**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**🕯✡🕮🕮✡🕯**

**Logo

Description automatically generated**

**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI**

**QUẢN LÝ ĐIỂM THI CỦA CÁC TRƯỜNG PHỔ THÔNG DÙNG GIAO DIỆN TKINTER PYTHON**

**GVHD: Trần Tiến Đức**

**Nhóm thực hiện: nhóm 2**

**Thành viên nhóm**

**Bùi Lê Quốc Hưng 19110220**

**Đỗ Minh Quân 19119122**

**Trang lót**

Mục lục

[I. Đặc tả 5](#_Toc89630943)

[II. Phân công công việc 6](#_Toc89630944)

[Table 1 - Phân công công việc 6](#_Toc89630945)

[III. Thiết kế: 7](#_Toc89630946)

[1. Thuật toán: 7](#_Toc89630947)

[2. Thiết kế modules: 7](#_Toc89630948)

[Table 2 - Chi tiết modules 7](#_Toc89630949)

[3. Thiết kế lớp: 8](#_Toc89630950)

[Table 3 - Chi tiết các lớp 8](#_Toc89630951)

[Table 4 - Chi tiết các phương thức 9](#_Toc89630952)

[4. Thiết kế cơ sở dữ liệu 41](#_Toc89630953)

[Figure 1 - Cơ sở dữ liệu 41](#_Toc89630954)

[Table 5 - Chi tiết bảng 42](#_Toc89630955)

[5. Mô tả Field trong Table 42](#_Toc89630956)

[Table 6 - Chi tiết trường của bảng DanhSach 42](#_Toc89630957)

[Table 7 - Chi tiết trường của bảng Diem 42](#_Toc89630958)

[Table 8 - Chi tiết trường của bảng Truong 43](#_Toc89630959)

[6. Giao diện chương trình 43](#_Toc89630960)

[Table 9 - Các giao diện 43](#_Toc89630961)

[IV. Cài đặt và kiểm thử 47](#_Toc89630962)

[Table 10 - Kiểm thử 47](#_Toc89630963)

[V. Kết luận: 51](#_Toc89630964)

[1. Ưu điểm: 51](#_Toc89630965)

[2. Nhược điểm: 51](#_Toc89630966)

**Danh mục các bảng**

[Table 1 - Phân công công việc 6](#_Toc89630746)

[Table 2 - Chi tiết modules 7](#_Toc89630747)

[Table 3 - Chi tiết các lớp 8](#_Toc89630748)

[Table 4 - Chi tiết các phương thức 9](#_Toc89630749)

[Table 5 - Chi tiết bảng 42](#_Toc89630750)

[Table 6 - Chi tiết trường của bảng DanhSach 42](#_Toc89630751)

[Table 7 - Chi tiết trường của bảng Diem 42](#_Toc89630752)

[Table 8 - Chi tiết trường của bảng Truong 43](#_Toc89630753)

[Table 9 - Các giao diện 43](#_Toc89630754)

[Table 10 - Kiểm thử 47](#_Toc89630755)

**Danh mục các hình**

[Figure 1 - Cơ sở dữ liệu 41](#_Toc89630764)

# Đặc tả

Phần mềm quản lý điểm thi học sinh của các trường phổ thông dùng giao diện tkinter Python.

Dữ liệu đầu vào gồm 3 file: DanhSach.txt, Diem.txt và Truong.txt

Trong mỗi file đã có sẵn data cần thiết để làm đồ án.

Mục đích của tính năng bao gồm:

1. Có chức năng mở 3 file trên và xem mỗi file trong tkinter.Text hay tkinter.Listbox
2. Cho phép xóa một mẫu tin.
3. Cho phép chèn một mẫu tin mới vào trước một mẫu tin.
4. Cho phép chèn một mẫu tin mới vào sau một mẫu tin.
5. Cho phép xem trong tkinter.Text hay tkinter.Listbox gồm các field:

SBD, Ho, Ten, Phai, NgaySinh, Toan, Van, TongDiem, DTN, XepLoai, MaTruong

Trong đó:

DTN: Là điểm thấp nhất của 2 môn Toan, Van.

TongDiem: Toan + Van

XepLoai:

Nếu TongDiem >= 16 và không có môn nào dưới 7: Giỏi

Nếu TongDiem >= 14 và không có môn nào dưới 6: Khá

Nếu TongDiem >= 10 và không có môn nào dưới 4: Trung Bình

Ngược lại: Kém

Kết quả sắp tăng dần theo MaTruong, cùng MaTruong sắp tăng dần theo Ten, cùng Ten sắp tăng dần theo Ho

1. Cho phép xem trong tkinter.Text hay tkinter.Listbox Học sinh có điểm Toán lớn hơn điểm Văn với các field:

SBD, Ho, Ten, Phai, NTNS, Toan, Van, MaTruong

1. Cho phép xem trong tkinter.Text hay tkinter.Listbox Học sinh Kém với các field:

SBD, Ho, Ten, Phai, NgaySinh, Toan, Van, TongDiem, XepLoai, MaTruong

Kết quả sắp tăng dần theo MaTruong, cùng MaTruong sắp giảm dần theo tổng điểm.

1. Cho phép xem trong tkinter.Text hay tkinter.Listbox Danh sách Học Sinh thủ khoa của từng trường có tổng điểm cao nhất với các field:

SBD, Ho, Ten, Phai, NgaySinh, Toan, Van, TongDiem, XepLoai, MaTruong

1. Cho phép nhập vàoMaTruong và xem trong tkinter.Text hay tkinter.Listbox chỉ gồm các học sinh của trường đó với các field:

SBD, Ho, Ten, Phai, NgaySinh, Toan, Van, TongDiem, XepLoai, MaTruong.

Cuối danh sách là dòng thống kê: Số HS trong trường, Đậu, Trượt, Tỷ lệ phần trăm

1. Cho phép xem trong tkinter.Text hay tkinter.Listbox kết quả sau:

GIAY BAO DIEM GIAY BAO DIEM GIAY BAO DIEM

So Bao Danh: 1 So Bao Danh: 2 SoBaoDanh: 3

Anh: Nguyen Viet Hong Chi: Nguyen Hong Hanh Chi: Dinh Thi Tam

Toan: 7 Toan: 6 Toan: 5

Van: 7.5 Van: 6.5 Van: 5.5

Tong Diem: 14.5 Tong Diem: 12.5 Tong Diem: 10.5

Xep Loai: Kha Xep Loai: Trung Binh Xep Loai: Trung Binh

…

Ghi chú: Từ yêu cầu 5 đến yêu cầu 10 đều phải ghi ra file .txt và xem file .txt trong tkinter.Text hay tkinter.Listbox.

# Phân công công việc

### Table 1 - Phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **Nhiệm vụ chi tiết** | **Mức độ đóng góp** |
| 1 | Bùi Lê Quốc Hưng | Xứ lý code phần backend dùng csdl Mysql | 100% |
| 2 | Đỗ Minh Quân | Xử lý code phần backendtxt không dùng csdl Mysql | 100% |
| 3 | Bùi Lê Quốc Hưng | Xứ lý code phần frontend giao diện Tkinter | 50% |
| 4 | Đỗ Minh Quân | Xứ lý code phần frontend giao diện Tkinter | 50% |
|  | **Total** |  | **100%** |

# Thiết kế:

## Thuật toán:

* 1. Tìm kiếm phần tử trong mảng:

Sử dụng vòng lặp for để lặp qua từng phần tử của mảng, sau đó so sánh nếu phần tử cần tìm xuất hiện trong mảng thì in giá trị và vị trí của nó ra.

* 1. Sắp xếp:

Các thuật toán sắp xếp thường được các nhà phát triển dùng để đặt dữ liệu theo cách có tổ chức.

Trong đồ án này sử dụng chức năng sắp xếp được dựng sẵn của ngôn ngữ Python – sorted()

* 1. Phân tích liên kết:

Liên kết các mẫu tin trong 3 files data được cho sẵn trong qua các trường như MaHS, MaTruong

## Thiết kế modules:

### Table 2 - Chi tiết modules

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên module** | **Mục đích** | **Họ và tên** |
| 1 | frontend | Nơi chứa code liên quan đến giao diện tkinter | Bùi Lê Quốc Hưng  Đỗ Minh Quân |
| 2 | backend | Nơi chứa code liên quan đến xử lý lưu trữ dữ liệu vào MYSQL database | Bùi Lê Quốc Hưng |
| 3 | backendtxt | Nơi chứa code lien quan đến xử lý lưu trữ dữ liệu không dung CSDL | Đỗ Minh Quân |

## Thiết kế lớp:

* 1. Các lớp được sử dụng

### Table 3 - Chi tiết các lớp

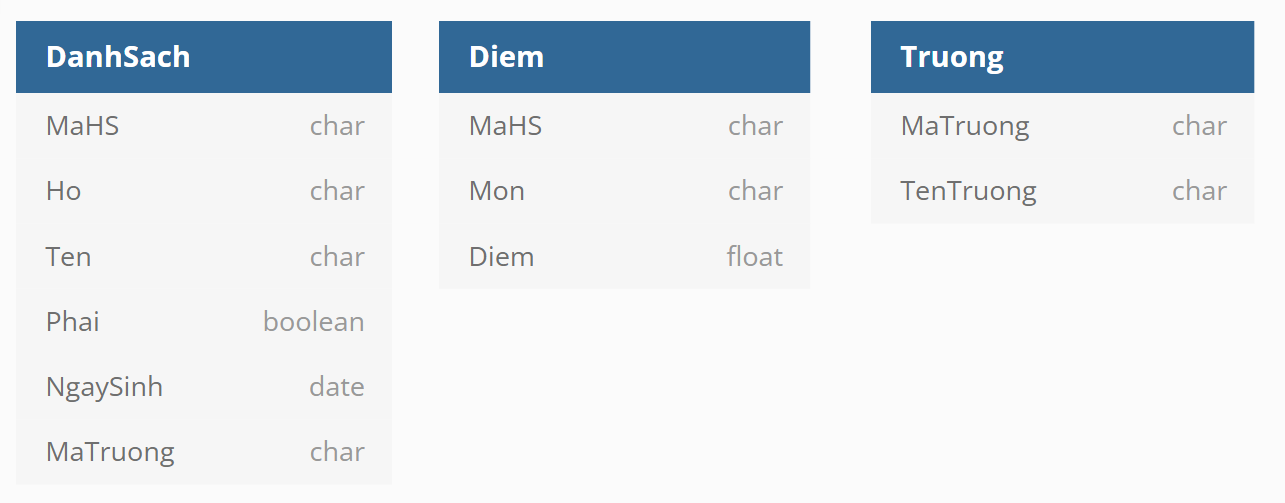
|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên lớp** | **Mục đích** | **Họ và tên** |
| 1 | TK()  dẫn xuất từ tkinter | Hỗ trợ cho việc lập trình GUI. | Bùi Lê Quốc Hưng |
| 2 | Listbox()  dẫn xuất từ tkinter | Một tiện ích giúp hiển thị danh sách bản ghi và có thể chọn bản ghi. | Đỗ Minh Quân |
| 3 | Button()  dẫn xuất từ tkinter | Tiện ích được sử dụng để thêm các nút trong ứng dụng Python. Các nút này có thể hiển thị văn bản hoặc hình ảnh truyền đạt mục đích của các nút. Bạn có thể đính kèm một chức năng hoặc một phương thức vào một nút được gọi tự động khi bạn nhấp vào nút. | Đỗ Minh Quân |
| 4 | Entry()  dẫn xuất từ tkinter | Tiện ích Entry được sử dụng để chấp nhận các chuỗi văn bản một dòng từ người dùng. | Bùi Lê Quốc Hưng |
| 5 |  |  |  |

* 1. Các phương thức được sử dụng

### Table 4 - Chi tiết các phương thức

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Phương thức** | **Mục đích** | **Tên file, số thứ tự dòng chứa khai báo** |
| 1 | viewds\_command()  Input: Không có  Output: Không có  Pseudo code:  lb1.delete(0,END)  for row in backend.view\_ds():  lb1.insert(END,newRow) | Chèn và hiển thị danh sách học sinh vào listbox | frontend.py (36) |
| 2 | viewdiem\_command()  Input: Không có  Output: Không có  Pseudo code:  lb1.delete(0,END)  for row in backend.view\_ds():  lb1.insert(END,newRow) | Chèn và hiển thị danh sách điểm vào listbox | frontend.py (59) |
| 3 | viewtruong\_command()  Input: Không có  Output: Không có  Pseudo code:  lb1.delete(0,END)  for row in backend.view\_ds():  lb1.insert(END,newRow) | Chèn và hiển thị danh sách trường vào listbox | frontend.py (70) |
| 4 | delete\_command()  Input: Không có  Output: Không có  Pseudo code:  backend.delete(selected\_tuple[0], selected\_view) | Gọi phương thức xóa ở backend | frontend.py (174) |
| 5 | add\_before\_command()  Input: Không có  Output: Không có  Pseudo code:  if selected\_view == 1:  backend.insert\_ds(…)  elif selected\_view == 2:  backend.insert\_diem(…)  elif selected\_view == 3:  backend.insert\_truong(…) | Gọi phương thức chèn dữ liệu ở backend | frontend.py (143) |
|  | add\_after\_command()  Input: Không có  Output: Không có  Pseudo code:  if selected\_view == 1:  backend.insert\_ds(…)  elif selected\_view == 2:  backend.insert\_diem(…)  elif selected\_view == 3:  backend.insert\_truong(…) | Gọi phương thức chèn dữ liệu ở backend | frontend.py (157) |
|  | view\_item5\_command()  Input: Không có  Output: Không có  Pseudo code:  # viet ra file: "item5.txt"  output\_file = backend.view\_item5()  # doc noi dung file  list\_data = backend.read\_file\_data(output\_file)  # khai báo Tinker Window listbox mới  item5Window = Tk()  item5Window.title("Yêu cầu số 5 - Danh sach từng HS và điểm")  item5Window.geometry('1200x400')  # loop và display từng record trong file txt  for i in list\_data:  listbox.insert(END, item) | Gọi phương thức đọc data cho yêu cầu số 5 từ backend.  Sau đó hiển thị data lên tkinter listbox từ file txt. | frontend.py (233) |
|  | view\_item6\_command()  Input: Không có  Output: Không có  Pseudo code:  # viet ra file: "item6.txt"  output\_file = backend.view\_item6()  # doc noi dung file  list\_data = backend.read\_file\_data(output\_file)  # khai báo Tinker Window listbox mới  Item6Window = Tk()  Item6Window.title("Yêu cầu số 6 - Học sinh có điểm Toán lớn hơn điểm Văn")  Item6Window.geometry('1200x400')  # loop và display từng record trong file txt  for i in list\_data:  listbox.insert(END, item) | Gọi phương thức đọc data cho yêu cầu số 6 từ backend.  Sau đó hiển thị data lên tkinter listbox từ file txt. | frontend.py (269) |
|  | view\_item7\_command()  Input: Không có  Output: Không có  Pseudo code:  # viet ra file: "item7.txt"  output\_file = backend.view\_item7()  # doc noi dung file  list\_data = backend.read\_file\_data(output\_file)  # khai báo Tinker Window listbox mới  Item7Window = Tk()  Item7Window.title("Yêu cầu số 7 - Học sinh Kém")  Item7Window.geometry('1200x400')  # loop và display từng record trong file txt  for i in list\_data:  listbox.insert(END, item) | Gọi phương thức đọc data cho yêu cầu số 7 từ backend.  Sau đó hiển thị data lên tkinter listbox từ file txt. | frontend.py (302) |
|  | view\_item8\_command()  Input: Không có  Output: Không có  Pseudo code:  # viet ra file: "item8.txt"  output\_file = backend.view\_item8()  # doc noi dung file  list\_data = backend.read\_file\_data(output\_file)  # khai báo Tinker Window listbox mới  Item8Window = Tk()  Item8Window.title("Yêu cầu số 8 - Học Sinh thủ khoa của từng trường có tổng điểm cao nhất")  Item8Window.geometry('1200x400')  # loop và display từng record trong file txt  for i in list\_data:  listbox.insert(END, item) | Gọi phương thức đọc data cho yêu cầu số 8 từ backend.  Sau đó hiển thị data lên tkinter listbox từ file txt. | frontend.py (337) |
|  | view\_item9\_command()  Input: Không có  Output: Không có  Pseudo code:  # viet ra file: "item9.txt"  output\_file = backend.view\_item9(e6.get())  # doc noi dung file  list\_data = backend.read\_file\_data(output\_file)  # khai báo Tinker Window listbox mới  Item9Window = Tk()  Item9Window.title("Yêu cầu số 9 - Chỉ gồm các học sinh của trường đó")  Item9Window.geometry('1200x400')  # loop và display từng record trong file txt  for i in list\_data:  listbox.insert(END, item) | Lấy mã trường từ Entry e6 (Mã Trường) và gọi phương thức đọc data cho yêu cầu số 9 từ backend.  Sau đó hiển thị data lên tkinter listbox từ file txt. | frontend.py (372) |
|  | view\_item10\_command()  Input: Không có  Output: Không có  Pseudo code:  # viet ra file: "item10.txt"  output\_file = backend.view\_item9()  # doc noi dung file  list\_data = backend.read\_file\_data(output\_file)  header = "GIAY BAO DIEM"  # khai báo Tinker Window listbox mới  Item10Window = Tk()  Item10Window.title("Yêu cầu số 10 - GIAY BAO DIEM")  Item10Window.geometry('1200x400')  # loop và display từng record trong file txt  # listbox thứ 1  for i in list\_data:  listbox10\_1.insert(END, item)  # listbox thứ 2  for i in list\_data:  listbox10\_2.insert(END, item)  # listbox thứ 3  for i in list\_data:  listbox10\_3.insert(END, item) | Gọi phương thức đọc data cho yêu cầu số 10 từ backend.  Sau đó hiển thị data lên tkinter listbox từ file txt. | frontend.py (407) |
|  | update\_item10\_data\_command ()  Input: Không có  Output: Không có  Pseudo code: Tương tự view\_item10\_command() | Gọi phương thức đọc data cho yêu cầu số 10 từ backend.  Sau đó cập nhật danh sách hiển thị data lên tkinter listbox từ file txt. | frontend.py (468) |
|  | view\_ds()  Input: Không có  Output: List dữ liệu danh sách học sinh truy xuất từ CSDL  Pseudo code:  conn = mysql.connector.connect(host = "localhost",user = DB\_USER, password =DB\_PASSWORD,database="students")  cur = conn.cursor()  cur.execute("SELECT \* FROM danhsach")  rows = cur.fetchall()  conn.close()  return rows | Truy vấn dữ liệu từ bảng danhsach trong CSDL. | backend.py (41) |
|  | view\_diem()  Input: Không có  Output: List dữ liệu điểm truy xuất từ CSDL  Pseudo code:  conn = mysql.connector.connect(host = "localhost",user = DB\_USER, password =DB\_PASSWORD,database="students")  cur = conn.cursor()  cur.execute("SELECT \* FROM diem")  rows = cur.fetchall()  conn.close()  return rows | Truy vấn dữ liệu từ bảng diem trong CSDL. | backend.py (54) |
|  | view\_truong()  Input: Không có  Output: List dữ liệu trường truy xuất từ CSDL  Pseudo code:  conn = mysql.connector.connect(host = "localhost",user = DB\_USER, password =DB\_PASSWORD,database="students")  cur = conn.cursor()  cur.execute("SELECT \* FROM truong")  rows = cur.fetchall()  conn.close()  return rows | Truy vấn dữ liệu từ bảng truong trong CSDL. | backend.py (66) |
|  | delete(id, table, option\_val="")  Input: id - Mã xác định để xóa; table - bảng cần truy vấn để xóa; option\_val - data them khi cần xác định ngoài Id.  Output: Không có  Pseudo code:  conn = mysql.connector.connect(host = "localhost", user = DB\_USER, password = DB\_PASSWORD, database="Students")  cur = conn.cursor()  if table == 1:  cur.execute("DELETE FROM DanhSach WHERE MaHS='%s'" % (id.strip(),))  elif table == 2:  cur.execute("DELETE FROM Diem WHERE MaHS='%s' AND Mon='%s'" % (id.strip(),option\_val.strip(),))  elif table == 3:  cur.execute("DELETE FROM Truong WHERE MaTruong='%s'" % (id.strip(),))  conn.commit()  conn.close() | Xóa bản ghi từ CSDL theo id | backend.py (438) |
|  | insert\_ds(index,mahs,ho,ten,phai,ngaysinh,matruong,vitri="truoc")  Input: index – Vị trí cần chèn; mahs,ho,ten,phai,ngaysinh,matruong – dữ liệu bản ghi cần chèn; vị trí – “truoc/sau” để xác định chèn trước hay sau.  Output: Không có  Pseudo code:  startEffectedRow = index  if vitri == "sau":  startEffectedRow += 1  conn = mysql.connector.connect(host = "localhost",  user = DB\_USER, password = DB\_PASSWORD,  database="students")  cur = conn.cursor()  cur.execute("DROP TABLE IF EXISTS tempDanhSach")  cur.execute("CREATE TABLE tempdanhsach AS SELECT \* from DanhSach limit %s,1000" % (startEffectedRow,))    cur.execute("DELETE FROM DanhSach where MaHS in (SELECT MaHS from tempDanhSach)")  cur.execute("INSERT INTO DanhSach (MaHS, Ho, Ten, Phai, NgaySinh, MaTruong) VALUES (%s,%s,%s,%s,%s,%s)", (mahs,ho,ten,phai,ngaysinh,matruong))    cur.execute("INSERT INTO DanhSach (MaHS, Ho, Ten, Phai, NgaySinh, MaTruong) SELECT MaHS, Ho, Ten, Phai, NgaySinh, MaTruong FROM tempDanhSach")    cur.execute("DROP TABLE IF EXISTS tempDanhSach")  conn.commit()  conn.close() | Chèn bản ghi mới vô trước / sau vô một vị trí nhất định trên bảng DanhSach trong CSDL | backend.py (78) |
|  | insert\_diem(index,mahs,mon,diem,vitri="truoc")  Input: index – Vị trí cần chèn; mahs,mon,diem – dữ liệu bản ghi cần chèn; vị trí – “truoc/sau” để xác định chèn trước hay sau.  Output: Không có  Pseudo code: Tương tự insert\_ds() | Chèn bản ghi mới vô trước / sau vô một vị trí nhất định trên bảng Diem trong CSDL | backend.py (107) |
|  | insert\_truong(index,matruong,tentruong,vitri="truoc")  Input: index – Vị trí cần chèn; matruong,tentruong,vitri – dữ liệu bản ghi cần chèn; vị trí – “truoc/sau” để xác định chèn trước hay sau.  Output: Không có  Pseudo code: Tương tự insert\_ds() | Chèn bản ghi mới vô trước / sau vô một vị trí nhất định trên bảng Truong trong CSDL | backend.py (136) |
|  | view\_item5()  Input: Không có  Output: Tên file chứa kết quả item5.txt  Pseudo code:  danh\_sach = execute\_sql("select \* from DanhSach")  diem = execute\_sql("select \* from Diem")  list\_result = []  for ds in danh\_sach:  data = {}  data["SDB"] = ds[0]  data["Ho"] = ds[1]  data["Ten"] = ds[2]  data["Phai"] = ds[3]  data["NgaySinh"] = ds[4]  data["Toan"] = 0  data["Van"] = 0  data["TongDiem"] = 0  data["DTN"] = 0  data["XepLoai"] = None  data["MaTruong"] = ds[5]  for d in diem:  if data["SDB"] == d[0]:  if d[1].strip()=="Toan":  data["Toan"] = d[2]  if d[1].strip()=="Van":  data["Van"] = d[2]  data["TongDiem"] = data["Toan"] + data["Van"]  data["DTN"] = min(data["Toan"],data["Van"])  data["XepLoai"] = get\_xeploai(data["Toan"], data["Van"])  list\_result.append(data)  mylist = sorted(list\_result, key=lambda k: (k['MaTruong'].lower(), k['Ten'].lower(), k['Ho'].lower()))  header = "SBD, Ho, Ten, Phai, NgaySinh, Toan, Van, TongDiem, DTN, XepLoai, MaTruong"  file\_name = "item5.txt"  write\_to\_file(file\_name, mylist, header)  return file\_name | Truy xuất data theo yêu cầu số 5 - Danh sách từng HS và điểm.  Ghi kết quả dữ liệu ra file item5.txt  Sau đó phương thức trả về tên file. | backend.py (165) |
|  | view\_item6()  Input: Không có  Output: Tên file chứa kết quả item6.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 6 - Học sinh có điểm Toán lớn hơn điểm Văn.  Ghi kết quả dữ liệu ra file item6.txt  Sau đó phương thức trả về tên file. | backend.py (203) |
|  | view\_item7()  Input: Không có  Output: Tên file chứa kết quả item7.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 7 - Học sinh Kém.  Ghi kết quả dữ liệu ra file item7.txt  Sau đó phương thức trả về tên file. | backend.py (244) |
|  | view\_item8()  Input: Không có  Output: Tên file chứa kết quả item8.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 8 - Học Sinh thủ khoa của từng trường có tổng điểm cao nhất.  Ghi kết quả dữ liệu ra file item8.txt  Sau đó phương thức trả về tên file. | backend.py (283) |
|  | view\_item9(nameSchool)  Input: nameSchool – Mã trường dùng để tìm kiếm  Output: Tên file chứa kết quả item9.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 9 - Chỉ gồm các học sinh của trường đó.  Ghi kết quả dữ liệu ra file item9.txt  Sau đó phương thức trả về tên file. | backend.py (348) |
|  | view\_item10()  Input: Không có  Output: Tên file chứa kết quả item10.txt  Pseudo code: Tương tự view\_item5(), và chỉ lấy ra random 3 học sinh từ kết quả  num\_to\_select = 3 # set the number to select here.  list\_of\_random\_items = random.sample(danh\_sach, num\_to\_select) | Truy xuất data theo yêu cầu số 10 – GIẤY BÁO ĐIỂM.  Ghi kết quả dữ liệu ra file item10.txt  Sau đó phương thức trả về tên file. | backend.py (387) |
|  | read\_file\_data(file\_name)  Input: file\_name – Tên file cần đọc  Output: Danh sách dữ liệu hang trong file.  Pseudo code:  #Read file and add all line to a list variable  with open(file\_name, 'r')as f:  a = f.readlines()  lst = []  for i in a:  lst.append(i.replace(',0,',',Nam,').replace(',1,',',Nu,').replace("\n",""))  return lst | Đọc nội dung file được dùng trong các yêu cầu 5,6,7,8,9,10 | backend.py (498) |
|  | write\_to\_file (filename, a\_list, header)  Input: file\_name – Tên file cần ghi; a\_list – dữ liệu cần ghi vô file; header – tên header cần ghi vô file  Output: Không có.  Pseudo code:  textfile = open(filename, "w")  textfile.write(header + "\n")  for element in a\_list:  line = ""  for key,value in element.items():  line += str(value) + ","  textfile.write(line + "\n")  textfile.close() | Xuất ra file txt có header. | backend.py (477) |
|  | view\_ds()  Input: Không có  Output: List dữ liệu danh sách học sinh truy xuất từ file trong thư mục datafiles  Pseudo code:  rows = []  with open('./datafiles/DanhDach.csv', 'r') as f:  reader = csv.reader(f)  for i in reader:  rows.append(i)  # print(rows)  return rows | Lấy dữ liệu từ file DanhDach.csv trong thư mục datafilestxt | backendtxt.py (8) |
|  | view\_diem()  Input: Không có  Output: List dữ liệu điểm truy xuất từ file trong thư mục datafiles  Pseudo code:  rows = []  with open('./datafiles/Diem.csv', 'r') as f:  reader = csv.reader(f)  for i in reader:  rows.append(i)  return rows | Lấy dữ liệu từ file Diem.csv trong thư mục datafilestxt | backendtxt.py (19) |
|  | view\_truong ()  Input: Không có  Output: List dữ liệu trường truy xuất từ file trong thư mục datafiles  Pseudo code:  rows = []  with open('./datafiles/Truong.csv', 'r') as f:  reader = csv.reader(f)  for i in reader:  rows.append(i)  return rows | Lấy dữ liệu từ file Truong.csv trong thư mục datafilestxt | backendtxt.py (29) |
|  | delete(id, table, option\_val="")  Input: id - Mã xác định để xóa; table – file cần truy vấn để xóa; option\_val - data them khi cần xác định ngoài Id.  Output: Không có  Pseudo code:  danh\_sach = view\_ds()  danh\_sach\_copy = danh\_sach.copy()  diem = view\_diem()  diem\_copy = diem.copy()  truong = view\_truong()  truong\_copy = truong.copy()  if table == 1:  filePath = "./datafiles/DanhDach.csv"  for i in danh\_sach:  if i[0] == id:  danh\_sach\_copy.remove(i)  write\_file\_csv(filePath,danh\_sach\_copy)  elif table == 2:  filePath = "./datafiles/Diem.csv"  for i in diem:  if i[0] == id and i[1] == option\_val:  diem\_copy.remove(i)  write\_file\_csv(filePath,diem\_copy)  elif table == 3:  filePath = "./datafiles/Truong.csv"  for i in truong:  if i[0] == id:  truong\_copy.remove(i)  write\_file\_csv(filePath,truong\_copy) | Xác định bản ghi theo id, sau đó xóa bản ghi từ file được chỉ định. | backendtxt.py (39) |
|  | insert\_ds(index,mahs,ho,ten,phai,ngaysinh,matruong,vitri="truoc")  Input: index – Vị trí cần chèn; mahs,ho,ten,phai,ngaysinh,matruong – dữ liệu bản ghi cần chèn; vị trí – “truoc/sau” để xác định chèn trước hay sau.  Output: Không có  Pseudo code:  filePath = "./datafiles/DanhDach.csv"  startEffectedRow = index  if vitri == "sau":  startEffectedRow += 1  listDs = view\_ds()  elm = [mahs,ho,ten,phai,ngaysinh,matruong]  diem\_copy = listDs[0:startEffectedRow] + [elm] + listDs[startEffectedRow:]  write\_file\_csv(filePath,diem\_copy) | Chèn bản ghi vô file DachSach vào trước hay sau một vị trí bất kỳ. | backendtxt.py (70) |
|  | insert\_diem(index,mahs,mon,diem,vitri="truoc")  Input: index – Vị trí cần chèn; mahs,mon,diem – dữ liệu bản ghi cần chèn; vị trí – “truoc/sau” để xác định chèn trước hay sau.  Output: Không có  Pseudo code:  filePath = "./datafiles/Diem.csv"  startEffectedRow = index  if vitri == "sau":  startEffectedRow += 1  listDiem = view\_diem()  elm = [mahs,mon,diem]  diem\_copy = listDiem[0:startEffectedRow] + [elm] + listDiem[startEffectedRow:]  write\_file\_csv(filePath,diem\_copy) | Chèn bản ghi vô file Diem vào trước hay sau một vị trí bất kỳ. | backendtxt.py (86) |
|  | insert\_truong(index,matruong,tentruong,vitri="truoc")  Input: index – Vị trí cần chèn; matruong,tentruong,vitri – dữ liệu bản ghi cần chèn; vị trí – “truoc/sau” để xác định chèn trước hay sau.  Output: Không có  Pseudo code:  filePath = "./datafiles/Truong.csv"  startEffectedRow = index  if vitri == "sau":  startEffectedRow += 1  listTruong = view\_truong()  elm = [matruong,tentruong]  truong\_copy = listTruong[0:startEffectedRow] + [elm] + listTruong[startEffectedRow:]  write\_file\_csv(filePath,truong\_copy) | Chèn bản ghi vô file Truong vào trước hay sau một vị trí bất kỳ. | backendtxt.py (101) |
|  | view\_item5()  Input: Không có  Output: Tên file chứa kết quả item5.txt  Pseudo code:  danh\_sach = view\_ds()  diem = view\_diem()  list\_result = []  for ds in danh\_sach:  data = {}  data["SBD"] = ds[0]  data["Ho"] = ds[1]  data["Ten"] = ds[2]  data["Phai"] = ds[3]  data["NgaySinh"] = ds[4]  data["Toan"] = 0  data["Van"] = 0  data["TongDiem"] = 0  data["DTN"] = 0  data["XepLoai"] = None  data["MaTruong"] = ds[5]  for d in diem:  if data["SBD"] == d[0]:  if d[1].strip()=="Toan":  data["Toan"] = float(d[2])  if d[1].strip()=="Van":  data["Van"] = float(d[2])  if data["Van"] != 0 and data["Toan"] != 0:  data["TongDiem"] = data["Toan"] + data["Van"]  data["DTN"] = min(data["Toan"],data["Van"])  data["XepLoai"] = get\_xeploai(data["Toan"], data["Van"])  break  list\_result.append(data)  mylist = sorted(list\_result, key=lambda k: (k['MaTruong'].lower(), k['Ten'].lower(), k['Ho'].lower()))  header = "SBD, Ho, Ten, Phai, NgaySinh, Toan, Van, TongDiem, DTN, XepLoai, MaTruong"  file\_name = "item5.txt"  write\_to\_file(file\_name, mylist, header)  return file\_name | Truy xuất data theo yêu cầu số 5 - Danh sách từng HS và điểm.  Ghi kết quả dữ liệu ra file item5.txt  Sau đó phương thức trả về tên file. | backendtxt.py (116) |
|  | view\_item6()  Input: Không có  Output: Tên file chứa kết quả item6.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 6 - Học sinh có điểm Toán lớn hơn điểm Văn.  Ghi kết quả dữ liệu ra file item6.txt  Sau đó phương thức trả về tên file. | backendtxt.py (156) |
|  | view\_item7()  Input: Không có  Output: Tên file chứa kết quả item7.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 7 - Học sinh Kém.  Ghi kết quả dữ liệu ra file item7.txt  Sau đó phương thức trả về tên file. | backendtxt.py (192) |
|  | view\_item8()  Input: Không có  Output: Tên file chứa kết quả item8.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 8 - Học Sinh thủ khoa của từng trường có tổng điểm cao nhất.  Ghi kết quả dữ liệu ra file item8.txt  Sau đó phương thức trả về tên file. | backendtxt.py (229) |
|  | view\_item9(nameSchool)  Input: nameSchool – Mã trường dùng để tìm kiếm  Output: Tên file chứa kết quả item9.txt  Pseudo code: Tương tự view\_item5() | Truy xuất data theo yêu cầu số 9 - Chỉ gồm các học sinh của trường đó.  Ghi kết quả dữ liệu ra file item9.txt  Sau đó phương thức trả về tên file. | backendtxt.py (283) |
|  | view\_item10()  Input: Không có  Output: Tên file chứa kết quả item10.txt  Pseudo code: Tương tự view\_item5(), và chỉ lấy ra random 3 học sinh từ kết quả  num\_to\_select = 3 # set the number to select here.  list\_of\_random\_items = random.sample(danh\_sach, num\_to\_select) | Truy xuất data theo yêu cầu số 10 – GIẤY BÁO ĐIỂM.  Ghi kết quả dữ liệu ra file item10.txt  Sau đó phương thức trả về tên file. | backendtxt.py (328) |
|  | read\_file\_data(file\_name)  Input: file\_name – Tên file cần đọc  Output: Danh sách dữ liệu hang trong file.  Pseudo code:  #Read file and add all line to a list variable  with open(file\_name, 'r')as f:  a = f.readlines()  lst = []  for i in a:  lst.append(i.replace(',0,',',Nam,').replace(',1,',',Nu,').replace("\n",""))  return lst | Đọc nội dung file được dùng trong các yêu cầu 5,6,7,8,9,10 | backendtxt.py (417) |
|  | write\_to\_file (filename, a\_list, header)  Input: file\_name – Tên file cần ghi; a\_list – dữ liệu cần ghi vô file; header – tên header cần ghi vô file  Output: Không có.  Pseudo code:  textfile = open(filename, "w")  textfile.write(header + "\n")  for element in a\_list:  line = ""  for key,value in element.items():  line += str(value) + ","  textfile.write(line + "\n")  textfile.close() | Xuất ra file txt có header. | backendtxt.py (392) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Thiết kế cơ sở dữ liệu



### Figure 1 - Cơ sở dữ liệu

Mô tả các Table trong CSDL

### Table 5 - Chi tiết bảng

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên bảng** | **Mục đích** |
| 1 | DanhSach | Danh sách học sinh |
| 2 | Diem | Điểm số của học sinh theo mã học sinh |
| 3 | Truong | Tên trường theo mã trường |

## Mô tả Field trong Table

* 1. DanhSach

### Table 6 - Chi tiết trường của bảng DanhSach

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | MaHS | char | Mã Học Sinh |
| 2 | Ho | char | Họ |
| 3 | Ten | char | Tên |
| 4 | Phai | boolean | Giới Tính |
| 5 | NgaySinh | date | Ngày Sinh |
| 6 | MaTruong | char | Mã Trường |

* 1. Diem

### Table 7 - Chi tiết trường của bảng Diem

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | MaHS | char | Mã Học Sinh |
| 2 | Mon | char | Môn |
| 3 | Diem | float | Điểm số |

* 1. Truong

### Table 8 - Chi tiết trường của bảng Truong

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mục đích** |
| 1 | MaTruong | char | Mã Trường |
| 2 | TenTruong | char | Tên Trường |

## Giao diện chương trình

### Table 9 - Các giao diện

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Màn hình/Cửa Sổ/Dialog** | **Mục đích** | **Giải thích** |
| 1 | Phần mềm quản lý học sinh  Graphical user interface, application  Description automatically generated | Giao diện chính hiển thị dữ liệu từ DanhSach, Diem, Truong | Để cho gọn thì dùng một Listbox nhưng sẽ hiển thị dữ liệu cho 3 table khác nhau tùy vào việc chọn nút bấm nào.  Tất cả các nút bấm đều ở bên phải cho dễ thao tác. |
| 2 | Xem DanhSach  Graphical user interface, application  Description automatically generated | Hiển thị danh sách học sinh. | Khi chọn vào dòng nào thì dữ liệu dòng đó sẽ hiển thị lên các Entry phía trên để cho rõ thông tin. |
|  | Xem Diem  Graphical user interface, application, Teams  Description automatically generated | Hiển thị điểm. | Khi chọn vào dòng nào thì dữ liệu dòng đó sẽ hiển thị lên các Entry phía trên để cho rõ thông tin. |
|  | Xem Truong  Graphical user interface, text, application  Description automatically generated | Hiển thị trường. | Khi chọn vào dòng nào thì dữ liệu dòng đó sẽ hiển thị lên các Entry phía trên để cho rõ thông tin. |
|  | Xóa Bản Ghi  Graphical user interface, application  Description automatically generated | Chọn bất kỳ một dòng nào đó rồi bấm nút “Xoa” | Chọn dòng để dễ dàng xác định cái cần xóa |
|  | Chèn Trước  Graphical user interface, application  Description automatically generated | Chọn một dòng bất kỳ trong danh sách, điểm, trường, và điền thông tin cần chèn vô entry input.  Sau đó nhấn “Chèn Trước” | Cần có các field để lấy thông tin cần để tạo bản ghi, nên sử dụng các Entry input phía trên. |
|  | Chèn Sau  Graphical user interface, application, Teams  Description automatically generated | Chọn một dòng bất kỳ trong danh sách, điểm, trường, và điền thông tin cần chèn vô entry input.  Sau đó nhấn “Chèn Sau” | Cần có các field để lấy thông tin cần để tạo bản ghi, nên sử dụng các Entry input phía trên. |
|  | Yêu cầu số 5 - Danh sach từng HS và điểm  A screenshot of a computer  Description automatically generated with medium confidence | Hiển thị danh sách học sinh và điểm số với xếp loại theo yêu cầu số 5. | Cần có một Listbox mới đủ lớn để hiển thị dữ liệu cho rõ ràng. |
|  | Yêu cầu số 6 - Học sinh có điểm Toán lớn hơn điểm Văn | Hiển thị danh sách học sinh có điểm Toán lớn hơn điểm Văn theo yêu cầu số 6. | Cần có một Listbox mới đủ lớn để hiển thị dữ liệu cho rõ ràng. |
|  | Yêu cầu số 7 - Học sinh Kém  Table  Description automatically generated | Hiển thị danh sách học sinh Kém theo yêu cầu số 7. | Cần có một Listbox mới đủ lớn để hiển thị dữ liệu cho rõ ràng. |
|  | Yêu cầu số 8 - Học Sinh thủ khoa của từng trường có tổng điểm cao nhất  Table  Description automatically generated | Hiển thị danh sách học sinh thủ khoa của từng trường có tổng điểm cao nhất theo yêu cầu số 8. | Cần có một Listbox mới đủ lớn để hiển thị dữ liệu cho rõ ràng. |
|  | Yêu cầu số 9 - Chỉ gồm các học sinh của trường đó  Graphical user interface, application, table  Description automatically generated | Nhập mã trường vào entry input rồi hiển thị học sinh của trường đó theo yêu cầu 9 | Cần một nơi để điền mã trường.  Cần có một Listbox mới đủ lớn để hiển thị dữ liệu cho rõ ràng. |
|  | Yêu cầu số 10 – Giấy báo điểm  Graphical user interface  Description automatically generated | Hiển thị chi tiết giấy báo điểm của 3 học sinh bất kỳ theo yêu cầu số 10. | Thêm button Search để tự dộng lấy 3 học sinh khác nếu cần. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Cài đặt và kiểm thử

### Table 10 - Kiểm thử

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tình huống** | **Mục đích** | **Giải thích** |
| 1 | Chạy chương trình không dùng CSDL.  B1. Trong file frontend.py sẽ import backendtxt.py để làm backend  B2. Chạy chương trình với câu lệnh python frontend.py trên terminal | Kiểm tra chương trình chạy không dùng CSDL | Khi chạy không dùng CSDL thì dùng file backendtxt.py |
| 2 | Thêm học sinh giỏi mới.  B1. Màn hình danh sách chèn học sinh mới với mã HS: 0009, các trường khác đặt bất kỳ, vd: tên: Duong, họ Nguyen,…  B2. Chèn trước hoặc chèn sau trong danh sách  B3. Qua màn hình điểm, chèn điểm số 10, 9 cho 2 môn Toán, Văn với mã HS là: 0009  B4. Xem mục 5. Kết quả có học sinh mã 0009  B5. Xem mục 6. Kết quả có học sinh mã 0009  B6. Xem mục 7. Kết quả không có 0009 trong học sinh kém.  B7. Xem mục 8. Kết quả có 0009 trong top. | Kiểm tra chương trình hiển thị dữ liệu mới một cách chính xác | Kiểm thử một cách hoạt động chương trình từ đầu tới cuối khi chèn một học sinh có điểm đứng top. |
| 3 | Xem danh sách học sinh của trường  B1. Màn hình danh sách học sinh, điền mã trường LQD  B2. Xem mục 9. Kết quả danh sách toàn học sinh của LQD | Kiểm tra chương trình đang lọc đúng dữ liệu. | Dữ liệu đầu vào là mã trường sẽ được đọc chính xác khi tìm kiếm. |
| 4 | Xem giấy báo  B1. Xem mục 10. Kết quả giấy báo của 3 học sinh bất kỳ hiển thị.  B2. Nhấn nút “Search”. Kết quả 3 học sinh khác sẽ hiện ra. | Kiểm tra chức năng xem giấy báo. | Vì là lấy random nên không có dữ liệu đầu vào. |
| 5 | Xóa bản ghi  B1. Vô màn hình trường  B2. Chọn một bản ghi bất kỳ và xóa. Kết quả trường sẽ bị xóa. | Kiểm tra bản ghi được xóa | Xóa trường thành công. |
| 6 | Chạy chương trình dùng CSDL.  B1. Trong file frontend.py sẽ import backend.py để làm backend  B2. Điền username và password của MySQL database vô backend.py  B3. Chạy chương trình với câu lệnh python frontend.py trên terminal.  B4. Tên database phải là:  students\_quochung  khi tạo database trong Mysql | Kiểm tra chương trình chạy dùng CSDL MySQL | Khi chạy dùng CSDL MySQL thì dùng file backend.py |
| 6 | Chạy lại các tình huống kiểm thử 2,3,4,5 | Kiểm tra lại toàn bộ các chức năng khi dùng CSDL MySQL | Việc chạy lại các tình huống này sẽ đãm bảo chương trình hoạt động vẫn đúng với CSDL. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Kết luận:

## Ưu điểm:

* Mức độ hoàn thành yêu cầu là 100%.
* Frontend code và backend code được phân chia rõ ràng.

## Nhược điểm:

* Khi muốn chuyển backend giữa CSDL và không CSDL, chúng ta phải đổi câu import trong frontend. Cách khắc phục: nghiên cứu tạo thêm một file config, và để một biến trong file này để frontend tự biết chạy với backend nào.
* Hiện tại chưa có ràng buộc duy nhất trên MaHS, MaTruong. Cách khắc phục: chèn them các đoạn validation để đảm bảo data khi chèn dữ liệu là chưa tồn tại ở CSDL.
* UI dùng tkinter vẫn còn khá đơn sơ. Cách khắc phục: có thể nâng cấp frontend của chương trình dùng web, khi đó kết hợp với html, css sẽ cho ra giao diện đẹp hơn.
* Chương trình chỉ chạy trên máy local. Cách khắc phục: nâng cấp thành Web application và deploy lên một host, lúc đó có thể truy cập bằng các thiết bị di động cũng như bất kỳ máy tính nào.