****

**Praca projektowa programowanie obiektowe**

*System zarządzania notatkami*

Prowadzący: Autor:

mgr inż. Ewa Żesławska Ewelina Kaniewska

nr albumu: 125129

Kierunek: Informatyka, grupa lab 2

Rzeszów 2024

Spis treści

[**1.** **Opis założeń projektu** 3](#_Toc155609263)

[**2.** **Specyfikacja wymagań** 5](#_Toc155609264)

[**3.** **Opis struktury projektu** 6](#_Toc155609265)

[**4.** **Harmonogram realizacji projektu** 9](#_Toc155609266)

[**5.** **Prezentacja warstwy użytkowej projektu** 10](#_Toc155609267)

[**6.** **Podsumowanie** 15](#_Toc155609268)

[**7.** **Literatura** 16](#_Toc155609269)

1. **Opis założeń projektu**

W dzisiejszym dynamicznym środowisku pracy, skuteczne zarządzanie informacjami staje się kluczowe dla efektywności i organizacji codziennych zadań. Każdy użytkownik komputera przynajmniej raz spotkał się z sytuacją, kiedy potrzebował zapisać szybko jakąś informację, a w pobliżu nie było żadnego kawałka papieru. Nawet jeśli informacja jest zapisana na karteczce, istnieje duże ryzyko zgubienia takowej kartki. System do Zarządzania Notatkami jest odpowiedzią na potrzeby pracowników biurowych, studentów i wszystkich tych, którzy szukają intuicyjnego i wszechstronnego narzędzia do przechowywania, organizacji oraz łatwego dostępu do swoich notatek.

**Cele i założenia projektu**

Celem projektu jest umożliwienie użytkownikom tworzenia i przechowywania notatek w zewnętrznej bazie danych. Projekt będzie rozwiązaniem dla problemu skutecznego przechowywania informacji, który często wynika z tego, że pracownicy, uczniowie, studenci są roztargnieni i informacja, którą chcą przechować fizycznie na kartce papieru może zaginąć wśród dokumentów, zadań, czy książek. To powoduje późniejsze problemy w pracy, szkole lub na uczelni związane na przykład z zapomnieniem o jakimś ważnym wydarzeniu (spotkanie, kolokwium itd.). Oczywistym jest fakt, że człowiek nie jest jednostką idealną i nie jest w stanie zapamiętać wszystkich drobnych informacji jakie na co dzień do niego trafiają. Aby problem skutecznego zarządzania informacjami został rozwiązany, należy stworzyć narzędzie, które przejmuje część odpowiedzialności za przechowywanie informacji z człowieka na komputer. Narzędzie zostanie stworzone w formie opisywanego w tym dokumencie projektu. Pierwszym krokiem w realizacji problemu będzie specyfikacja wymagań i założeń, następnie zaprojektowanie graficznego interfejsu użytkownika, stworzenie bazy danych, implementacja klas i funkcji i na końcu testy. Wynikiem tych działań będzie system do zarządzania notatkami.

Użytkownik będzie mógł stworzyć nowe konto, które będzie umożliwiało mu dostęp i korzystanie z systemu. Jeśli użytkownik będzie posiadał już konto w systemie, będzie miał możliwość zalogowania się i będzie posiadać dostęp do panelu głównego systemu. Użytkownik będzie mógł dodawać notatki, które będą przypisane do jego konta. Każda notatka będzie mogła być usunięta lub zaktualizowana przez użytkownika, który ją stworzył. Z poziomu każdej pojedynczej notatki będzie można stworzyć nową. Użytkownik będzie miał możliwość wyświetlenia wszystkich notatek z bazy danych, które są przypisane do jego konta. Użytkownik będzie mógł wyeksportować notatkę do pliku csv lub zaimportować wszystkie notatki znajdujące się w tym pliku.

1. **Specyfikacja wymagań**

**Wymagania funkcjonalne**

* Tworzenie notatek – system pozwala na tworzenie notatek przez użytkownika i zapisuje je do bazy danych
* Usuwanie notatek – system pozwala na usuwanie notatek z bazy danych przez użytkownika, ale tylko tych, które sam stworzył.
* Wyświetlanie notatek – system pozwala na wyświetlenie przez użytkownika notatek, które stworzył
* Aktualizowanie notatek – system pozwala na aktualizację istniejącej już notatki
* Eksport notatek do pliku csv – system pozwala na eksport notatek lokalnie do folderu z projektem
* Import notatek z pliku csv – system pozwala załadować notatki z pliku csv znajdującego się w folderze z projektem
* Zabezpieczenia notatek – system zapewnia zabezpieczenie notatek poprzez użycie mechanizmu logowania i rejestracji. Użytkownik może przeglądać, modyfikować i usuwać tylko te notatki, których jest autorem

**Wymagania niefunkcjonalne**

* Dostępność – system będzie dostępny przez blisko 100% czasu w ciągu doby przez najbliższy rok z wyłączeniem czasu na przeprowadzanie prac konserwacyjnych
* Każda funkcja systemu powinna mieć czas reakcji mniejszy niż 5s
* Maksymalna liczba użytkowników powinna być określona na poziomie co najmniej 1000 użytkowników
* Maksymalna liczba notatek dla każdego użytkownika powinna wynosić co najmniej 100
* Interfejs użytkownika powinien być przejrzysty, intuicyjny i łatwy w obsłudze
* Wszystkie notatki powinny być zabezpieczone przed dostępem użytkownika który nie jest ich autorem

1. **Opis struktury projektu**

* Środowisko programistyczne Javy: Java JDK 17.0.1.
* Środowisko programistyczne:

IntelliJ IDEA 2023.3.3 (Community Edition)

Build #IC-233.14015.106, built on January 25, 2024

Runtime version: 17.0.9+7-b1087.11 amd64

VM: OpenJDK 64-Bit Server VM by JetBrains s.r.o.

* Preferowane urządzenie: urządzenie z system operacyjnym Windows 10 z dostępem do internetu.
* System był projektowany na komputerze stacjonarnym z systemem Windows 10.
* Język aplikacji: Język angielski.

Obraz zawierający tekst, diagram, Plan, Równolegle

Opis wygenerowany automatycznie

Rysunek 1 Diagram klas w projekcie

Baza danych jest złożona z dwóch tabel : Users i Notes. Tabela users posiada kolumny: user\_id, która przechowuje id użytkownika, login – jest to login za pomocą którego użytkownik loguje się na swoje konto, email – email, który użytkownik podaje podczas rejestracji i password – hasło, dzięki któremu użytkownik może zalogować się na swoje konto. Tabla Notes składa się z kolumn: notes\_id – id konkretnej notatki, content – treść notatki oraz user\_id – klucz obcy łączący tabelę Users z tabelą Notes.

Wszystkie klasy dziedziczą po JDialog. Odbywa się to poprzez hierarchię klas jaka występuje w projekcie. Klasa Database zapewniająca główną funkcjonalność jaką jest połączenie do bazy danych została umieszczona na szczycie hierarchii. Jako, że klasa MainPanel jest sercem projektu, inne klasy dziedziczą po niej pośrednio lub bezpośrednio. Klasa Login dziedziczy po klasie Register i dzięki temu również może korzystać z metod jakie oferuje JDialog.

Najważniejsze metody w projekcie:

* getCon w klasie Database – metoda odpowiedzialna za połączenie z bazą danych
* getAutenticateUser w klasie Login – metoda odpowiedzialna za weryfikację danych wpisanych przez użytkownika. Metoda sprawdza czy w bazie danych istnieje użytkownik o podanym loginie i haśle i zwraca wartość true lub false. Jeśli użytkownik istnieje, metoda przypisuje zawartość odpowiednich kolumn do pól klasy User, która tworzy obecnie zalogowanego użytkownika.
* Metoda registerUser w klasie Register – metoda odpowiedzialna za zarejestrowanie nowego użytkownika. Metoda waliduje dane wprowadzone przez użytkownika tj. sprawdza czy uzupełnione są wszystkie pola, czy hasła są zgodne oraz czy wybrana nazwa użytkownika jest dostępna. Jeśli rejestracja nie powiedzie się na poziomie wprowadzania danych do bazy, również zostanie wyświetlony odpowiedni komunikat.
* Metoda addUsertoDatabase w klasie Register – metoda wywoływana przez metodę registerUser. Jest to dalszy etap rejestracji nowego użytkownika. W tym momencie otwierane jest połączenie z bazą danych i faktyczne zapisywanie danych w bazie
* Klasa MainPanel zawiera ActionListenery przypisane do przycisków pojawiających się w oknie panelu głównego. Odpowiadają one za: otwarcie okna z nową notatką, otwarcie wszystkich notatek autorstwa zalogowanego użytkownika, zamknięcie aplikacji oraz wczytanie notatek z pliku csv..
* Klasa Note zawiera metody odpowiadające za wykonywanie operacji na notatkach w bazie danych. Metoda addToDatabase() dodaje notatkę do bazy danych i tworzy nowy obiekt notatki, na którym możliwe są pozostałe operacje. Metoda updateDatabase służy do zaktualizowania treści notatki w bazie danych i również tworzy nowy obiekt. Metoda deleteFromDatabase usuwa notatkę korzystając z jej id przypisanego w momencie zapisywania lub aktualizowania. Klasa umożliwia również eksport notatki do pliku csv.
* Klasa nNote zawiera pola opisujące notatkę oraz metodę służącą do pobierania id notatki z bazy danych.
* Klasa User zawiera pola opisujące użytkownika oraz metodę służącą do pobierania id użytkownika z bazy danych.

Przy każdej próbie połączenia z bazą stosowane są bloki try – catch w celu obsługi wyjątku w sytuacji, kiedy połączenie z bazą danych będzie niemożliwe.

1. **Harmonogram realizacji projektu**

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Rysunek 2 Diagran Ganta

Wykonanie projektu zajęło około 35 godzin. Najdłuższym procesem okazało się stworzenie bazy danych online, która będzie hostowana przez co najmniej rok. Problematyczną częścią było podłączenie bazy danych do projektu. Wynikało to z problemu ze sterownikiem do połączenia z bazą danych Postgre.

Projekt realizowany był z wykorzystaniem systemu kontroli wersji Git, wszystkie pliki źródłowe projektu znajdują się pod adres: [https://github.com/ZiemTo/PO-PROJECT/](https://github.com/ZiemTo/PO-PROJECT/%20) i będą dostępne do 04.02.2024. Na rysunku 3 przedstawiono zrzut ekranu pokazujący historię commitów.

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Rysunek 3 Historia commitów

1. **Prezentacja warstwy użytkowej projektu**

Projekt należy uruchomić poprzez uruchomienie klasy Login. Pojawi się okno logowania.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, design

Opis wygenerowany automatycznie

Rysunek 4 Okno logowania

Do pól Login i Password należy wpisać dale logowania do konta systemu. Przycisk Clear umożliwia wyczyszczenie wpisanych wartości z pól. Przycisk Log in rozpoczyna procedurę logowania. Jeżeli użytkownik nie posiada jeszcze konta, należy nacisnąć przycisk Register, który przenosi użytkownika do formularza rejestracji. Po kliknięciu tego przycisku okno logowania zamyka się , a otwiera okno rejestracji.

Obraz zawierający tekst, zrzut ekranu, Czcionka, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 5 Okno rejestracji

W otwartym oknie użytkownik jest proszony o podanie danych do założenia konta. W przypadku jeśli wybrany przez użytkownika login jest już zajęty, pojawi się stosowny komunikat.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Rysunek 6 Komunikat błędu

Podobne komunikaty pojawią się jeśli chociaż jedno pole będzie puste, jeśli hasła nie będą do siebie pasować oraz jeśli wystąpi błąd z połączeniem z bazą danych. Przycisk Clear również w tym oknie umożliwia wyczyszczenie wpisanych wartości. Po wprowadzeniu poprawnych danych i naciśnięciu przycisku Register rozpocznie się procedura rejestracji i użytkownik zostanie dodany do bazy danych. Okno zamknie się i znowu pojawi się okno logowania do którego należy zalogować się danymi z nowo utworzonego konta. Po zalogowaniu pojawia się główny panel systemu.

Obraz zawierający tekst, zrzut ekranu, Czcionka, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 7 Panel główny

Użytkownik ma możliwość dodania nowej notatki. Naciśnięcie przycisku Note spowoduje pojawienie się okna z notatką.

Obraz zawierający zrzut ekranu, tekst, komputer, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 8 Okno notatki

Użytkownik ma możliwość dodania kolejnej notatki z poziomu tego okna. Naciśnięcie przycisku „+” spowoduje pojawienie się kolejnego okna z notatką. Naciśnięcie przycisku „-„ spowoduje zamknięcie się okna. Użytkownik może wpisać swoją notatkę w polu tekstowym. Aby ją zapisać należy nacisnąć przycisk Save – wtedy notatka zostaje zapisana w bazie danych. Aby zaktualizować notatkę należy nacisnąć przycisk Update. Uwaga! Jeżeli użytkownik naciśnie wielokrotnie przycisk Save nie zmieniając zawartości notatki, spowoduje to zapisanie się kilku takich samych notatek – jest to sytuacja niepożądana. Przycisk Delete powoduje usunięcie notatki z bazy danych. Przycisk powinno się nacisnąć tylko jeśli dana notatka jest już zapisana w bazie. Przycisk Export to file powoduje zapisanie się notatki do pliku csv, który tworzy się w katalogu projektu.

Główny panel umożliwia również pokazanie wszystkich notatek. Naciskając przycisk Show All użytkownik spowoduje, że wyświetlą się okna z wszystkimi notatkami z bazy danych jakie są przypisane do jego konta. Notatki te mają takie same funkcjonalności jak notatki nowo utworzone. Przycisk Import from file powoduje zaimportowanie notatek z pliku csv, który znajduje się w katalogu projektu. Notatki te będą zapisane w bazie dopiero po kliknięciu przycisku Save w oknie każdej notatki. Przycisk Log out służy do zamknięcia aplikacji. Po uruchomieniu użytkownik będzie musiał ponownie podać dane logowania.

1. **Podsumowanie**

Dzięki zrealizowaniu projektu powstało użyteczne narzędzie System do zarządzania notatkami. Jest to pierwsza wersja tego systemu. Planowane kolejne wersje będą zawierać obsługę języka polskiego oraz funkcjonalność listy zadań.

1. **Literatura**
2. Materiały udostępnione przez prowadzącą na zajęciach z Programowania Obiektowego.
3. <https://www.elephantsql.com/docs/index.html>
4. <https://docs.oracle.com/en/java/>
5. <https://www.baeldung.com/java-jdbc>