjonalności aplikacji

Opis celów aplikacji: Głównym celem aplikacji jest analiza prowadzonej przez użytkownika diety. Ma ona dostarczyć użytkownikom narzędnie, które pozwoli na sprawdzenie czy aktualnie spożywane produkty są dla nich odpowiednie oraz wskazać niedobory w ich diecie. Aplikacja dodatkowo umożliwia wprowadzenie własnej diety, gdzie użytkownik będzie mógł dowiedzieć się ile kalorii dziennie spożywa oraz otrzyma sumowana ilość danego składnika odżywczego. Kolejną funkcjonalnością aplikacji jest możliwość doboru najbardziej odpowiedniej dla użytkownika diety, zgodnej z jego preferencjami odżywczymi takimi jak brak tolerancji dla laktozy czy glutenu oraz czy posiłki, które chce spożywać powinny być wegeteriańskie czy wegańskie. Aplikacja powinna być móc używana przez szerokie spectrum użytkowników, niezależnie od posiadanej wiedzy na temat żywienia. Zapewni to intuicyjny interfejs użytkownika i personalizowane rekomendacje żywieniowe, które przyczynią się do poprawy zdrowia użytkowników. Wymagania funkcjonalne: Rejestracja: Pozwala na utworzenie konta użytkownika

4.2. Architektura aplikacji

4.3. Modelowanie procesów aplikacji

4.3.1. Diagram przepływu danych

4.3.2. Diagram przypadków użycia

4.4. Integracja aplikacji z bazą danych

4.5. Interfejs użytkownika

4.6. Zastosowane algorytmy

Algorytm wykorzystywany aplikacji służy, aby obliczyć dzienne zapotrzebowanie kaloryczne użytkowników. W zależności jak bardzo chcemy być dokładni możemy wykorzystać różne algorytmy. Należy jednak pamiętać, aby użyć bardziej zawansowane algorytmy należy posiadać większą ilość danych od użytkownika min. wszystkie wykonnywane przez użytkownika aktywności wciągu pracy i w czasie wolnym czy też beztłuszczową masę ciała, które wymagały by od użytkownika przeprowadzenie odpowiednich badań. Zgodnie z założeniami aplikacji, miała być ona przeznaczona dla każdego użytkownika, nawet dla tych, którzy nie mają żadnego pojęcia na temat diet. Dlatego zdecydowałem się na użycie wzoru Harrisa-Benedicta, który jest dość precyzyjny natomiast potrzebuje danych, które każdy użytkownik powinien znać. Na początku musimy obliczyć podstawową przemiane materii, w skrócie PPM. Jest to minimalna ilość energii jaką potrzebuje nasz organizm aby potrzymać funkcje życiowe, czyli oddychanie, utrzymanie ciepła czy budowa i odbudowa tkanek. Powinno się dokonywać jego pomiaru wczesnym rankiem, jednak nie można wcześniej przez dwanaście godzin korzystać z żadnych używek. PPM u większości ludzi stanowi największy wydatek energetyczny. W wcześniej podanym wzorze, zależności od płci musimy użyć danego wzoru:

- dla mężczyzn PPM [kcal] = 66,47 + (13,75 × masa ciała w kg) + (5 × wysokość

ciała w cm) – (6,75 × wiek),

- dla kobiet PPM [kcal] = 665,09 + (9,56 × masa ciała w kg) + (1,85 × wysokość ciała w cm) – (4,67 × wiek).

Następnie pod uwagę musimy wziąć aktywność fizyczna, wykonywana przez danego użytkownika. Obliczamy więc poziom aktywności fizycznej (PAL), w tym celu użytkownik musi określić swoją aktywność w pracy oraz w czasie wolnym. Następnie musimy pomnożyć wcześniej uzyskany wynik PPM przez poniżej przedstawiona tabele. Otrzymany wynik to dzienne zapotrzebowaie kaloryczne danego użytkownika, teraz w zależności czy dany użytkownik chce przytyć czy też schudnać odejmujemy lub dodajemy od 200 do 300 kalorii. Pozwoli to użytkownikowi na otrzymanie żądanego rezultatu.

4.7. Środowisko uruchomieniowe

4.8. Autoryzacja użytkowników

4.9. Bezpieczeństwo aplikacji

4.10. Obsługa błędów

Rozdział 5

Testowanie aplikacji

5.1. Testy manualne

5.2. Testy jednostkowe

5.3. Testy interfejsu użytkownika

Rozdział 6

Wnioski

Rozdział 7

Literatura

Dodatek A

"Kody źródłowe”

Do materiałów, które mogą uzupełnić pracę, oprócz rysunków, tablic, ilustracji itp. należą także dodatki, nazwane również aneksami. Dają one możliwość albo dołączenia do tekstu głównego różnorodnego rodzaju informacji dodatkowych, albo wyłączenia z tekstu głównego tych wiadomości, które nie są w nim konieczne. Niekiedy pewne wiadomości wplecione w tekst niepotrzebnie go obciążają, przerywają zasadniczy wątek lub są nadmiernie szczegółowe. Jeśli mimo to wiadomości te są użyteczne i mogą być przydatne, warto oczyścić z nich tekst główny i zgrupować je na końcu pracy w postaci dodatków.

Przykładem użycia dodatków może być opis zawartości płyty CD lub DVD dołączonej do pracy lub instrukcje laboratoryjne stworzone w oparciu o napisaną pracę.

Bibliografia