

Dokumentacja projektu Dziennika

1. Wstęp

GradeFlow to elektroniczny dziennik szkolny działający lokalnie, zaprojektowany w celu usprawnienia zarządzania procesami dydaktycznymi i administracyjnymi bez konieczności połączenia z internetem. System umożliwia administratorom zakładanie i konfigurację kont użytkowników, nauczycielom wprowadzanie ocen, rejestrację frekwencji oraz dodawanie uwag, a uczniom podgląd własnych wyników i planu lekcji. Projekt dedykowany jest szkołom podstawowym i średnim.

2. Założenia funkcjonalne

1. Logowanie i autoryzacja

- **Lokalna autentykacja użytkowników z podziałem na trzy role: Administrator, Nauczyciel, Uczeń.**

2. Zarządzanie kontami użytkowników

- **Operacje CRUD (ang. Create, Read, Update, Delete) na kontach uczniów, nauczycieli i administratorów, czyli tworzenie, odczyt, aktualizacja i usuwanie kont.**
- **Przypisywanie użytkowników do ról i klas.**

3. Zarządzanie planem lekcji

- **Definiowanie klas, przedmiotów, godzin lekcyjnych.**
- **Przydział przedmiotów do nauczycieli i klas.**
- **Nauczyciel może przeglądać swój własny plan zajęć.**

4. Wprowadzanie i zarządzanie ocenami

- **Dodawanie, edytowanie i usuwanie ocen o różnych typach.**
- **Automatyczne obliczanie średnich ocen.**

5. Rejestracja frekwencji

- **Oznaczanie obecności, nieobecności i spóźnień.**
- **Generowanie raportów frekwencji.**

6. Uwagi i notatki nauczycieli

- **Wprowadzanie komentarzy dotyczących uczniów.**
- **Filtrowanie uwag według daty lub autora.**
- **Uczeń może przeglądać otrzymane uwagi.**

7. Przegląd danych

- Lokalny podgląd ocen, frekwencji, planu lekcji i uwag. Założenia нефunkcjonalne
- Tryb offline: aplikacja działa w całości lokalnie, bez wsparcia połączeń sieciowych.
- Prosty, przejrzysty interfejs: GUI oparte na Qt Widgets, minimalna liczba kroków do wykonania podstawowych operacji.
- Przechowywanie danych w plikach: oceny, konta i frekwencja zapisywane w formacie DB w katalogu aplikacji.

4. Wymagania sprzętowe i środowiskowe

- System operacyjny: Windows
- Procesor: dowolny
- Pamięć RAM: min 50MB.
- Dysk: co najmniej 50 MB wolnej przestrzeni na pliki danych.

5. Technologie

- Język programowania: C++
- GUI: Qt Widgets
- Baza danych: SQLite 3

6. Architektura systemu

Aplikacja GradeFlow zbudowana jest w trzech warstwach:

1. Warstwa prezentacji (GUI)

- Okna i komponenty w Qt:
 - Login (login.h/.cpp)
 - Admin (admin.h/.cpp)
 - Nauczyciel (nauczyciel.h/.cpp)
 - Uczeń (uczen.h/.cpp)
- Dziedziczenie po QMainWindow (oraz QDialog dla okna logowania) oraz użycie kontrolki Qt (QTableView, QPushButton, QInputDialog, QMessageBox).
- Sygnały i sloty do obsługi zdarzeń użytkownika oraz przekazywania wyników.

2. Warstwa logiki aplikacji

- Obsługa zdarzeń GUI bezpośrednio w klasach okien:
 - Login: przycisk zaloguj loguje użytkownika po sprawdzeniu zgodności danych
 - Admin: przyciski wykonują operacje CRUD.
 - Nauczyciel:
 - przyciski do: zarządzania ocenami, rejestracji frekwencji, dodawania uwag,
 - przeglądania planu lekcji (db_schedule_list).
 - Uczeń: przyciski wywoływania funkcji do listowania (db_studgrd_list, db_studattndc_list, db_schd_list, db_note_list) i prezentacja w widokach.

3. Warstwa dostępu do danych (SQLite)

- Pojedynczy plik bazy gradeflow_db.db.
- Inicjalizacja bazy i schematu tabel:
 - db_create() tworzy tabele: UsrTypes, Users, Classes, Subjects, Schedules, GradeTypes, Grades, Attendance, Notes.
- Weryfikacja użytkownika:
 - int db_check wykonuje zapytanie SQL i zwraca id roli i użytkownika.
- Operacje CRUD
 - Funkcje db_<entity>_list(), on_<rola>Add*Btn_clicked(), on_<rola>Del*Btn_clicked(), itp. zaimplementowane w odpowiednich plikach .cpp.
- Zapytania wykonywane przy użyciu QSqlQuery, a wyniki prezentowane przez QSqlQueryModel.
- Klasy Login, Admin, Nauczyciel i Uczeń ma swoją zaprzyjaźnioną klasę <nazwa>Database, która ma na celu obsługiwać funkcje bazy danych

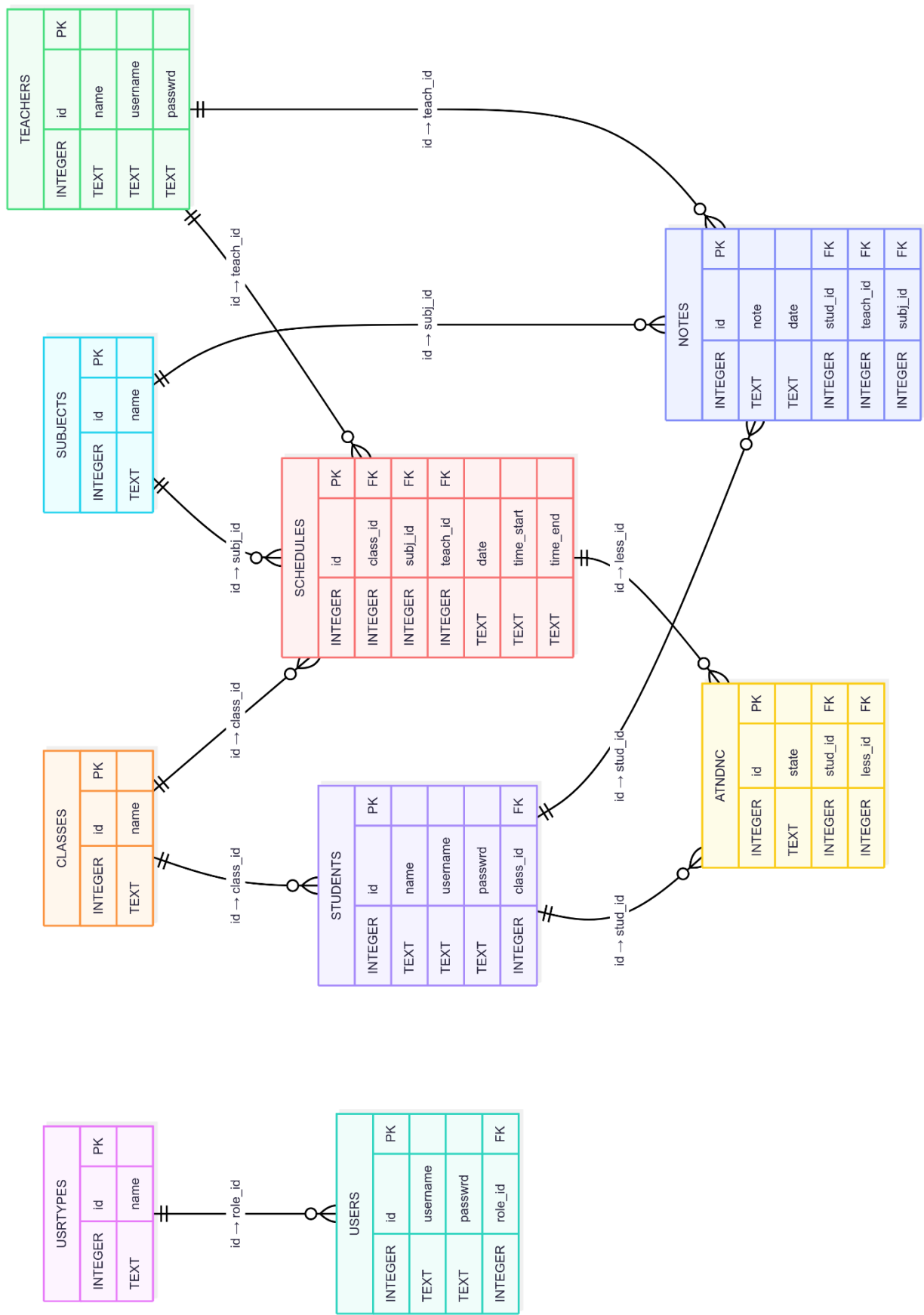
Komunikacja między warstwami odbywa się poprzez wywoływanie funkcji db_* w odpowiedzi na akcje użytkownika oraz mechanizm sygnałów i slotów Qt.

7. Struktura bazy danych

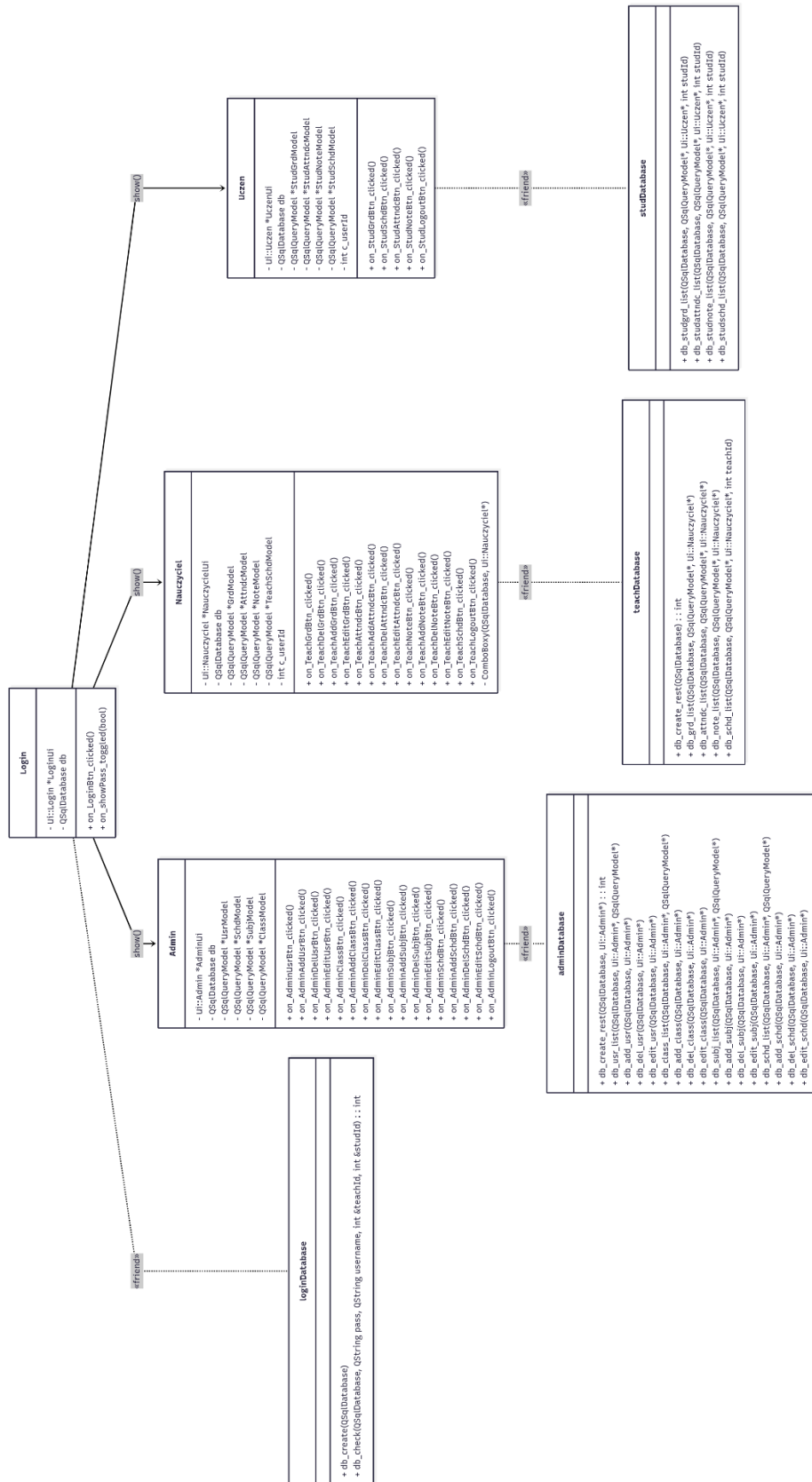
Pojedynczy plik SQLite gradeflow_db.db zawiera tabele:

Tables		
Atndnc		
id	INTEGER	"id" INTEGER
state	TEXT	"state" TEXT NOT NULL
stud_id	INTEGER	"stud_id" INTEGER
less_id	INTEGER	"less_id" INTEGER
Classes		
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL
GradeTypes		
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL
Grades		
id	INTEGER	"id" INTEGER
val	REAL	"val" REAL NOT NULL
date	TEXT	"date" TEXT NOT NULL
stud_id	INTEGER	"stud_id" INTEGER
teach_id	INTEGER	"teach_id" INTEGER
subj_id	INTEGER	"subj_id" INTEGER
type_id	INTEGER	"type_id" INTEGER
Notes		
id	INTEGER	"id" INTEGER
note	TEXT	"note" TEXT NOT NULL
date	TEXT	"date" TEXT NOT NULL
stud_id	INTEGER	"stud_id" INTEGER
teach_id	INTEGER	"teach_id" INTEGER
subj_id	INTEGER	"subj_id" INTEGER
Schedules		
id	INTEGER	"id" INTEGER
dotw	TEXT	"dotw" TEXT NOT NULL
hour	TEXT	"hour" TEXT NOT NULL
subj_id	INTEGER	"subj_id" INTEGER
teach_id	INTEGER	"teach_id" INTEGER
class_id	INTEGER	"class_id" INTEGER
Students		
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL
surname	TEXT	"surname" TEXT NOT NULL
role_id	INTEGER	"role_id" INTEGER
user_id	INTEGER	"user_id" INTEGER
class_id	INTEGER	"class_id" INTEGER
Subjects		
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL
TeachSubjs		
teach_id	INTEGER	"teach_id" INTEGER
subj_id	INTEGER	"subj_id" INTEGER
Teachers		
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL
surname	TEXT	"surname" TEXT NOT NULL
role_id	INTEGER	"role_id" INTEGER
user_id	INTEGER	"user_id" INTEGER
Users		
id	INTEGER	"id" INTEGER
username	TEXT	"username" TEXT NOT NULL
passwd	TEXT	"passwd" TEXT NOT NULL
role_id	INTEGER	"role_id" INTEGER
UsrTypes		
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL

Relacje między tabelami:



8. Opis klas i modułów



8.1. LoginWindow (login.h/.cpp)

- Przycisk **zaloguj** wywołuje funkcję `db_create`, która tworzy bazę danych, jeśli nie istnieje wczytując przykładowych użytkowników.
- Następnie wywoływana jest funkcja `db_check`, która sprawdza czy dane wprowadzone przez użytkownika są zapisane w bazie danych.
- Jeżeli dane się zgadzają, otwierane jest odpowiednie okno w zależności od roli użytkownika.

8.2. AdminWindow (admin.h/.cpp)

- **Sidebar:**
 - Przycisk **Użytkownicy** wywołuje funkcję `db_create_rest`, uzupełnia bazę danych o tabele `Classes`, `Subjects` itp., jeśli one nie istnieją. Następnie wywoływana jest funkcja `db_usr_list`, która przygotowuje format tabeli wyświetlania użytkowników. Końcowo wyświetla się tabela z załadowanymi danymi.
 - Przycisk **Klasy** wywołuje funkcję `db_create_rest`, uzupełnia bazę danych o tabele `Classes`, `Subjects` itp., jeśli one nie istnieją. Następnie wywoływana jest funkcja `db_class_list`, która przygotowuje format tabeli wyświetlania klas. Końcowo wyświetla się tabela z załadowanymi danymi.
 - Przycisk **Przedmioty** wywołuje funkcję `db_create_rest`, uzupełnia bazę danych o tabele `Classes`, `Subjects` itp., jeśli one nie istnieją. Następnie wywoływana jest funkcja `db_subj_list`, która przygotowuje format tabeli wyświetlania przedmiotów. Końcowo wyświetla się tabela z załadowanymi danymi.
 - Przycisk **Plan zajęć** wywołuje funkcję `db_create_rest`, uzupełnia bazę danych o tabele `Classes`, `Subjects` itp., jeśli one nie istnieją. Następnie wywoływana jest funkcja `db_schd_list`, która przygotowuje format tabeli wyświetlania lekcji. Końcowo wyświetla się tabela z załadowanymi danymi.
- **Panel zarządzania użytkownikami.**
 - Przycisk **Dodaj użytkownika** otwiera okna dialogowe, które wprowadzają informacje do bazy danych. Następnie wywoływana jest funkcja `db_usr_list`, która ma na celu odświeżyć wyświetlaną tabelę.
 - Przycisk **Usuń użytkownika** wywołuje okno dialogowe, które ma na celu potwierdzić czy usunięcie użytkownika ma się wydarzyć. Następnie usuwany jest zapis użytkownika z bazy danych.
 - Przycisk **Edytuj użytkownika** otwiera okna dialogowe, które zamieniają informacje w bazie danych.
- **Panel zarządzania klasami.**

- Przycisk Dodaj klasę otwiera okna dialogowe, które wprowadzają informacje do bazy danych. Następnie wywoływana jest funkcja `db_class_list`, która ma na celu odświeżyć wyświetlaną tabelę.
- Przycisk Usuń klasę wywołuje okno dialogowe, które ma na celu potwierdzić czy usunięcie użytkownika ma się wydarzyć. Następnie usuwany jest zapis klasy z bazy danych.
- Przycisk Edytuj klasę otwiera okna dialogowe, które zamieniają informacje w bazie danych.
- Panel zarządzania przedmiotami.
 - Przycisk Dodaj przedmiot otwiera okna dialogowe, które wprowadzają informacje do bazy danych. Następnie wywoływana jest funkcja `db_subj_list`, która ma na celu odświeżyć wyświetlaną tabelę.
 - Przycisk Usuń przedmiot wywołuje okno dialogowe, które ma na celu potwierdzić czy usunięcie przedmiotu ma się wydarzyć. Następnie usuwany jest zapis przedmiotu z bazy danych.
 - Przycisk Edytuj przedmiot otwiera okna dialogowe, które zamieniają informacje w bazie danych.
- Panel zarządzania planem zajęć.
 - Przycisk Dodaj lekcję otwiera okna dialogowe, które wprowadzają informacje do bazy danych. Następnie wywoływana jest funkcja `db_schd_list`, która ma na celu odświeżyć wyświetlaną tabelę.
 - Przycisk Usuń lekcję wywołuje okno dialogowe, które ma na celu potwierdzić czy usunięcie lekcji ma się wydarzyć. Następnie usuwany jest zapis lekcji z bazy danych.
 - Przycisk Edytuj lekcję otwiera okna dialogowe, które zamieniają informacje w bazie danych.

8.3. TeacherWindow (nauczyciel.h/.cpp)

- Sidebar:
 - Przycisk Oceny wywołuje funkcje `db_grd_list`, która przygotowuje model tabeli ocen. Końcowo wyświetla się tabela z załadowanymi danymi.
 - Przycisk Frekwencja wywołuje funkcje `db_atndc_list`, która przygotowuje format tabeli wyświetlania frekwencji. Końcowo wyświetla się tabela z załadowanymi danymi.
 - Przycisk Uwagi wywołuje funkcje `db_note_list`, która przygotowuje format tabeli wyświetlania przedmiotów. Końcowo wyświetla się tabela z załadowanymi danymi.

- Przycisk Plan zajęć wywołuje db_schd_list, która przygotowuje format tabeli wyświetlania lekcji. Końcowo wyświetla się tabela z załadowanymi danymi.
- Panel zarządzania ocenami.
 - Przycisk Dodaj ocenę otwiera okna dialogowe, które wprowadzają informacje do bazy danych. Następnie wywoływana jest funkcja db_grd_list, która ma na celu odświeżyć wyświetlaną tabelę.
 - Przycisk Usuń ocenę wywołuje okno dialogowe, które ma na celu potwierdzić czy usunięcie oceny ma się wydarzyć. Następnie usuwany jest zapis oceny z bazy danych.
 - Przycisk Edytuj ocenę otwiera okna dialogowe, które zamieniają informacje w bazie danych.
- Panel zarządzania frekwencją.
 - Przycisk Dodaj obecność otwiera okna dialogowe, które wprowadzają informacje do bazy danych. Następnie wywoływana jest funkcja db_attn dc_list, która ma na celu odświeżyć wyświetlaną tabelę.
 - Przycisk Usuń obecność wywołuje okno dialogowe, które ma na celu potwierdzić czy usunięcie obecności ma się wydarzyć. Następnie usuwany jest zapis obecności z bazy danych.
 - Przycisk Edytuj obecność otwiera okna dialogowe, które zamieniają informacje w bazie danych.
- Panel zarządzania uwagami.
 - Przycisk Dodaj uwagę otwiera okna dialogowe, które wprowadzają informacje do bazy danych. Następnie wywoływana jest funkcja db_note_list, która ma na celu odświeżyć wyświetlaną tabelę.
 - Przycisk Usuń uwagę wywołuje okno dialogowe, które ma na celu potwierdzić czy usunięcie uwagi ma się wydarzyć. Następnie usuwany jest zapis uwagi z bazy danych.
 - Przycisk Edytuj uwagę otwiera okna dialogowe, które zamieniają informacje w bazie danych.
- Panel przeglądu planu zajęć.
 - Przycisk Plan zajęć wywołuje db_schd_list, która przygotowuje format tabeli wyświetlania planu zajęć. Po stworzeniu modelu tabeli, jest ona wyświetlana na ekranie

8.4. StudentWindow (uczen.h/.cpp)

- Wyświetlanie ocen, frekwencji, uwag i planu.
 - Przycisk Oceny/Frekwencja/Uwagi/Plan Zajęć wywołuje odpowiednio funkcję db_studgrd_list/db_studattn dc_list/db_studnotes_list/db_studschd_list, która

przygotowuje format tabeli wyświetlania ocen/frekwencji/uwag/planu zajęć. Po stworzeniu modelu tabeli, jest ona wyświetlana na ekranie.

9. Instrukcja obsługi – podstawowe scenariusze

1. Logowanie

- Logowanie jako administrator:
 - login: admin
 - hasło: 123
- Logowanie jako nauczyciel: można stworzyć konto za pomocą panelu admina, i w ten sam sposób się zalogować.
- Logowanie jako uczeń: można stworzyć konto za pomocą panelu admina, i w ten sam sposób się zalogować.

2. Zarządzanie użytkownikami, klasami, przedmiotami, planem zajęć (Administrator)

- Wybierz Użytkownicy/Klasy/Przedmioty/Plan zajęć → Dodaj * , wypełnij dane, zatwierdź.
- Wybierz Użytkownicy/Klasy/Przedmioty/Plan zajęć → zaznacz pozycję w tabeli → Usuń * , potwierdź.
- Wybierz Użytkownicy/Klasy/Przedmioty/Plan zajęć → Edytuj * , wypełnij dane, zatwierdź.

3. Zarządzanie ocenami, frekwencją, uwagami (Nauczyciel)

- Przejdź do Oceny/Frekwencja/Uwagi → Dodaj * , uzupełnij formularz, potwierdź.
- Wybierz Oceny/Frekwencja/Uwagi → zaznacz pozycję w tabeli → Usuń * , potwierdź.
- Wybierz Oceny/Frekwencja/Uwagi → Edytuj * , uzupełnij formularz, zatwierdź.

4. Podgląd ocen, planu zajęć, frekwencji, uwag (Uczeń)

- Wejdź w Oceny/Plan zajęć/Frekwencja/Uwagi.