

WYZSZA SZKOŁA INFORMATYKI I ZARZADZANIA KOLEGIUM INFORMATYKI STOSOWANEJ

Kierunek: INFORMATYKA

Projekt: System Wypożyczania Rzeczy

Artemii Snitko w68316

Contents

| | | |
|----------|---|----------|
| 1 | Opis założeń projektu | 1 |
| 1.1 | Platforma | 2 |
| 1.2 | Baza Danych | 2 |
| 1.3 | Funkcje Systemu | 2 |
| 1.4 | Wymagania Niefunkcjonalne | 2 |
| 1.5 | Cele Projektu | 2 |
| 2 | Wymagania Funkcjonalne i Niefunkcjonalne | 2 |
| 2.1 | Wymagania Funkcjonalne | 2 |
| 2.2 | Wymagania Niefunkcjonalne | 3 |
| 3 | Klasy | 3 |
| 3.1 | Klasa AdminForm | 3 |
| 3.1.1 | Metody | 3 |
| 3.2 | Klasa Form1 | 3 |
| 3.2.1 | Metody i Zdarzenia | 3 |
| 3.3 | Klasa Form2 | 3 |
| 3.3.1 | Metody i Zdarzenia | 3 |
| 3.4 | Klasa Item | 4 |
| 3.4.1 | Metody | 4 |
| 3.5 | Klasa DbManager | 4 |
| 3.5.1 | Metody | 4 |
| 3.6 | Klasa Customer | 4 |
| 3.6.1 | Metody | 4 |
| 3.7 | Klasa Transaction | 4 |
| 3.7.1 | Metody | 4 |

1 Opis założeń projektu

Projekt "System Wypożyczania Rzeczy" ma na celu stworzenie kompleksowego systemu umożliwiającego efektywne zarządzanie procesem wypożyczania różnorodnych przedmiotów. System ten został zaprojektowany z myślą o zautomatyzowaniu oraz usprawnieniu zarządzania wypożyczalnią, eliminując manualne procedury i usprawniając dostępność informacji.

1.1 Platforma

System będzie oparty na platformie Windows Forms, co umożliwi łatwą integrację z systemami operacyjnymi Windows. Język programowania: C#.

1.2 Baza Danych

Używamy systemu zarządzania bazą danych PostgreSQL dla trwałego przechowywania informacji o dostępnych przedmiotach, klientach, transakcjach, itp. Struktura bazy danych będzie zgodna z zasadami normalizacji, co zapewni spójność i integralność danych.

1.3 Funkcje Systemu

- Rejestracja i zarządzanie użytkownikami.
- Przeglądanie dostępnych przedmiotów.
- Proces wypożyczania i zwracania.
- System oceniania.

1.4 Wymagania Niefunkcjonalne

- Wygodne Interfejsy Użytkownika.
- Bezpieczeństwo Danych.
- Usprawnienie Procesów.

1.5 Cele Projektu

- Usprawnienie procesów związanych z zarządzaniem wypożyczalnią.
- Poprawa doświadczenia użytkownika.
- Bezpieczeństwo danych.
- Rozwój platformy.

2 Wymagania Funkcjonalne i Niefunkcjonalne

2.1 Wymagania Funkcjonalne

1. Rejestracja Użytkownika.
2. Logowanie.
3. Dodawanie Nowych Przedmiotów.
4. Przeglądanie Dostępnych Przedmiotów.
5. Zmiana Stanu Przedmiotu.
6. Usuwanie Przedmiotów.
7. Zmiana Danych Użytkownika.
8. Obsługa Ról Użytkowników.

2.2 Wymagania Niefunkcjonalne

1. Bezpieczeństwo Danych.
2. Wydajność.
3. Intuicyjny Interfejs Użytkownika.
4. Rozszerzalność.
5. Zarządzanie Błędami.
6. Komunikacja z Baza Danych.
7. Zgodność Zasad Bezpieczeństwa.

3 Klasy

3.1 Klasa AdminForm

Klasa AdminForm reprezentuje interfejs graficzny dla użytkownika z uprawnieniami administracyjnymi w systemie.

3.1.1 Metody

- `button1_Click`: Obsługuje zdarzenie kliknięcia przycisku "Dodaj przedmiot".
- `button3_Click`: Obsługuje zdarzenie kliknięcia przycisku "Wczytaj dane".
- `button2_Click`: Obsługuje zdarzenie kliknięcia przycisku "Usuń przedmiot".
- `button4_Click`: Obsługuje zdarzenie kliknięcia przycisku "Zmień przedmiot".

3.2 Klasa Form1

Klasa Form1 to główna klasa interfejsu graficznego, reprezentująca formularz logowania użytkownika do systemu.

3.2.1 Metody i Zdarzenia

- `button1_Click`: Sprawdza dane logowania użytkownika.
- `button2_Click`: Obsługuje proces rejestracji nowego użytkownika.
- `button3_Click`: Przełącza między trybem logowania a rejestracji.

3.3 Klasa Form2

Klasa Form2 reprezentuje interfejs użytkownika po zalogowaniu.

3.3.1 Metody i Zdarzenia

- `button1_Click`: Ładuje dane dotyczące przedmiotów.
- `button2_Click`: Zmienia przynależność przedmiotu do zalogowanego użytkownika.
- `Form2_FormClosed`: Obsługuje zamknięcie formularza.

3.4 Klasa Item

Klasa Item reprezentuje przedmiot w systemie.

3.4.1 Metody

- `Insertdatas`: Dodaje nowy przedmiot do bazy danych.
- `DeleleItem`: Usuwa przedmiot z bazy danych.
- `ChangeItem`: Modyfikuje dane przedmiotu w bazie danych.

3.5 Klasa DbManager

Klasa DbManager zarządza połączeniem z baza danych PostgreSQL.

3.5.1 Metody

- `ChangeValue`: Aktualizuje wartość w tabeli.
- `GetValueByColumn`: Pobiera wartość z tabeli na podstawie warunku.
- `InsertRecordUser`, `InsertRecordItem`: Dodaje nowy rekord do tabeli użytkowników lub przedmiotów.
- `LoadDataIntoDataGridView`: Ładuje dane z tabeli przedmiotów do kontrolki DataGridView.
- `DeleteRecordById`: Usuwa rekord z tabeli na podstawie identyfikatora.

3.6 Klasa Customer

Klasa Customer reprezentuje klienta w systemie.

3.6.1 Metody

- `Insertdatas`: Dodaje nowego klienta do bazy danych.
- `FoundId`, `FoundRole`, `FoundPass`: Pobiera identyfikator, rolę i hasło klienta.

3.7 Klasa Transaction

Klasa Transaction reprezentuje transakcje w systemie wypożyczania.

3.7.1 Metody

- `AddTransaction`: Dodaje nową transakcję do bazy danych.
- `GetTransactionDetails`: Pobiera szczegóły transakcji.
- `CalculatePenalty`: Oblicza kary za zwrot po terminie.