**TRƯỜNG ĐẠI HỌC AN GIANG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN**

**Chuyên đề Python (COS525)**

**Nhóm - 18**

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ BÀI HÁT**

**VỚI PYTHON, TKINTER VÀ SQL SERVER**

**Giảng viên hướng dẫn:**

**Ths. Nguyễn Ngọc Minh**

**Sinh viên thực hiện:**

**Lê Phúc Đức Hiền – DTH235650**

**Phạm Hoàng Giang – DTH235636**

**An Giang, Tháng 9, Năm 2025**

|  |  |
| --- | --- |
| Họ và tên | Công việc thực hiện |
| Lê Phúc Đức Hiền | Code, Thiết kế Form |
| Phạm Hoàng Giang | Word, SQL |

### LỜI NÓI ĐẦU

**1. Tính Cần Thiết của Đề Tài**

Trong kỷ nguyên số, việc quản lý và truy cập kho dữ liệu âm nhạc cá nhân ngày càng trở nên quan trọng. Các ứng dụng hiện tại thường đòi hỏi tài nguyên hệ thống lớn hoặc phụ thuộc vào kết nối Internet liên tục . Nhu cầu về một ứng dụng gọn nhẹ, hiệu suất cao, có thể quản lý bài hát một cách trực quan trên máy tính cá nhân chạy Windows là cần thiết. Đề tài "Xây dựng Ứng dụng Quản lý Bài hát với Python, Tkinter và SQL Server" ra đời nhằm đáp ứng nhu cầu này. Ứng dụng tập trung vào việc tạo ra một giao diện người dùng thân thiện, trực quan, cho phép người dùng dễ dàng thêm, sửa, xóa, tìm kiếm và quản lý các tệp âm nhạc của mình.

**2. Mục Tiêu Nghiên Cứu**

Mục tiêu chính của đề tài là:

1. Vận dụng kiến thức ngôn ngữ lập trình Python và thư viện đồ họa Tkinter để xây dựng giao diện người dùng (GUI) hiện đại, trực quan, thể hiện qua các module như Đăng nhập, Menu chính, Quản lý Bài hát, Ca sĩ, Nhạc sĩ.
2. Thiết kế và triển khai Cơ sở dữ liệu sử dụng **Microsoft SQL Server** (thay thế cho các hệ quản trị mã nguồn mở khác) để lưu trữ thông tin bài hát (tên, ca sĩ, nhạc sĩ, thể loại, v.v.).
3. Tích hợp các chức năng quản lý dữ liệu cho phép người dùng tương tác hiệu quả với kho nhạc.
4. Xây dựng các tính năng bổ sung như đăng nhập phân quyền (Admin/User) và quản lý danh sách yêu thích cá nhân.

**3. Cấu Trúc Báo Cáo**

Báo cáo được trình bày gồm các phần chính sau :

1. Đặt Vấn Đề (Tính cần thiết, Mục tiêu)
2. Tổng Quan và Cơ Sở Lý Thuyết (Python, Tkinter, SQL Server)
3. Phân Tích, Thiết Kế và Triển Khai Ứng Dụng (Thiết kế CSDL, GUI, chi tiết mã nguồn CRUD)
4. Kết Quả Đạt Được và Đánh Giá (Đánh giá tính năng, hướng phát triển)

### 1. Đặt vấn đề

**1.1. Bối cảnh**

Trong kỷ nguyên số, nhu cầu về việc quản lý hiệu quả các tệp dữ liệu số cá nhân, đặc biệt là thư viện âm nhạc, ngày càng tăng cao. Việc thiếu một công cụ quản lý bài hát gọn nhẹ, trực quan và có khả năng tùy biến cao cho máy tính cá nhân là một hạn chế. Đề tài này nhằm xây dựng một ứng dụng giải quyết vấn đề đó bằng cách tận dụng những ưu điểm sau:

* **Ngôn ngữ Python:** Là ngôn ngữ lập trình đa năng, cú pháp rõ ràng, hỗ trợ mạnh mẽ cho việc phát triển ứng dụng desktop.
* **Thư viện Tkinter:** Là bộ công cụ chuẩn cho GUI của Python, cho phép xây dựng giao diện người dùng đơn giản, dễ tùy chỉnh, đáp ứng tiêu chí gọn nhẹ của ứng dụng.
* **Hệ quản trị CSDL SQL Server:** Được lựa chọn để đảm bảo tính ổn định, bảo mật và khả năng quản lý dữ liệu chặt chẽ trên môi trường Windows. Đây là hệ quản trị phù hợp để lưu trữ và truy vấn thông tin bài hát với các ràng buộc quan hệ phức tạp.

**1.2. Mục Tiêu Nghiên Cứu của Đề Tài**

Mục tiêu nghiên cứu được xác định một cách cụ thể, rõ ràng và có tính khả thi như sau:

1. Vận dụng thành thạo kiến thức về lập trình Python và thư viện Tkinter để xây dựng một ứng dụng desktop hoàn chỉnh.
2. Thiết kế và triển khai Cơ sở dữ liệu quan hệ (CSDL) trên hệ quản trị **SQL Server** để lưu trữ thông tin chi tiết của các đối tượng trong ứng dụng, bao gồm BaiHat, Album, CaSi, NhacSi và NguoiDung.
3. Triển khai các chức năng quản lý dữ liệu cơ bản (CRUD: Create, Read, Update, Delete) cho các đối tượng trong ứng dụng.
4. Xây dựng các tính năng bổ sung quan trọng như Tìm kiếm và Yêu thích, cũng như các tính năng Đăng nhập/Phân quyền để quản lý phiên người dùng.
5. Đảm bảo ứng dụng có khả năng kết nối ổn định giữa Python và SQL Server thông qua thư viện pyodbc.

### 2. Tổng quan và cơ sở lý thuyết

**2.1. Bối Cảnh Bài Toán**

Bài toán quản lý bài hát đặt ra trong đề tài này xuất phát từ nhu cầu kiểm soát và cá nhân hóa thư viện âm nhạc số. Nội dung bài toán đặt ra cần giải quyết bao gồm:

* Xây dựng một giao diện người dùng (GUI) hiện đại, trực quan bằng Tkinter.
* Quản lý metadata (siêu dữ liệu) của bài hát một cách có hệ thống (Tên bài hát, Ca sĩ, Nhạc sĩ, Thể loại, Album, Thời lượng).
* Đảm bảo tính ổn định và bảo mật của dữ liệu thông qua việc sử dụng hệ quản trị SQL Server.

**2.2. Phạm Vi của Đề Tài**

Đề tài tập trung giải quyết các vấn đề trong phạm vi sau:

* **Công nghệ sử dụng:** Giới hạn trong bộ ba Python, thư viện GUI Tkinter và HQTCSDL Microsoft SQL Server.
* **Chức năng chính:** Tập trung vào các chức năng cơ bản của một ứng dụng quản lý bài hát: CRUD (Thêm, Xem, Sửa, Xóa) cho thông tin Bài hát, Nhạc sĩ, Ca sĩ, Album.
* **Quản lý người dùng:** Bao gồm chức năng Đăng nhập để phân quyền Admin/User và quản lý danh sách yêu thích cá nhân.

**2.3. Phương Pháp Nghiên Cứu và Hướng Giải Quyết**

1. **Phương pháp Nghiên cứu Lý thuyết:** Nghiên cứu về kiến trúc ứng dụng desktop, mô hình CSDL quan hệ (SQL Server), và các thư viện Python chuyên dụng (pyodbc, tkcalendar).
2. **Phương pháp Phát triển Thực nghiệm:** Áp dụng mô hình phát triển lặp: Thiết kế CSDL -> Thiết kế GUI -> Viết mã kết nối CSDL và xử lý logic -> Kiểm thử và Lặp lại.

**2.4. Trình Bày Tóm Tắt Cơ Sở Lý Thuyết**

1. **Ngôn ngữ Lập trình Python:** Ngôn ngữ cốt lõi để xây dựng logic ứng dụng và xử lý dữ liệu.
2. **Thư viện Tkinter:** API chuẩn của Python để tạo ra ứng dụng giao diện người dùng đồ họa (GUI) desktop.
3. **Hệ quản trị CSDL SQL Server:** Là một HQTCSDL quan hệ mạnh mẽ của Microsoft. Trong đề tài này, SQL Server được dùng để lưu trữ bảng NguoiDung, BaiHat... Thư viện **pyodbc** được sử dụng để kết nối Python với SQL Server.

### 3. Kết quả đạt được và Demo Code

**3.1. Tổng Quan về Kết Quả Đạt Được**

Sau quá trình nghiên cứu và phát triển, đề tài đã hoàn thành thành công các mục tiêu đã đề ra, xây dựng được một ứng dụng quản lý bài hát trên nền tảng Python, Tkinter và SQL Server.

**3.2. Các Kết Quả Chi Tiết**

**3.2.1. Thiết Kế và Triển Khai Giao Diện Người Dùng (GUI)**

Triển khai thành công hệ thống đa cửa sổ với các form chính bao gồm: Đăng nhập, Menu Chính, Quản lý Bài hát, Quản lý Ca sĩ/Nhạc sĩ/Album. Sử dụng các control hiện đại như DateEntry (lịch), Combobox (danh sách chọn) và Treeview (bảng dữ liệu) giúp thao tác chính xác.

**3.2.2. Triển Khai Cơ Sở Dữ Liệu SQL Server**

* **Thiết lập CSDL:** Đã thiết lập thành công kết nối và cấu trúc dữ liệu trên SQL Server.
* **Cấu trúc Schema:** Xây dựng thành công các bảng chính :
  + BaiHat: Lưu thông tin chi tiết (Mã, Tên, Thời lượng, Lượt nghe...).
  + NguoiDung: Lưu tài khoản và phân quyền (Admin/User).
  + CaSi, NhacSi, Album, TheLoai: Các bảng danh mục chuẩn hóa.
  + DanhSachYeuThich: Lưu trữ bài hát yêu thích của từng user.

**3.2.3. Triển Khai Chức Năng Cốt Lõi**

* **Chức năng CRUD:** Đã triển khai đầy đủ các thao tác Thêm, Sửa, Lưu và Xóa cho tất cả các danh mục.
* **Phân quyền:** Xây dựng module Đăng nhập, tự động nhận diện quyền Admin (full quyền) hoặc User (hạn chế quyền) .
* **Tìm kiếm và Lọc:** Triển khai chức năng tìm kiếm bài hát theo tên và hiển thị chi tiết thông tin kèm tên Ca sĩ, Nhạc sĩ (thay vì mã số).

**3.3. Đánh Giá và Hướng Phát Triển**

* **Ưu điểm:** Ứng dụng hoạt động ổn định trên Windows, GUI trực quan. Việc sử dụng SQL Server đảm bảo tính toàn vẹn dữ liệu cao .
* **Hạn chế:** Chưa có tính năng phát nhạc chuyên nghiệp (Media Player) tích hợp sâu, chưa đồng bộ đám mây.
* **Hướng phát triển:** Tích hợp thư viện pygame để phát file nhạc mp3 thực tế, phát triển báo cáo thống kê lượt nghe .

### A. THIẾT KẾ GIAO DIỆN QUẢN LÝ BÀI HÁT

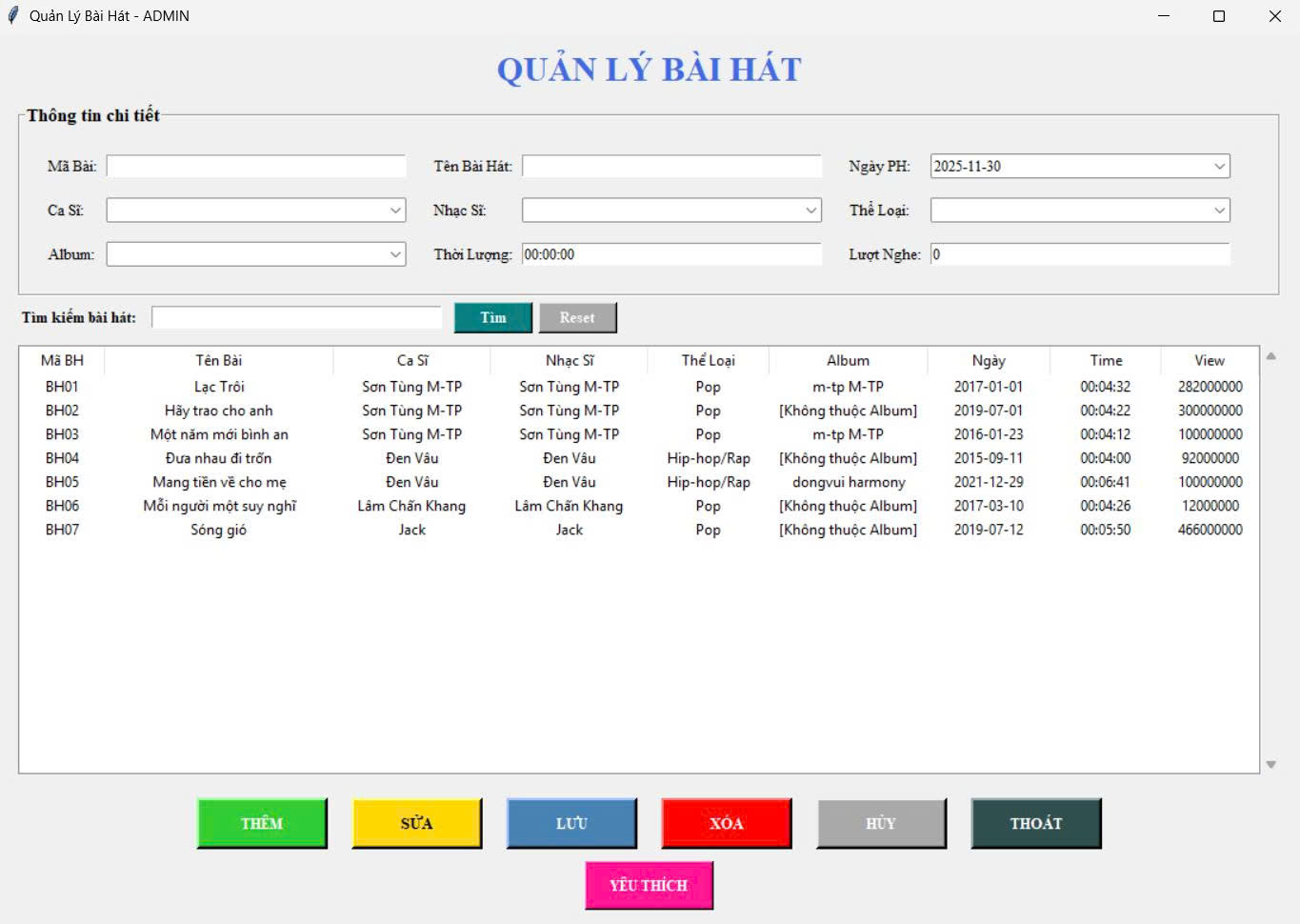
1. **Giao diện Đăng nhập:**



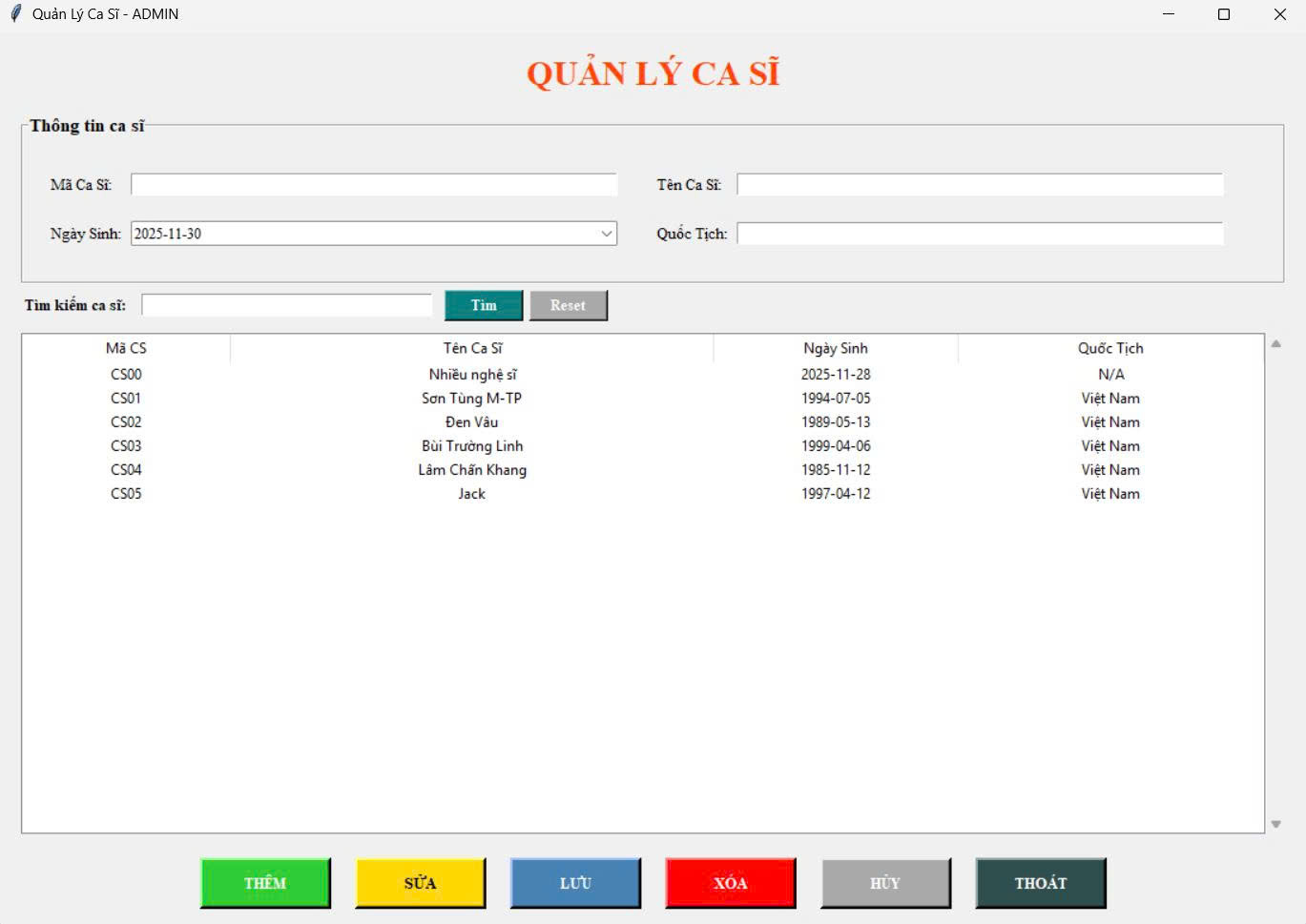
1. **Giao diện Menu:**



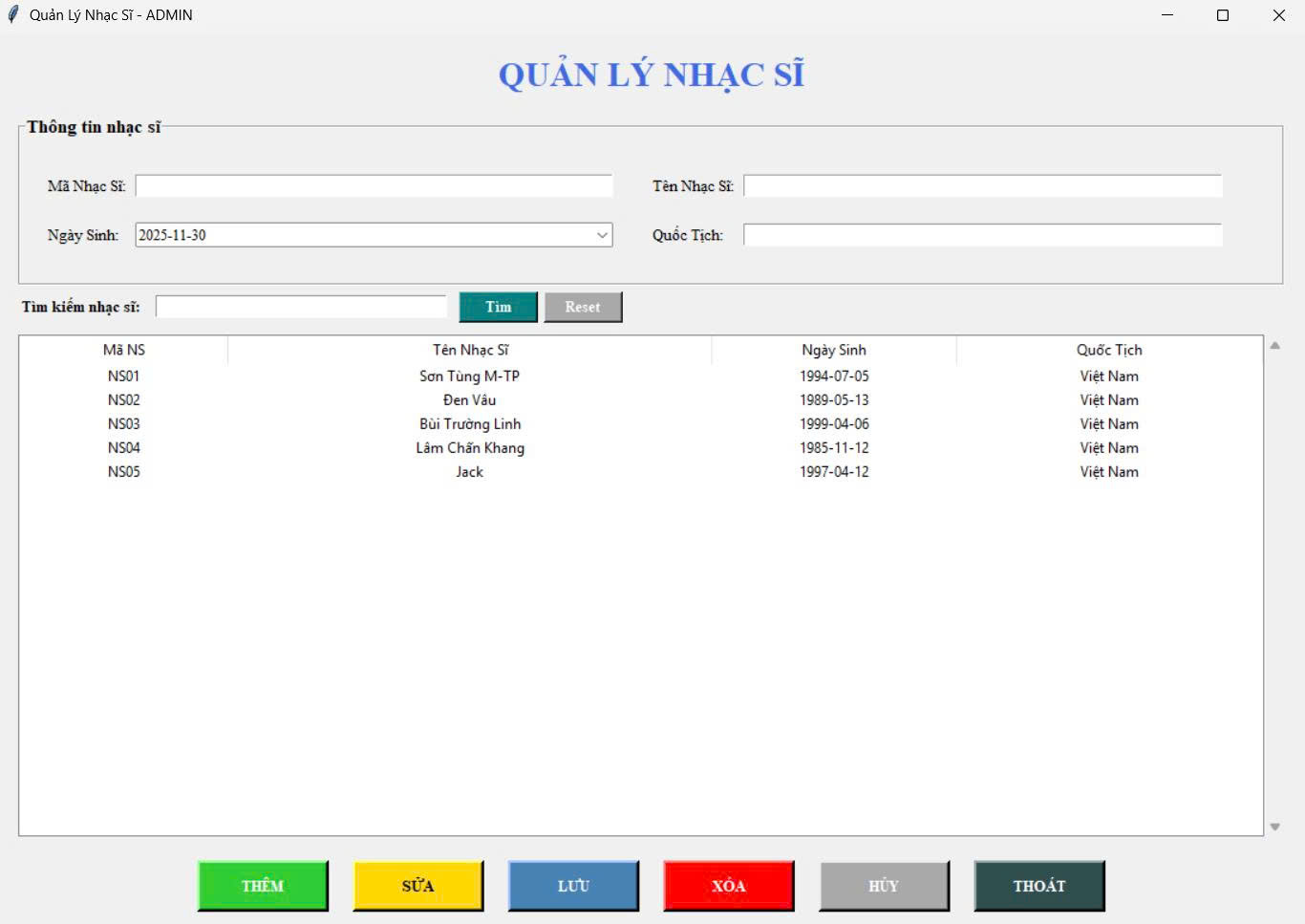
1. **Giao diện Quản lý Bài hát:**



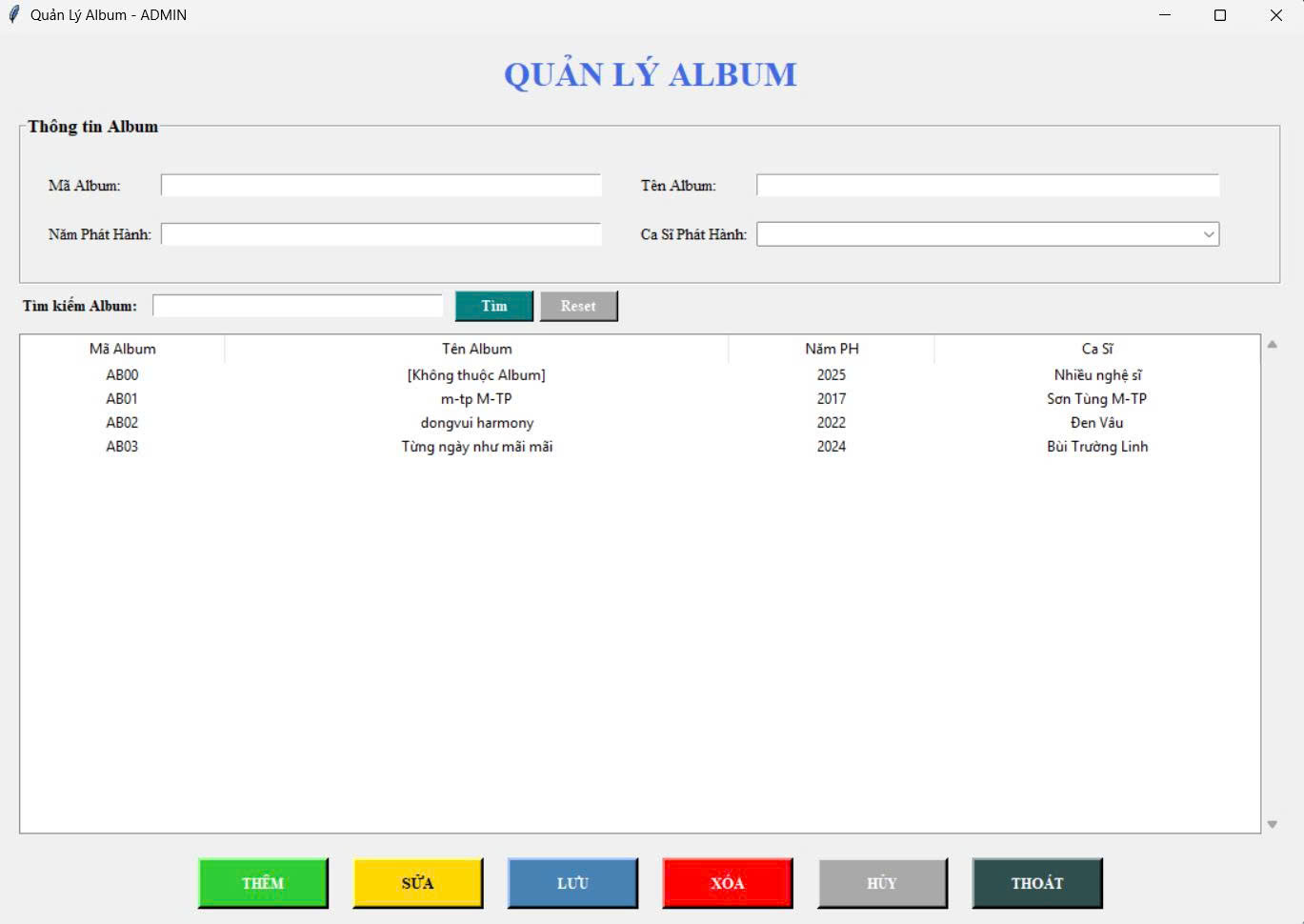
1. **Giao diện Quản lý Ca sĩ:**



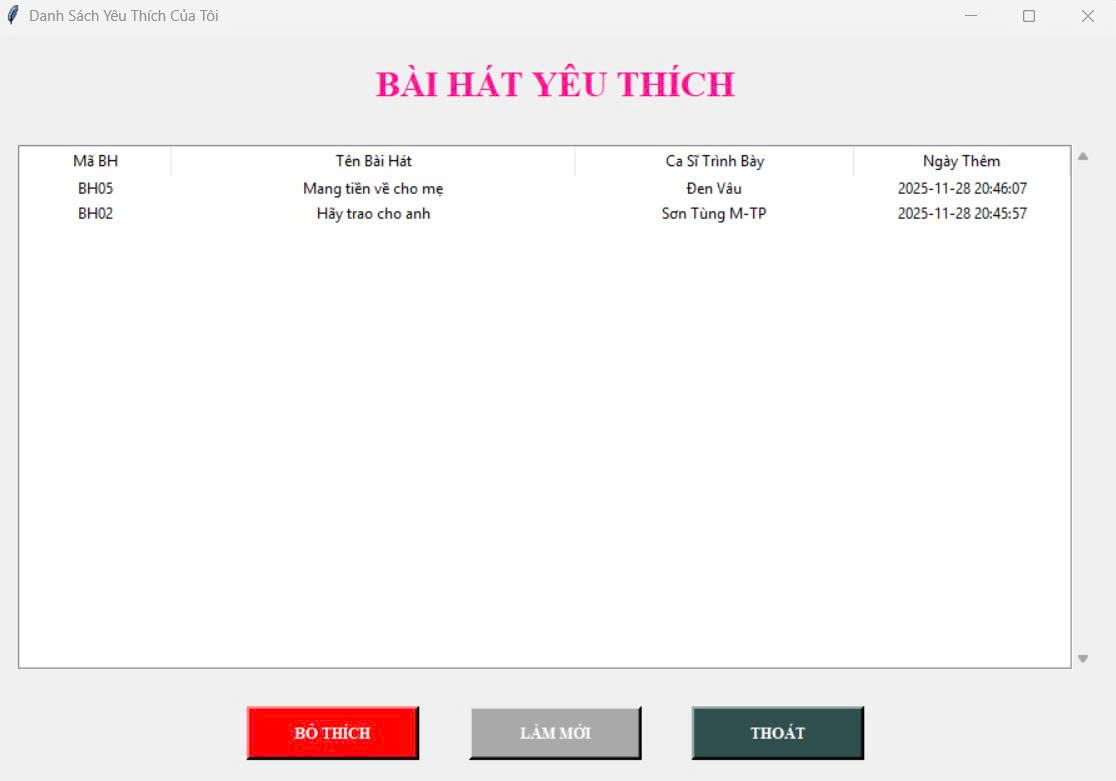
1. **Giao diện Quản lý Nhạc Sĩ:**



1. **Giao diện Quản lý Album:**



1. **Giao diện Bài hát yêu thích:**



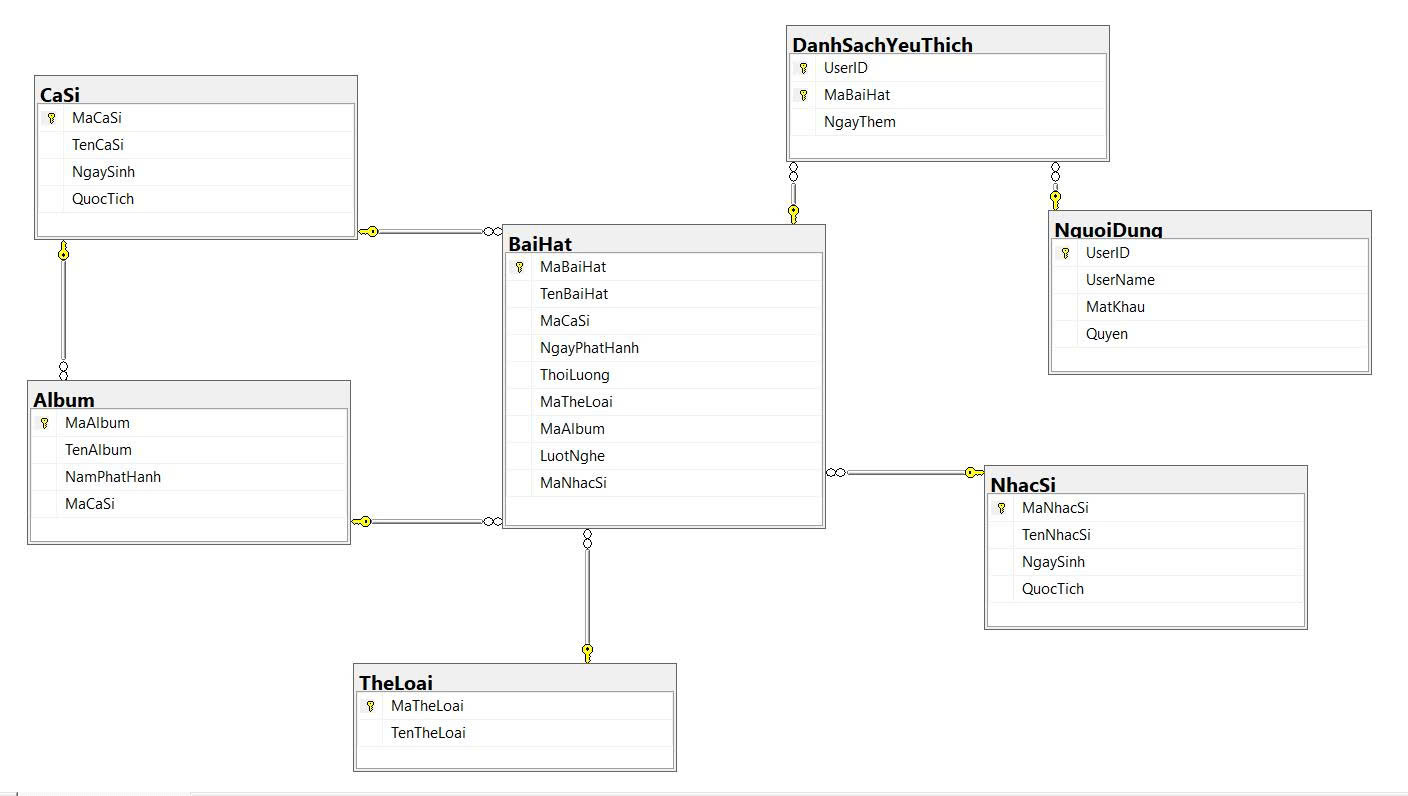
### B. CÁC BƯỚC XÂY DỰNG

Để xây dựng ứng dụng, đề tài đã thực hiện quy trình phát triển module hóa như sau:

**1. Chuẩn bị Môi trường và Công cụ Phát triển**

* **Cài đặt Python và SQL Server:** Đảm bảo máy tính đã cài đặt Python 3.x và SQL Server Express.
* **Cài đặt Thư viện:** Sử dụng lệnh pip install pyodbc tkcalendar để cài đặt thư viện kết nối CSDL và thư viện lịch .

**2. Thiết kế Cơ sở Dữ liệu (CSDL) SQL Server**



* **Tạo Database:** Tạo cơ sở dữ liệu QuanLyBaiHat.
* **Thiết kế Schema:** Xây dựng bảng BaiHat, NguoiDung, CaSi... và thiết lập các khóa ngoại (Foreign Key) để đảm bảo ràng buộc dữ liệu .

**3. Xây dựng Khung Giao diện Ứng dụng (GUI Framework)**

**Khởi tạo Cửa sổ:** Sử dụng tk.Tk() cho form đăng nhập và tk.Toplevel() cho các form chức năng con.

**4. Triển khai các Module Chức năng Chính**

* **Module Kết nối CSDL:** Viết hàm connect\_db() sử dụng pyodbc để kết nối tới LOCALHOST\SQLEXPRESS.
* **Module Đăng nhập:** Xây dựng logic kiểm tra tài khoản từ bảng NguoiDung. Nếu đúng, chuyển sang Menu kèm theo thông tin UserID và Quyen .
* **Module Quản lý Bài hát:**
  + Sử dụng Treeview hiển thị dữ liệu.
  + Dùng LEFT JOIN trong câu lệnh SQL để lấy tên Ca sĩ, Nhạc sĩ từ bảng liên quan.
  + Xử lý sự kiện nút Thêm/Sửa/Xóa và nút Yêu thích .

### C. HƯỚNG DẪN CÀI ĐẶT VÀ CODE ỨNG DỤNG

Phần này mô tả các bước cần thiết để thiết lập môi trường và cấu trúc mã nguồn đã triển khai .

**1. HƯỚNG DẪN CÀI ĐẶT MÔI TRƯỜNG**

**1.1. Cài đặt các Công cụ Nền tảng**

* **Python:** Cài đặt Python 3.8+.
* **SQL Server:** Cài đặt Microsoft SQL Server và SSMS.

**1.2. Cài đặt các Thư viện Python** Mở Terminal và chạy lệnh sau:

Bash

pip install pyodbc tkcalendar

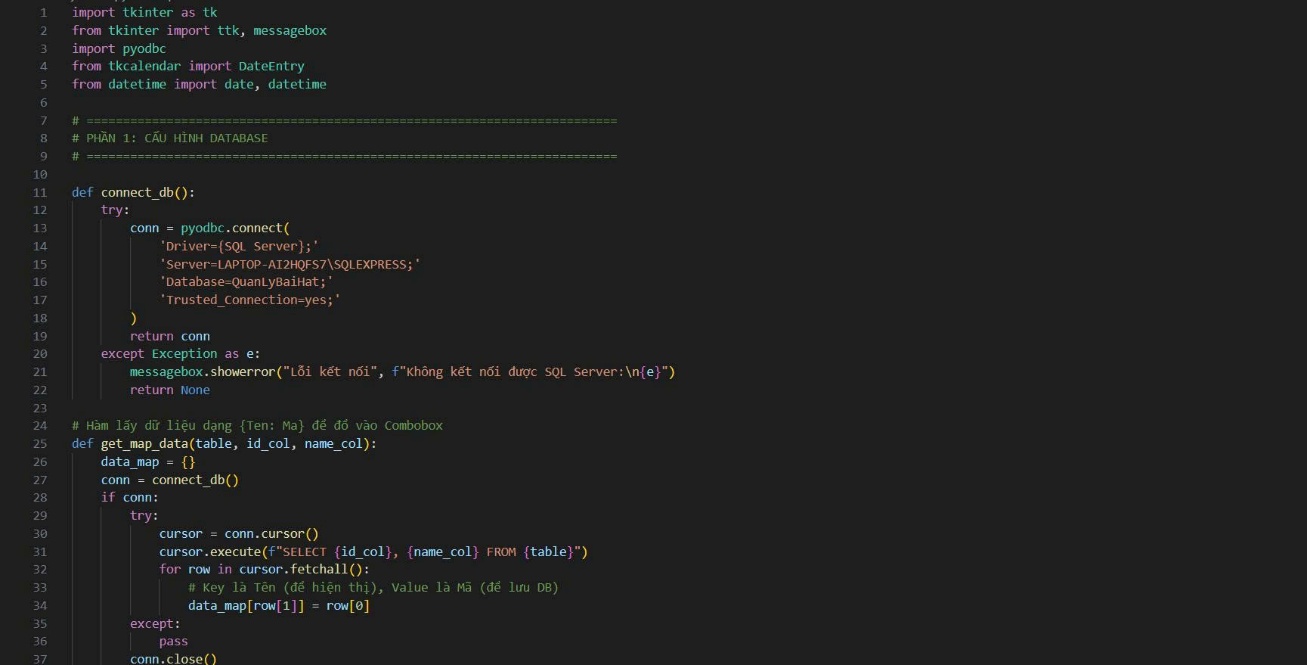
Thư viện pyodbc dùng để kết nối SQL Server, thay thế cho psycopg2 hay mysql-connector.

**1.3. Khởi tạo Cơ sở Dữ liệu (CSDL)**

* Mở SQL Server Management Studio (SSMS).
* Chạy file script QLBH.sql (kèm theo báo cáo này) để tạo Database QuanLyBaiHat và các bảng dữ liệu mẫu.

**2. CẤU TRÚC CODE ỨNG DỤNG VÀ CHỨC NĂNG**

**2.1. Kết nối CSDL**: Hàm connect\_db kết nối tới SQL Server và hàm get\_map\_data để lấy dữ liệu cho Combobox.



**2.2. Viết code ứng dụng với Tkin + SQL SERVER:**

