

TRƯỜNG ĐẠI HỌC AN GIANG
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN
Chuyên đề Python (COS525)

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ CÁC
TUYẾN DU LỊCH VỚI PYTHON, TKINTER VÀ
MYSQL**

Giảng viên hướng dẫn: Ths. Nguyễn Ngọc Minh

Sinh viên thực hiện: <DPM245504>.<Nguyễn Thanh Trí>.<25PM.Python_Nh4_To3>

1. Đặt vấn đề

1.1 Tính cần thiết của đề tài

Trong thời đại công nghệ số, nhu cầu du lịch ngày càng phát triển mạnh mẽ. Các công ty lữ hành cần một hệ thống giúp quản lý các tuyến du lịch một cách khoa học, hạn chế sai sót trong khâu quản lý thủ công như: trùng lịch trình, cập nhật giá tour chậm, khó kiểm soát số lượng khách, dễ thất lạc thông tin như booking...

Tuy đã có nhiều ứng dụng quản lý du lịch, nhưng đa số chưa tối ưu cho các doanh nghiệp nhỏ hoặc cho mục đích học tập, nghiên cứu. Vì vậy, việc xây dựng một ứng dụng đơn giản, dễ sử dụng, hỗ trợ quản lý tuyến du lịch là rất cần thiết.

1.2 Mục tiêu nghiên cứu

- Xây dựng được ứng dụng quản lý hoàn chỉnh các tuyến du lịch: thêm, sửa, xóa, tìm kiếm.
 - Quản lý được thông tin lịch trình, giá tour, thời gian diễn ra tour.
 - Hỗ trợ quản lý khách hàng và đặt tour.
 - Giao diện thân thiện, dễ sử dụng, hoạt động ổn định.
- Mục tiêu được thiết kế cụ thể, có tính khả thi, không quá rộng.

2. Tổng quan và cơ sở lý thuyết

2.1 Đặt vấn đề

Trong thực tế, việc quản lý tuyến du lịch thủ công thường gặp khó khăn như dữ liệu phân tán, dễ sai sót, khó thống kê. Đề tài đặt ra yêu cầu xây dựng một phần mềm giúp tự động hóa các nghiệp vụ quản lý tour.

2.2 Lịch sử giải quyết vấn đề

Hiện nay nhiều công ty sử dụng các hệ thống lớn như TravelERP, Amadeus... nhưng chúng khá phức tạp, chi phí cao, không phù hợp với mô hình vừa và nhỏ hoặc bài tập môn học. Các bài toán nhỏ trong trường học thường chỉ dừng ở mức quản lý thông tin đơn giản, chưa tích hợp đầy đủ chức năng về lịch trình, giá tour và quản lý khách đặt tour. Vì vậy, đề tài mang tính mới trong phạm vi học thuật, hướng đến việc xây dựng hệ thống nhỏ nhưng đầy đủ tính năng.

2.3 Phạm vi của đề tài

Phạm vi nội dung: Quản lý tuyến du lịch, lịch trình, giá tour, khách hàng, đơn đặt tour.

Phạm vi người dùng: Quản trị viên hoặc nhân viên điều hành tour.

Giới hạn đề tài: Không tập trung vào thanh toán trực tuyến hoặc bản đồ định vị.

2.4 Phương pháp nghiên cứu/hướng giải quyết vấn đề:

- Nghiên cứu lý thuyết về lập trình Python, xử lý file, cấu trúc dữ liệu (list, dict,...).
- Áp dụng mô hình CRUD (Create – Read – Update – Delete) để quản lý tuyến du lịch.
- Thiết kế chương trình theo dạng module hoặc hướng đối tượng (OOP) để dễ mở rộng.
- Thử nghiệm chương trình với nhiều bộ dữ liệu khác nhau để đảm bảo tính ổn định.

2.5 Trình bày tóm tắt cơ sở lý thuyết:

- Kiến thức về Python cơ bản: biến, kiểu dữ liệu, hàm, vòng lặp,...
- Kiến thức về Python nâng cao hơn: lập trình hướng đối tượng, xử lý file (txt, json, csv).
- Kiến thức về quản lý dữ liệu dạng danh sách (list) và từ điển (dict).
- Các nguyên tắc thiết kế phần mềm đơn giản: dễ dùng, dễ mở rộng, dễ bảo trì.

3. Kết quả đạt được và demo

A. THIẾT KẾ GIAO DIỆN QUẢN LÝ CÁC TUYẾN DU LỊCH

Quản Lý Tuyển Du Lịch

QUẢN LÝ CÁC TUYỂN DU LỊCH

Thông tin tuyển du lịch

Mã tuyển:	Tên tuyển:
Điểm đi:	Điểm đến:
Thời gian:	Giá tour:

Danh sách các tuyển du lịch:

Matuyen	Tentuyen	Diemdi	Diemden	Thoigian	Giatour
1	An Giang - Đà Lạt	An Giang	Đà Lạt	3 ngày 2 đêm	2500000
2	Hà Nội - Sapa	Hà Nội	Sapa	4 ngày 3 đêm	3500000
3	TP.HCM - Phú Quốc	TP.HCM	Phú Quốc	3 ngày 2 đêm	2800000
4	Cần Thơ - Đà Nẵng	Cần Thơ	Đà Nẵng	5 ngày 4 đêm	4200000
5	Hà Nội - Hạ Long	Hà Nội	Hạ Long	2 ngày 1 đêm	1800000

[Thêm] [Sửa] [Xóa] [Lưu] [Hủy] [Thoát]

B. CÁC BƯỚC XÂY DỰNG ỨNG DỤNG

Để xây dựng ứng dụng quản lý các tuyến du lịch bằng Python, Tkinter và MySQL, cần cài đặt các thư viện cần thiết, thiết kế cơ sở dữ liệu MySQL để lưu trữ thông tin sản phẩm, sử dụng Tkinter để tạo giao diện người dùng, và kết hợp Python với thư viện mysql.connector để thực hiện thao tác như thêm, xóa, sửa, và hiển thị danh sách các tuyến du lịch.

1. Cài đặt các thư viện cần thiết:

- Python: Đảm bảo đã cài đặt Python trên máy của mình
- MySQL: Cài đặt máy chủ MySQL và tạo một cơ sở dữ liệu cho ứng dụng của bạn.
- mysql-connector-python: Cài đặt thư viện này để kết nối Python với cơ sở dữ liệu MySQL

2. Thiết kế dữ liệu MySQL workbench

- Tạo một bảng cơ sở dữ liệu MySQL để lưu trữ thông tin các tuyến du lịch như sau:

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree view has 'qldulich' selected. The main pane displays the following SQL code:

```
1
2 • CREATE DATABASE IF NOT EXISTS quanly_tuyen_du_lich;
3 • USE quanly_tuyen_du_lich;
4
5 • CREATE TABLE IF NOT EXISTS tuyen_du_lich (
6     matuyen INT AUTO_INCREMENT PRIMARY KEY,
7     tentuyen VARCHAR(100),
8     diemdi VARCHAR(200),
9     diemden VARCHAR(200),
10    thoigian DATE,
11    gia_tour DECIMAL(15,2)
12 );
13
14
```

3.Xây dựng giao diện người dùng (GUI) bằng Tkinter:

-Sử dụng thư viện Tkinter để tạo các cửa sổ, trường nhập liệu, nút bấm và bảng hiển thị thông tin Nhân viên.

-Thiết kế giao diện cho phép người dùng thực hiện các chức năng sau:

- + Thêm thông tin các tuyến du lịch mới.
- + Lưu thông tin các tuyến du lịch.
- + Sửa thông tin các tuyến du lịch.
- + Hủy thông tin các tuyến du lịch.
- + Xóa thông tin các tuyến du lịch.
- + Xem danh sách tất cả các tuyến du lịch..
- + Tìm kiếm các tuyến du lịch..

4. Viết mã Python kết nối cơ sở dữ liệu:

- Sử dụng mysql.connector để thiết lập kết nối với cơ sở dữ liệu MySQL.

- Viết các hàm:

+ Thêm tuyến du lịch: Lấy dữ liệu từ các trường nhập liệu trên giao diện và chèn vào bảng.

+ Lưu thông tin tuyến du lịch

+ Sửa thông tin tuyến du lịch: Sửa đổi thông tin du lịch trong cơ sở dữ liệu dựa trên matuyen

+ Hủy thông tin tuyến du lịch.

+ Xóa thông tin tuyến du lịch: Xóa tuyến du lịch khỏi cơ sở dữ liệu dựa trên matuyen

+ Xem danh sách tất cả các tuyến du lịch: Truy vấn cơ sở dữ liệu để lấy tất cả tuyến du lịch và hiển thị chúng trong bảng trên giao diện.

+ Tìm kiếm tuyến du lịch.

5. Xây dựng giao diện Tkinter

- Liên kết các nút bấm trên giao diện Tkinter với các hàm Python tương ứng để xử lý dữ liệu và tương tác với cơ sở dữ liệu.

- Các Chức Năng:

+ Thêm tuyến du lịch : Khi người dùng nhập tên tuyến, điểm đi, điểm đến và nhấn nút "Thêm", mã Python sẽ lấy các giá trị này và thêm một bản ghi mới vào bảng tuyendulich trong MySQL.

+ Xem tuyến du lịch: Khi người dùng mở cửa sổ ứng dụng, mã Python sẽ truy vấn cơ sở dữ liệu, lấy tất thông tin tuyến du lịch hiển thị chúng dưới dạng một danh sách hoặc bảng trên giao diện Tkinter.

C. HƯỚNG DẪN CÀI ĐẶT VÀ CODE ỨNG DỤNG

- Ứng dụng sẽ có:

- + Giao diện nhập thông tin nhân viên (Mã tuyển, Tên tuyển, Điểm đi, Điểm đến, Thời gian, Giá tour).
- + Bảng danh sách các tuyến du lịch.
- + Chức năng CRUD (Thêm, Sửa, Xóa, Lưu).
- + Dữ liệu lưu trực tiếp vào MySQL Database (không mất khi tắt ứng dụng).
- + Giao diện Tkinter thiết kế chuẩn như hình minh họa ở trên.

1. Cài đặt môi trường

- Trước tiên cài đặt thư viện cần thiết:

```
pip install mysql-connector-python tkcalendar
```

- Giải thích:

- + mysql-connector-python: dùng để kết nối Python với MySQL.
- + tkcalendar: để chọn ngày sinh bằng DateEntry

2. Chuẩn bị CSDL MySQL

- Mở MySQL (WorkBench, phpMyAdmin hoặc terminal) rồi tạo database và bảng :

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree view shows two databases: 'qldulich' and 'sys'. The 'qldulich' database is expanded, showing 'Tables', 'Views', 'Stored Procedures', and 'Functions'. The central area displays the SQL code for creating a database and a table:

```
1
2 • CREATE DATABASE IF NOT EXISTS quanly_tuyen_du_lich;
3 • USE quanly_tuyen_du_lich;
4
5 • CREATE TABLE IF NOT EXISTS tuyen_du_lich (
6     matuyen INT AUTO_INCREMENT PRIMARY KEY,
7     tentuyen VARCHAR(100),
8     diemdi VARCHAR(200),
9     diemden VARCHAR(200),
10    thoigian DATE,
11    gia_tour DECIMAL(15,2)
12 );
13
14
```

The 'Administration' tab is selected at the bottom left, and the 'Information' tab is visible at the bottom right.

3. Viết code ứng dụng Tkinter + MySQL



```
 1 import tkinter as tk
 2 from tkinter import ttk, messagebox
 3 import mysql.connector
 4
 5
 6 def connect_db(): 4 usages
 7     return mysql.connector.connect(
 8         host="localhost",
 9         user="root",
10         password="123456",
11         database="qlnhhanvien"
12     )
13
14
15 def center_window(win, w=700, h=500): 1 usage
16     ws = win.winfo_screenwidth()
17     hs = win.winfo_screenheight()
18     x = (ws // 2) - (w // 2)
19     y = (hs // 2) - (h // 2)
20     win.geometry(f'{w}x{h}+{x}+{y}')
```

```

22
23     def clear_input(): 4 usages
24         entry_matuyen.delete( first: 0, tk.END)
25         entry_tentuyen.delete( first: 0, tk.END)
26         entry_diemdi.delete( first: 0, tk.END)
27         entry_thoigian.delete( first: 0, tk.END)
28         entry_giatien.delete( first: 0, tk.END)
29         cbb_phuongtien.set("")
30
31
32     def load_data(): 4 usages
33         for i in tree.get_children():
34             tree.delete(i)
35     >
36     try: ...
37     except Exception as e:
38         messagebox.showerror( title: "Lỗi CSDL", message: f"Không thể tải dữ liệu: {str(e)}\nVui lòng kiểm tra kết nối MySQL.")
39
40
41
42     def them_tuyendulich(): 1 usage
43         matuyen = entry_matuyen.get()
44         tentuyen = entry_tentuyen.get()
45         diemdi = entry_diemdi.get()
46         thoigian = entry_thoigian.get()
47         giatien = entry_giatien.get()
48
49         giatien = entry_giatien.get()
50         phuongtien = cbb_phuongtien.get()
51
52         if not all([matuyen, tentuyen, diemdi, thoigian, giatien, phuongtien]):
53             messagebox.showwarning( title: "Thiếu dữ liệu", message: "Vui lòng nhập đủ thông tin Tuyến Du lịch.")
54             return
55
56         try:
57             conn = connect_db()
58             cur = conn.cursor()
59             sql = "INSERT INTO tuyendulich VALUES (%s, %s, %s, %s, %s, %s)"
60             data = (matuyen, tentuyen, diemdi, thoigian, giatien, phuongtien)
61             cur.execute(sql, data)
62             conn.commit()
63             load_data()
64             clear_input()
65         except Exception as e:
66             messagebox.showerror( title: "Lỗi Thêm", str(e))
67         finally:
68             conn.close()
69
70
71
72     def xoa_tuyendulich(): 1 usage
73         selected = tree.selection()
74         if not selected:

```

```

75     if not selected:
76         messagebox.showwarning( title: "Chưa chọn", message: "Hãy chọn tuyến du lịch để xóa.")
77         return
78
79     matuyen = tree.item(selected)["values"][0]
80     try:
81         conn = connect_db()
82         cur = conn.cursor()
83         cur.execute("DELETE FROM tuyendulich WHERE matuyen=%s", (matuyen,))
84         conn.commit()
85         load_data()
86     except Exception as e:
87         messagebox.showerror( title: "Lỗi Xóa", str(e))
88     finally:
89         conn.close()
90
91
92     def sua_tuyendulich(): 1 usage
93         selected = tree.selection()
94         if not selected:
95             messagebox.showwarning( title: "Chưa chọn", message: "Hãy chọn tuyến du lịch để sửa.")
96             return
97
98         values = tree.item(selected)["values"]
99
100        clear_input()
101        entry_matuyen.insert( index: 0, values[0])
102        entry_tentuyen.insert( index: 0, values[1])
103        entry_diemdi.insert( index: 0, values[2])
104        entry_thoigian.insert( index: 0, values[3])
105        entry_giatien.insert( index: 0, values[4])
106        cbb_phuongtien.set(values[5])
107
108
109    def luu_tuyendulich(): 1 usage
110        matuyen = entry_matuyen.get()
111        tentuyen = entry_tentuyen.get()
112        diemdi = entry_diemdi.get()
113        thoigian = entry_thoigian.get()
114        giatien = entry_giatien.get()
115        phuongtien = cbb_phuongtien.get()
116
117        if not all([matuyen, tentuyen, diemdi, thoigian, giatien, phuongtien]):
118            messagebox.showwarning( title: "Thiếu dữ liệu", message: "Vui lòng nhập đủ thông tin Tuyến Du lịch.")
119            return

```

```

1.21     try:
1.22         conn = connect_db()
1.23         cur = conn.cursor()
1.24         sql = """UPDATE tuyendulich SET tentuyen=%s, diemdi=%s, thoigian=%s, giatien=%s, phuongtien=%s
1.25             WHERE matuyen=%s"""
1.26         data = (tentuyen, diemdi, thoigian, giatien, phuongtien, matuyen)
1.27         cur.execute(sql, data)
1.28         conn.commit()
1.29         load_data()
1.30         clear_input()
1.31     except Exception as e:
1.32         messagebox.showerror( title: "Lỗi Lưu", str(e))
1.33     finally:
1.34         conn.close()

1.35
1.36
1.37 root = tk.Tk()
1.38 root.title("Quản lý Tuyến Du lịch")
1.39 center_window(root, w: 700, h: 500)
1.40 root.resizable( width: False, height: False)
1.41
1.42 lbl_title = tk.Label(root, text="QUẢN LÝ TUYẾN DU LỊCH", font=("Arial", 18, "bold"))
1.43 lbl_title.pack(pady=10)

1.44
1.45 frame_info = tk.Frame(root)
1.46 frame_info.pack(pady=5, padx=10, fill="x")
1.47
1.48 tk.Label(frame_info, text="Mã tuyến").grid(row=0, column=0, padx=5, pady=5, sticky="w")
1.49 entry_matuyen = tk.Entry(frame_info, width=10)
1.50 entry_matuyen.grid(row=0, column=1, padx=5, pady=5, sticky="w")
1.51 tk.Label(frame_info, text="Phương tiện").grid(row=0, column=2, padx=5, pady=5, sticky="w")
1.52 cbb_phuongtien = ttk.Combobox(frame_info, values=["Máy bay", "Tàu hỏa", "Ô tô", "Xe khách"], width=20)
1.53 cbb_phuongtien.grid(row=0, column=3, padx=5, pady=5, sticky="w")
1.54
1.55 tk.Label(frame_info, text="Tên tuyến").grid(row=1, column=0, padx=5, pady=5, sticky="w")
1.56 entry_tentuyen = tk.Entry(frame_info, width=25)
1.57 entry_tentuyen.grid(row=1, column=1, padx=5, pady=5, sticky="w")
1.58 tk.Label(frame_info, text="Điểm đi").grid(row=1, column=2, padx=5, pady=5, sticky="w")
1.59 entry_diemdi = tk.Entry(frame_info, width=15)
1.60 entry_diemdi.grid(row=1, column=3, padx=5, pady=5, sticky="w")
1.61
1.62 tk.Label(frame_info, text="Thời gian (ngày)").grid(row=2, column=0, padx=5, pady=5, sticky="w")
1.63 entry_thoigian = tk.Entry(frame_info, width=10)
1.64 entry_thoigian.grid(row=2, column=1, padx=5, pady=5, sticky="w")
1.65 tk.Label(frame_info, text="Giá tiền (VNĐ)").grid(row=2, column=2, padx=5, pady=5, sticky="w")
1.66 entry_giatien = tk.Entry(frame_info, width=15)
1.67 entry_giatien.grid(row=2, column=3, padx=5, pady=5, sticky="w")

```

```

56
57
58
59     lbl_ds = tk.Label(root, text="Danh sách tuyến du lịch", font=("Arial", 10, "bold"))
60     lbl_ds.pack(pady=5, anchor="w", padx=10)
61
62     columns = ("matuyen", "tentuyen", "diemdi", "thoigian", "gatiens", "phuongtien")
63     tree = ttk.Treeview(root, columns=columns, show="headings", height=10)
64
65     tree.column("matuyen", width=60, anchor="center")
66     tree.heading("matuyen", text="Mã Tuyến")
67     tree.column("tentuyen", width=150)
68     tree.heading("tentuyen", text="Tên Tuyến")
69     tree.column("diemdi", width=100)
70     tree.heading("diemdi", text="Điểm Đi")
71     tree.column("thoigian", width=70, anchor="center")
72     tree.heading("thoigian", text="Thời Gian")
73     tree.column("gatiens", width=100, anchor="center")
74     tree.heading("gatiens", text="Giá Tiết")
75     tree.column("phuongtien", width=150)
76     tree.heading("phuongtien", text="Phương Tiện")
77
78
79
80
81
82
83
84
85
86
87
88     tree.pack(padx=10, pady=5, fill="both")
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188     tree.pack(padx=10, pady=5, fill="both")
189
190     frame_btn = tk.Frame(root)
191     frame_btn.pack(pady=5)
192
193     tk.Button(frame_btn, text="Thêm", width=8, command=them_tuyendulich).grid(row=0, column=0, padx=5)
194     tk.Button(frame_btn, text="Lưu", width=8, command=luu_tuyendulich).grid(row=0, column=1, padx=5)
195     tk.Button(frame_btn, text="Sửa", width=8, command=sua_tuyendulich).grid(row=0, column=2, padx=5)
196     tk.Button(frame_btn, text="Hủy", width=8, command=clear_input).grid(row=0, column=3, padx=5)
197     tk.Button(frame_btn, text="Xóa", width=8, command=xoa_tuyendulich).grid(row=0, column=4, padx=5)
198     tk.Button(frame_btn, text="Thoát", width=8, command=root.quit).grid(row=0, column=5, padx=5)
199
200
201     load_data()
202     root.mainloop()

```