**TRƯỜNG ĐẠI HỌC SƯ PHẠM THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**Học phần: Công nghệ JAVA**

−−− 🙡🕮🙣 −−−



**BÁO CÁO ĐỒ ÁN**

**Đề tài: Ứng dụng bán hàng online Shop4Girls**

**Giáo viên hướng dẫn: Lê hoàng Việt Tuấn**

**Sinh viên thực hiện:**

1. **Lý Mỹ Như 44.01.104.031**
2. **Phạm Thị Thanh Thảo 44.01.104.042**
3. **Trịnh Kim Chi 44.01.104.064**
4. **Phạm huỳnh Quốc Duy 44.01.104.069**
5. **Hồ Thị Kim Hà 44.01.104.077**

**Thành phố Hồ Chí Minh, ngày 27 tháng 10 năm 2020**

**LỜI CẢM ƠN!**

“ Đặc biệt, chúng em xin gửi lời cảm ơn sâu sắc đến giảng viên bộ môn – Thầy Lê Hoàng Việt Tuấn đã quan tâm hướng dẫn, truyền đạt những kiến thức quý báu cho chúng em trong suốt thời gian học tập vừa qua. Trong thời gian tham gia lớp học Phát triển ứng dụng giao diện của thầy, chúng em đã có thêm cho mình nhiều kiến thức bổ ích, tinh thần học tập hiệu quả, nghiêm túc. Đây chắc chắn sẽ là những kiến thức quý báu, là hành trang để chúng em có thể vững bước sau này.

Tuy nhiên, do vốn kiến thức còn nhiều hạn chế và khả năng tiếp thu thực tế còn nhiều bỡ ngỡ. Mặc dù, chúng em đã cố gắng hết sức nhưng chắc chắn khó có thể tránh khỏi những thiếu sót và nhiều chỗ còn chưa chính xác, kính mong quý thầy xem xét và góp ý để đồ án của chúng em được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!”

TP.HCM, tháng 5 năm 2020

**LỜI NÓI ĐẦU**

Công nghệ Thông tin là một trong những ngành khoa học ngày càng được quan tâm và sử dụng rộng rãi trong mọi lĩnh vực của cuộc sống. Với những ưu điểm mạnh, có thể ứng dụng được nhiều trong cuộc sống thì công nghệ thông tin đã giúp cho công việc quản lý được dễ dàng hơn. Để quản lý một công việc cụ thể của một cơ quan nào đó thì người quản lý cần thực hiện rất nhiều công việc. Nhờ có các phần mềm thì công việc đó trở nên đơn giản hơn rất nhiều.

Trong những năm gần đây, các chương trình quản lý không còn xa lạ với chúng ta. Quản lý đã xâm nhập vào các lĩnh vực như giáo dục, kinh doanh, ... Đặc biệt trong giáo dục và đào tạo, công tác quản lý học sinh trong quá trình học tập là rất quan trọng, phức tạp và đòi hỏi độ chính xác cao. Đảm bảo mỗi học sinh khi ra trường hoặc đang trong quá trình rèn luyện có thể theo dõi chính xác được quá trình học tập của mình. Trong suốt quá trình học tập, điểm quá trình của học sinh là cơ sở để xét duyệt học bổng, điều kiện lên lớp và phân loại xếp hạng học sinh. Do đó công tác quản lý điểm học sinh đòi hỏi phải có sự thống nhất và chính xác tuyệt đối giúp cho bộ phận quản lý sửa đổi, điều chỉnh hợp lý, kịp thời về phương pháp quản lý và kế hoạch cho đào tạo.

Hệ thống quản lý này sẽ giúp giải quyết được những khó khăn và tăng tính hiệu quả trong công tác quản lý học sinh. Vì sự hiểu biết và kinh nghiệm còn hạn chế nên chắc chắn đồ án của chúng em vẫn còn nhiều thiếu sót. Rất mong nhận được ý kiến đóng góp của thầy và các bạn.

MỤC LỤC

[PHẦN 1: GIỚI THIỆU LẬP TRÌNH ỨNG DỤNG GIAO DIỆN 5](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205145)

[1. Giao diện lập trình ứng dụng là gì? 5](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205146)

[2. Có những loại API nào? 5](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205147)

[3. Ưu điểm của API 6](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205148)

[4. Nhược điểm của API 7](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205149)

[PHẦN 2: PHẦN MỀM QUẢN LÝ HỌC SINH 7](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205150)

[CHƯƠNG 1: TỔNG QUAN HỆ THỐNG QUẢN LÝ HỌC SINH 7](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205151)

[1. Giới thiệu 7](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205152)

[2. Nhiệm vụ hệ thống 7](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205153)

[3. Chức năng của hệ thống 8](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205154)

[4. Các biểu mẫu sử dụng trong hệ thống 8](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205155)

[CHƯƠNG 2: CÁC CHỨC NĂNG CHÍNH CỦA PHẦN MỀM QUẢN LÝ 10](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205156)

[HỌC SINH 10](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205157)

[1. Tiếp nhận học sinh 10](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205158)

[2. Quản lý lớp 11](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205159)

[3. Quản lý điểm học sinh 12](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205160)

[4. Bảng điểm 13](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205161)

[5. Bảng tổng kết 14](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205162)

[6. Các form phụ 16](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205163)

[CHƯƠNG 3: SƠ ĐỒ QUAN HỆ VÀ CÁC LỚP GIẢI QUYẾT CHỨC NĂNG 18](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205164)

[1. Sơ đồ quan hệ 18](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205165)

[2. Các lớp giải quyết vấn đề 19](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205166)

[a. Lớp điểm 19](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205167)

[b. Lớp học sinh 20](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205168)

[c. Lớp điểm trung bình 23](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205169)

[d. Lớp học kì 24](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205170)

[e. Lớp Môn học 26](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205171)

[f. Lớp các lớp 27](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205172)

[g. Lớp tài khoản 29](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205173)

[CHƯƠNG 4: HƯỚNG DẪN SỬ DỤNG PHẦN MỀM 31](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205174)

[CHƯƠNG 5: ĐÁNH GIÁ PHẦN MỀM QUẢN LÝ HỌC SINH 34](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205175)

[Tổng kết 34](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205176)

[1. Khả thi về kỹ thuật 34](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205177)

[2. Công cụ phát triển 34](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205178)

[3. Bảng tổng kết 35](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205179)

[4. Yêu cầu bảo mật 35](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205180)

[5. Ngôn ngữ viết 35](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205181)

[KẾT LUẬN 36](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205182)

[BẢNG PHÂN CÔNG 37](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205183)

[NGUỒN THAM KHẢO: 37](file:///C:\Users\Long\Downloads\MauBaoCaoDoAn.docx#_Toc44205184)

# PHẦN 1: GIỚI THIỆU LẬP TRÌNH ỨNG DỤNG GIAO DIỆN

# Giao diện lập trình ứng dụng là gì?

API (viết tắt từ Application Programming Interface) là một giao diện mà một hệ thống máy tính hay ứng dụng cung cấp cho phép các yêu cầu dịch vụ có thể được tạo ra từ các chương trình máy tính khác, và/hoặc cho phép dữ liệu có thể được trao đổi qua lại giữa chúng. Chẳng hạn, một chương trình máy tính có thể (và thường là phải) dùng các hàm API của [hệ điều hành](https://vi.wikipedia.org/wiki/H%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh) để xin cấp phát bộ nhớ và truy xuất tập tin. Nhiều loại hệ thống và ứng dụng thực hiện API, như các hệ thống đồ họa, cơ sở dữ liệu, mạng, dịch vụ web, và ngay cả một số trò chơi máy tính.

Đây là phần mềm hệ thống cung cấp đầy đủ các chức năng và các tài nguyên mà các lập trình viên có thể rút ra từ đó để tạo nên các tính năng giao tiếp người- máy như: các trình đơn kéo xuống, tên lệnh, hộp hội thoại, lệnh bàn phím và các cửa sổ. Một trình ứng dụng có thể sử dụng nó để yêu cầu và thi hành các dịch vụ cấp thấp do hệ điều hành của máy tính thực hiện. Hệ giao tiếp lập trình ứng dụng giúp ích rất nhiều cho người sử dụng vì nó cho phép tiết kiệm được nhiều thời gian tìm hiểu các chương trình mới, do đó khích lệ mọi người dùng nhiều ứng dụng hơn.

# Có những loại API nào?

* Dựa trên phân ngành trong ngành công nghệ thông tin thì API có các loại sau:

#### **Web API – hệ thống API trên nền tảng web**

Loại này rất phổ biến, các website lớn đều [thiết kế web-app](https://freelancervietnam.vn/thiet-ke-web-application-dich-vu-freelancer-lap-trinh-web-app-co-phai-lua-chon-tot-danh-cho-ban/) với nền tảng hệ thống API giúp bạn kết nối, lấy dữ liệu hoặc đôi khi là cập nhật dữ liệu vào hệ thống.

#### **Hệ thống API trên hệ điều hành**

**Windows**cũng như **Linux** cung cấp các tài liệu API đặc tả các hàm, phương thức và các giao thức kết nối. Nhờ đó, lập trình viên có thể tạo được các phần mềm ứng dụng có chức năng tương tác với hệ điều hành. Đối với Windows thì bạn có thể [tạo ứng dụng chạy trên máy tính bằng C++ và Win32 API](https://docs.microsoft.com/en-us/windows/win32/). Còn với Linux bạn có thể tham khảo thử [Electron API](https://electronjs.org/).

#### **API của thư viện phần mềm (framework)**

API mô tả cũng như quy định các hoạt động mong muốn mà thư viện cung ứng. Một API có thể có nhiều cách triển khai hoạt động khác nhau. API cũng có thể giúp cho một chương trình được viết bằng ngôn ngữ này nhưng có thể sử dụng được thư viện được viết bằng ngôn ngữ khác.

* Dựa trên quyền truy cập thì có thể phân API thành các loại sau:
* **API mở (Open API):** Có sẵn, công khai, không hạn chế quyền truy cập.
* **API đối tác (Partner API):** Cần có quyền hoặc giấy phép cụ thể mới truy cập được.
* **API nội bộ (Internal API):** Chỉ dùng cho các hệ thống nội bộ (chẳng hạn như công ty, tổ chức). Các đội ngũ phát triển nội bộ khác nhau có thể sử dụng chúng để cải thiện cho các sản phẩm hay dịch vụ chỉ phục vụ cho riêng nhân viên công ty.

# **Ưu điểm của API**

**- Tính tự động hóa rất cao:** API có thể thay thế chúng ta quản lý công việc cực kì hiệu quả. API giúp các cơ quan có thể cập nhật, xử lý hoàn thiện công việc nhanh và chất lượng hơn.

**- Ứng dụng cực kỳ linh hoạt:** API có thể truy cập vào các thành phần ứng dụng giúp việc cung cấp dịch vụ và thông tin linh hoạt hơn nhiều.

**- Khả năng thích ứng tốt:** API có chức năng thay đổi cũng như dự đoán thay đổi theo thời gian cho nên dữ liệu được truyền tốt hơn, thông tin được chọn lọc kĩ hơn, dịch vụ tốt hơn.

**- Cá nhân hóa:** Người dùng có thể tinh chỉnh API cho phù hợp nhu cầu sử dụng.

**- Dữ liệu có sẵn:** API cho phép mọi thông tin được tạo ở dạng chính chủ luôn có sẵn cho mọi người xem được.

**- Phạm vi:** Ngoài dữ liệu mới có sẵn được chia sẻ rộng rãi, người dùng còn có thể hiệu chỉnh web API để cung cấp dịch vụ, thông tin cá nhân hóa.

# **Nhược điểm của API**

- API **chỉ hỗ trợ mặc định get, post** chưa hoàn toàn là restful service.

- Cấu hình cố hữu của WFC làm cấu hình của **API cực kì khó nhớ**. Nếu mới sử dụng, người dùng khó mà nhớ và dùng dễ dàng như các ứng dụng khác.

WCF (Windows Communication Foundations) là mô hình phát triển ứng dụng hướng dịch vụ trên nền tảng của Microsoft.

**- Web API tốn kha khá chi phí** vận hành, phát triển, hiệu chỉnh và đòi hỏi kiến thức chuyên sâu. Khi bị tấn công hệ thống, người dùng đôi lúc gặp rắc rối về bảo mật.

# PHẦN 2: PHẦN MỀM QUẢN LÝ HỌC SINH

# CHƯƠNG 1: TỔNG QUAN HỆ THỐNG QUẢN LÝ HỌC SINH

# **Giới thiệu**

Để xây dựng một hệ thống thông tin hoàn chỉnh chúng ta cần trải qua 6 giai đoạn: Khảo sát, phân tích hệ thống, thiết kế hệ thống, kiểm thử, cài đặt, vận hành và bảo trì hệ thống.

Nhưng đối với những học sinh năm 2 như chúng tôi thì nhu cầu cấp thiết nhất hiện nay là qua môn và đạt điểm. Cùng với danh sách đề tài được gợi ý nên vì thế chúng tôi đã xây dựng một chương trình quản lý học sinh để phục vụ nhu cầu này.

# **Nhiệm vụ hệ thống**

Theo dõi quá trình học tập của học sinh trong suốt quá trình học ở trong trường, đảm bảo không sai sót trong quá trình nhập điểm và xuất kết quả. Hệ thống này giúp cho nhà trường theo dõi, đánh giá kết quả học tập của học sinh một cách nhanh nhất và chính xác nhất. Nó sẽ làm giảm đáng kể công sức, giấy tờ, sổ sách lưu trư.

Ngoài việc cập nhật điểm của từng học sinh, hệ thống cũng có thể tra cứu một số thông tin cần thiết về học sinh như ngày sinh, quê quán, chỗ ở hiện tại...

Hệ thống lưu trữ, xử lý kết quả học tập của học sinh. Kiểm soát thống kê việc khen thưởng, xét học bổng của học sinh một cách nhanh chóng và chính xác. Thông qua công tác quản lý mà có thể có hồ sơ cụ thể về kết quả học tập rèn luyện của học sinh khi ra trường.

# **Chức năng của hệ thống**

Hệ thống quản lý điểm học sinh bao gồm 3 bộ phận, hoạt động có mối liên hệ chặt chẽ với nhau.

* Bộ phận quản lý đào tạo: Cập nhật môn học, xây dựng chương trình đạo tạo toàn khóa.
* Bộ phận quản lý học sinh: Bộ phận này có nhiệm vụ quản lý hồ sơ học sinh trong suốt quá trình học tập tại trường, đảm bảo các công việc có liên quan đến học sinh như khi có khóa mới nhập học thì bộ phận này tiến hành cập nhật học sinh khóa mới, danh sách môn học cho học sinh. Đồng thời bộ phận này cũng làm nhiệm vụ lưu hồ sơ của nhưng học sinh đã ra trường vào máy tính. Bộ phận này được sử dụng một PC có cài đặt hệ chương trình gọi là Hồ sơ học sinh (HSHS) trợ giúp các việc như cập nhật, thêm, xóa, sửa thông tin của học sinh.
* Ban khảo thí: Cập nhật điểm từng môn khi kết thúc môn học, học kỳ khi cập nhật có kiểm tra môn học đó. Tính điểm trung bình học tập của mỗi kỳ, của mỗi năm học. Sau đó bộ phận này sẽ thống kê toàn bộ kết quả học tập của học sinh theo từng học kỳ, toàn khóa học. Đồng thời thống kê báo cáo danh sách học sinh thi lại, học lại, thôi học, học bổng… dựa trên các quy định của nhà trường và điểm trung bình chung của học sinh. Chuyển kết quả cho cán bộ khác và học sinh. Bộ phận này cũng có một PC trên đó cài đặt hệ chương trình Nhập điểm trợ giúp việc cập nhật điểm và in phiếu điểm cho từng lớp và cho từng học sinh

# **Các biểu mẫu sử dụng trong hệ thống**

* Thêm học sinh

**THÊM HỌC SINH**

Lớp:

Khóa:

|  |  |  |
| --- | --- | --- |
| 1 | Họ tên | Giới tính |
| 2 | Ngày sinh | Địa chỉ |
| 3 | Email | SĐT |

* Danh sách lớp

**DANH SÁCH LỚP**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lớp: | | | Sỉ số: | |
|  | Họ tên | Giới tính | Năm sinh | Địa chỉ |
| 1 |  |  |  |  |
| 2 |  |  |  |  |

* Bảng điểm môn học

**DANH SÁCH HỌC SINH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lớp: | | | Môn: | |
| Học kì: | | |  | |
| STT | Họ tên | Điểm 15 phút | Điểm 1 tiết | Điểm cuối HK |
| 1 |  |  |  |  |
| 2 |  |  |  |  |

* Báo cáo tổng kết môn

**BÁO CÁO TỔNG KẾT MÔN**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Môn: | | | Học kì: | |
| STT | Lớp | Sỉ số | SL đạt | Tỉ lệ |
| 1 |  |  |  |  |
| 2 |  |  |  |  |

* Báo cáo tổng kết học kì

BÁO CÁO TỔNG KẾT HỌC KÌ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Học kì: | | | | |
| STT | Lớp | Sỉ số | SL đạt | Tỉ lệ |
| 1 |  |  |  |  |
| 2 |  |  |  |  |

# CHƯƠNG 2: CÁC CHỨC NĂNG CHÍNH CỦA PHẦN MỀM QUẢN LÝ

# HỌC SINH

# **Tiếp nhận học sinh**

Form tạo mới học sinh được sử dụng để thêm học sinh mới

Học sinh được nhập phải thỏa mãn các điều kiện về tuổi tối đa, tối thiểu.

Họ tên học sinh, mã học sinh, ngày sinh, giới tính và lớp không được bỏ trống, trong giao diện có nút lệnh

* Thêm học sinh: dùng để thêm học sinh
* Làm mới: Làm mới lại form

# **Quản lý lớp**

Tab danh sách lớp sử dụng bảng HocSinh, Lop của database.

Trong HocSinh lấy cột HoTen, MaHS, GioiTinh, NgaySinh, DiaChi, SoDienThoai, TenNguoiGiamHo, DanToc trong bảng Lop lấy ra danh sách TenLop

Chức năng của form danh sách lớp: xuất ra danh sách lớp, trong giao diện có các nút lệnh

* Tra cứu: Xuất ra danh sách lớp
* Làm mới: Làm mới lại form

# Quản lý điểm học sinh

Tab tra cứu sử dụng bảng HocSinh, Lop, DiemTB của database.

Trong HocSinh lấy cột HoTen, MaHS, trong bảng Lop lấy ra danh sách TenLop, trong DiemTB lấy cột DTB

Chức năng của giao diên: Xuất ra điểm trung bình học kì của các học sinh trong các lớp tương ứng. Có button chức năng:

* Tra cứu: dung để ra cứu danh sách điểm học sinh
* Làm mới: Làm mới lại form

Điểm trung bình được tính theo công thức: (diem15p + diem1t\*2 +diemcuoiHK\*3)/6.

Hạnh kiểm sẽ do bộ phận quản lí đánh giá và xếp loại dựa vào các tiêu chí đánh giá riêng của BCH đoàn ở một hệ thống khác.

# Bảng điểm

Tab bảng điểm sử dụng bảng HocSinh, Lop, MonHoc, Diem, SoHocKi của database.

Trong HocSinh lấy cột HoTen, MaHS, trong bảng Lop lấy ra danh sách TenLop, trong Mon lấy danh sách tên các môn, trong SoHocKi lấy danh sách tên các học kì

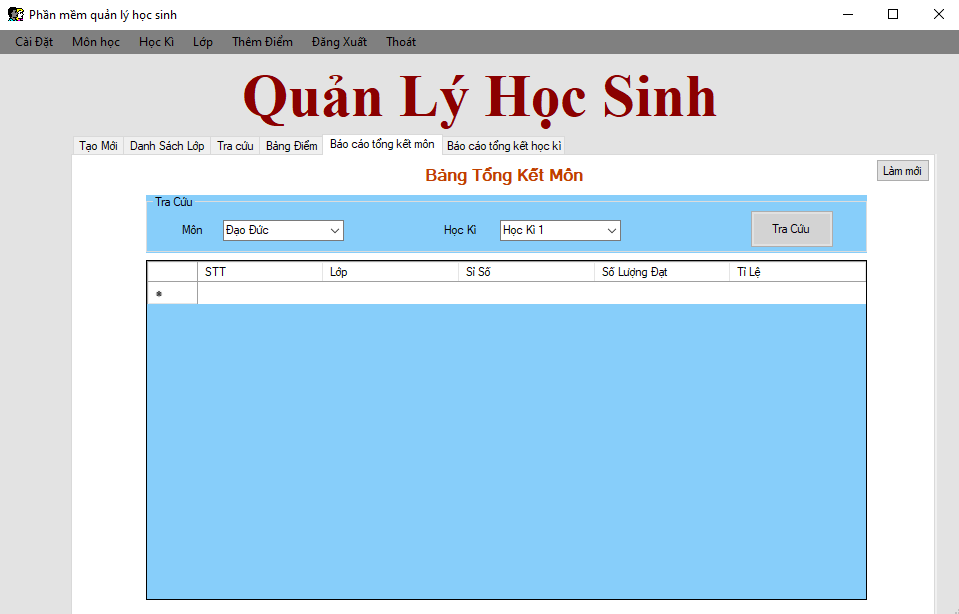
Chức năng của giao diên: Xuất ra điểm theo môn, học kì của các cá nhân học sinh trong các lớp tương ứng. Có button chức năng:

* Tra cứu: dùng để ra cứu điểm của cá nhân từng học sinh
* Làm mới: Làm mới lại form

Điểm trung bình được tính theo công thức: (diem15p + diem1t\*2 +diemcuoiHK\*3)/6.

1. Bảng tổng kết

* **BẢNG TỔNG KẾT MÔN**



Tab bảng tổng kết môn sử dụng bảng Lop, MonHoc, DiemTB, SoHocKi của database.

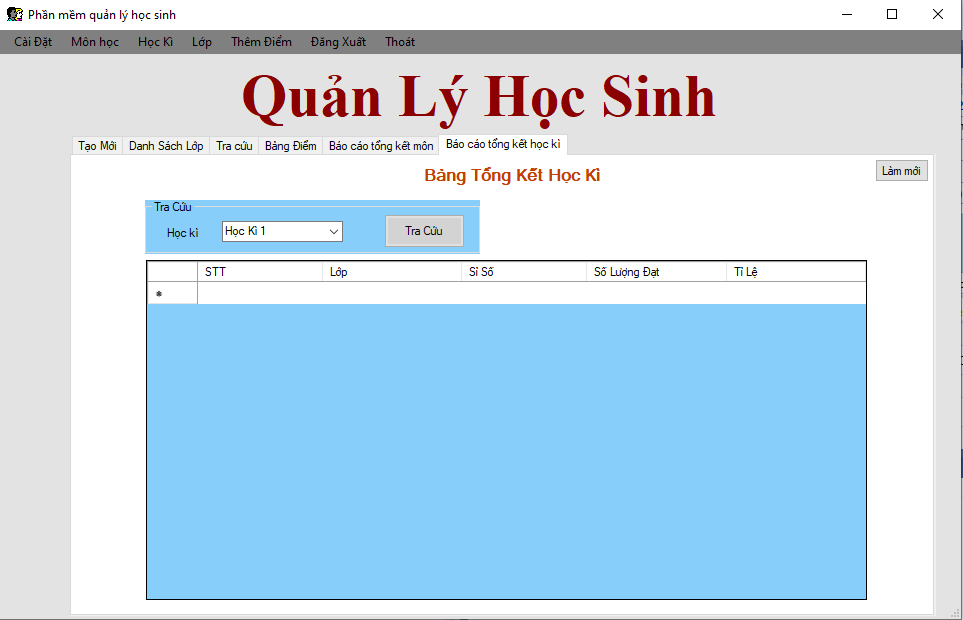
Trong Lớp lấy cột MaLop và TenLop trong bảng DiemTB đếm số lượng học sinh có điểm trung bình phù hợp với điều kiện và xuất ra cột “Số Lượng Đạt”, trong SoHocKi lấy danh sách học kì, trong MonHoc lấy danh sách môn học

Chức năng của giao diên: Xuất ra sỉ số lớp, số lượng các học sinh đạt điểm và tỉ lệ đạt điểm của lớp. Có button chức năng:

* Tra cứu: dùng để ra cứu sỉ số lớp và tỉ lệ
* Làm mới: Làm mới lại form

Tỉ lệ được tính theo công thức: (số lượng đạt /sỉ số)\*100

* **BẢNG TỔNG KẾT HỌC KÌ**

****

Tab bảng tổng kết học kì sử dụng bảng Lop, DiemTBHK, SoHocKi của database.

Trong Lớp lấy cột MaLop và TenLop trong bảng DiemTBHK đếm số lượng học sinh có điểm trung bình phù hợp với điều kiện và xuất ra cột “Số Lượng Đạt”, trong SoHocKi lấy danh sách học kì

Chức năng của giao diên: Xuất ra sỉ số lớp, số lượng các học sinh đạt điểm và tỉ lệ đạt điểm của lớp. Có button chức năng:

* Tra cứu: dùng để ra cứu sỉ số lớp và tỉ lệ
* Làm mới: Làm mới lại form

Số lượng đạt ở đây tính theo công thức:

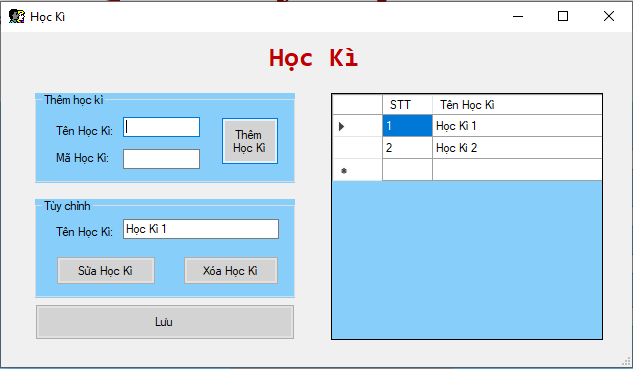
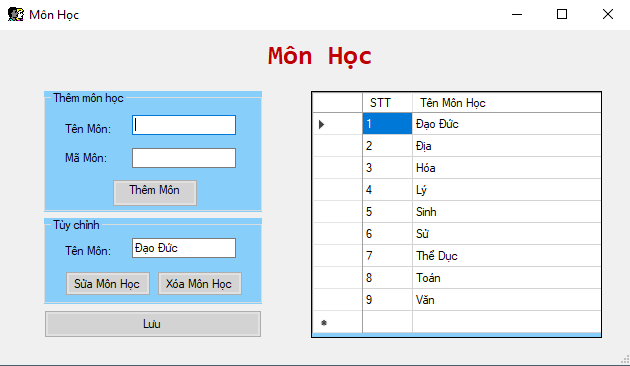
số lượng đạt tăng lên khi DiemTBHK.DTBHK > (DieuKien \* số lượng môn học)

Tỉ lệ được tính theo công thức: (số lượng đạt /sỉ số)\*100

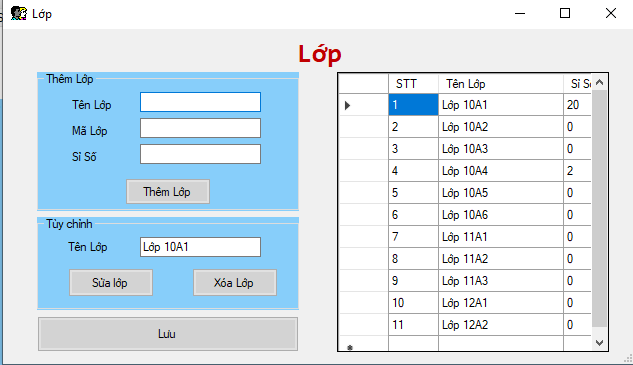
1. Các form phụ

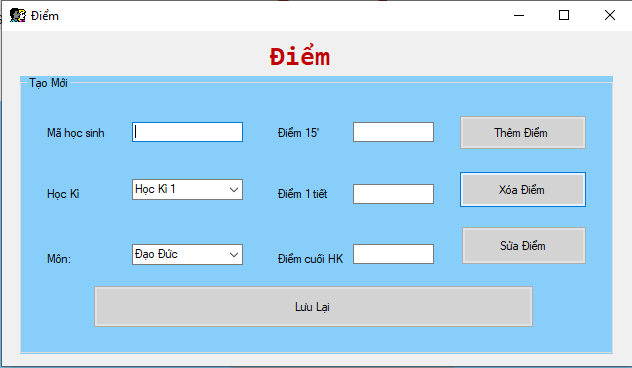
Form cài đặt: cài đặt lại các điều kiện như độ tuổi, sỉ số lớp, điểm chuẩn đạt môn

Form môn học: Điều chỉnh lại môn học, thêm sửa xóa

Form học kì: Điều chỉnh lại học kì, thêm sửa xóa

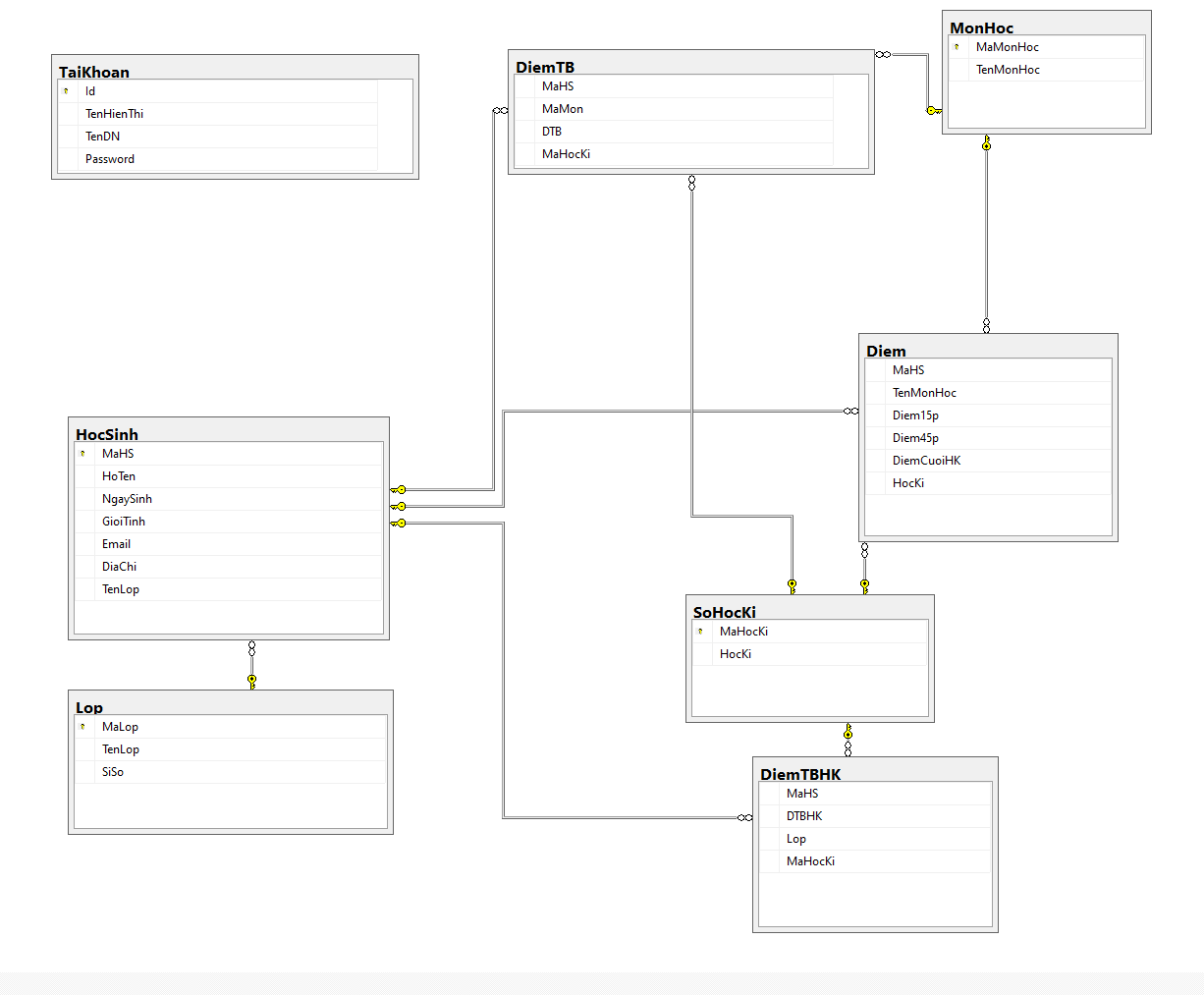
Form lớp: Điều chỉnh lại lớp, thêm sửa xóa



Form điểm: Điều chỉnh lại điểm, thêm sửa xóa

# CHƯƠNG 3: SƠ ĐỒ QUAN HỆ VÀ CÁC LỚP GIẢI QUYẾT CHỨC NĂNG

# Sơ đồ quan hệ



1. Các lớp giải quyết vấn đề

## **Lớp điểm**

using ProjectQLHocSinh.DAO;

using ProjectQUANLyHS.Entities;

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ProjectQUANLyHS.Business

{

class DiemBLL

{

private static volatile DiemBLL instance; //tranh xung dot giua cac thread

public static DiemBLL Instance

{

get

{

if (instance == null)

{

instance = new DiemBLL();

}

return DiemBLL.instance;

}

}

private DiemBLL()

{ }

//Thêm điểm

public bool AddDiem(DiemEntities Diem)

{

string cautruyvan = "exec USP\_ThemDiem @MaHS='"+Diem.MaHS1+"',@TenMonHoc=N'"+Diem.TenMonHoc1+"',@Diem15p="+Diem.Diem15p1+",@Diem45p="+Diem.Diem45p1+ ",@DiemCuoiHK="+Diem.DiemCuoiHK1+",@HocKi='"+Diem.HocKi1+"'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Load danh sách điểm

public DataTable LoadDiem()

{

string cautruyvan = "select Diem.MaHS,HocSinh.HoTen,Diem.Diem15p,Diem.Diem45p,Diem.DiemCuoiHK from HocSinh,Diem where Diem.MaHS=HocSinh.MaHS";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Sửa điểm

public bool UpdateDiem(DiemEntities Diem,float Diem15p,float Diem45p,float DiemCuoiHK)

{

string cautruyvan = "exec USP\_SuaDiem @MaHS='"+Diem.MaHS1+"',@Diem15p="+Diem15p+",@Diem45p="+Diem45p+",@DiemCuoiHK="+DiemCuoiHK+"";

return DataProvider.Instance.exedata(cautruyvan);

}

//Xóa điểm

public bool DeleteDiem(DiemEntities Diem)

{

string cautruyvan = "update Diem set Diem.Diem15p=0,Diem.Diem45p=0,Diem.DiemCuoiHK=0 where Diem.MaHS='"+Diem.MaHS1+"'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Tra cứu bảng điểm HS

public DataTable TraCuu(string TenLop,string TenMonHoc,string HocKi)

{

string cautruyvan = "exec USP\_TraCuuBangDiem @TenLop='" + TenLop + "',@TenMonHoc=N'" + TenMonHoc + "',@HocKi='" + HocKi + "'";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Lấy tên học sinh từ mã học sinh bảng điểm

public string LoadTenHs(string MaHS)

{

string cautruyvan = "select HocSinh.HoTen from HocSinh where HocSinh.MaHS='"+MaHS+"'";

return DataProvider.Instance.ExcuteScalar(cautruyvan).ToString();

}

//Load bảng tổng kết môn:

/\* public DataTable LoadBTKMon()

{

string cautruyvan="select Lop.TenLop,(select Count(\*) from HocSinh) from Lop"

}\*/

}

}

## **Lớp học sinh**

using ProjectQLHocSinh.DAO;

using ProjectQUANLyHS.Entities;

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ProjectQUANLyHS.Business

{

class HocSinhBLL

{

private static volatile HocSinhBLL instance; //tranh xung dot giua cac thread

public static HocSinhBLL Instance

{

get

{

if (instance == null)

{

instance = new HocSinhBLL();

}

return HocSinhBLL.instance;

}

}

private HocSinhBLL()

{ }

//hàm lấy danh sách tất cả học sinh

public DataTable LoadAllHocSinh()

{

string cautruyvan = "select \* from HocSinh";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//thêm học sinh

public bool AddHocSinh(HocSinhEntities HocSinh)

{

string cautruyvan="exec USP\_ThemHS @MaHS='"+HocSinh.MaHs1+"',@Hoten=N'" + HocSinh.HoTenHs1 + "',@NgaySinh='" + HocSinh.NgaySinhHs1 + "',@GioiTinh=N'" + HocSinh.GioiTinhHs1 + "',@Email=N'" + HocSinh.EmailHs1 + "',@DiaChi=N'" + HocSinh.DiaChiHs1 + "',@TenLop='" + HocSinh.TenLopHs1 + "'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Load danh sách học sinh theo lớp

public DataTable LoadHocSinh(HocSinhEntities HocSinh)

{

string cautruyvan = "select \* from HocSinh where HocSinh.TenLop = '"+HocSinh.TenLopHs1+"'";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Sỉ số học sinh

public int LoadSiSo(HocSinhEntities HocSinh)

{

string cautruyvan ="exec USP\_SiSo @Lop = '"+HocSinh.TenLopHs1+"'";

return (int)DataProvider.Instance.ExcuteScalar(cautruyvan);

}

//Danh sách các lớp

public DataTable LoadDSLop()

{

string cautruyvan = "select Lop.TenLop from Lop ";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Danh sách tên 1 học sinh trong lớp

public DataTable LoadTenHS(HocSinhEntities HocSinh)

{

string cautruyvan = "select HoTen from HocSinh where HocSinh.TenLop='"+HocSinh.TenLopHs1+"'";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Tra cứu họ tên và DTB của học sinh

public float TinhGiaTriTrungBinh(string MaHS,string HocKi)

{

/\*string truyvandiem15p = "select Sum(distinct Diem.Diem15p) from Diem, HocSinh, SoHocKi where Diem.MaHS = '"+MaHS+"' and Diem.HocKi = SoHocKi.MaHocKi and SoHocKi.MaHocKi = '"+HocKi+"'";

string truyvandiem1t = "select Sum(distinct Diem.Diem45p) from Diem, HocSinh, SoHocKi where Diem.MaHS = '" + MaHS + "' and Diem.HocKi = SoHocKi.MaHocKi and SoHocKi.MaHocKi = '" + HocKi + "'";

string truyvandiemCuoiHK = "select Sum(distinct Diem.DiemCuoiHK) from Diem, HocSinh, SoHocKi where Diem.MaHS = '" + MaHS + "' and Diem.HocKi = SoHocKi.MaHocKi and SoHocKi.MaHocKi = '" + HocKi + "'";

string truyvansoluongmon = "select Count(\*) from MonHoc";

Double Diem15p = Convert.ToDouble(DataProvider.Instance.ExcuteScalar(truyvandiem15p));

Double Diem1t = Convert.ToDouble(DataProvider.Instance.ExcuteScalar(truyvandiem1t));

Double DiemCuoiHK = Convert.ToDouble(DataProvider.Instance.ExcuteScalar(truyvandiemCuoiHK));

Double SoLuongMon = Convert.ToDouble(DataProvider.Instance.ExcuteScalar(truyvansoluongmon));

return ((Diem15p + Diem1t \* 2 + DiemCuoiHK \* 3) / (6\*SoLuongMon));\*/

string cautruyvan1 = "select Sum(DiemTB.DTB) from DiemTB where DiemTB.MaHS='" + MaHS + "' and DiemTB.MaHocKi='" + HocKi + "' ";

float result = float.Parse(DataProvider.Instance.ExcuteScalar(cautruyvan1).ToString());

string cautruyvan2 = "select count(MonHoc.MaMonHoc) from MonHoc";

int SoMonHoc = Int32.Parse(DataProvider.Instance.ExcuteScalar(cautruyvan2).ToString());

return (float)(result/SoMonHoc);

}

//Load bảng tra cứu học sinh

public DataTable LoadBangTraCuuHS(string MaLop)

{

string cautruyvan = "select HocSinh.MaHS,HocSinh.HoTen,Lop.TenLop from HocSinh,Lop where HocSinh.TenLop=Lop.MaLop and Lop.MaLop='"+MaLop+"'";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Thêm mã HS vào DS mã HS

public List<string> DSMaHS = new List<string>();

public void AddDSMaHS(string MaHS)

{

List<string> DSMaHS = new List<string>();

DSMaHS.Add(MaHS);

}

//Lấy mã HS

}

}

## **Lớp điểm trung bình**

using ProjectQLHocSinh.DAO;

using ProjectQUANLyHS.Entities;

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ProjectQUANLyHS.Business

{

class DiemTBBLL

{

private static volatile DiemTBBLL instance; //tranh xung dot giua cac thread

public static DiemTBBLL Instance

{

get

{

if (instance == null)

{

instance = new DiemTBBLL();

}

return DiemTBBLL.instance;

}

}

private DiemTBBLL()

{ }

public bool AddDTB(DiemTBEntities DiemTB)

{

string cautruyvan = "insert into DiemTB values('"+DiemTB.MaHS1+"','"+DiemTB.MaMon1+"',"+DiemTB.DTB1+",'"+DiemTB.MaHocKi1+"')";

return DataProvider.Instance.exedata(cautruyvan);

}

public int GetSlDat(DiemTBEntities DiemTB,string Lop)

{

string cautruyvan = "Select count(HocSinh.MaHS) from HocSinh,DiemTB where HocSinh.MaHS=DiemTB.MaHS and DiemTB.MaMon='"+DiemTB.MaMon1+"' and DiemTB.MaHocKi='"+DiemTB.MaHocKi1+"' and HocSinh.TenLop='"+Lop+"' and DiemTB.DTB>"+DiemTB.DK1+" ";

int result = (int)(DataProvider.Instance.ExcuteScalar(cautruyvan));

return result;

}

public int GetSLDatBangTongKet(DiemTBEntities Diem,string Lop)

{

string cautruyvan = "select count(HocSinh.MaHS) from DiemTB,HocSinh where DiemTB.MaHS = HocSinh.MaHS and HocSinh.TenLop = '"+Lop+"'and DiemTB.MaHocKi = '"+Diem.MaHocKi1+"' and DiemTB.DTB > "+Diem.DK1+" ";

return (int)DataProvider.Instance.ExcuteScalar(cautruyvan);

}

}

}

## **Lớp học kì**

using ProjectQLHocSinh.DAO;

using ProjectQUANLyHS.Entities;

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ProjectQUANLyHS.Business

{

class HocKiBLL

{

private static volatile HocKiBLL instance; //tranh xung dot giua cac thread

public static HocKiBLL Instance

{

get

{

if (instance == null)

{

instance = new HocKiBLL();

}

return HocKiBLL.instance;

}

}

private HocKiBLL()

{ }

//Thêm học kì

public bool AddHocKi(HocKiEntities HocKi )

{

return DataProvider.Instance.exedata("exec USP\_ThemHocKi @MaHocKi='"+HocKi.MaHocKi1+"',@HocKi='"+HocKi.HocKi1+"'");

}

//Lấy danh sách học kì

public DataTable LoadHocKi()

{

string cautruyvan = "select \* from SoHocKi";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Sửa học kì

public bool UpdateHocKi(string HocKiSua, string DieuKienSua)

{

string cautruyvan = "UPDATE SoHocKi SET SoHocKi.HocKi = '" + HocKiSua + "' WHERE SoHocKi.MaHocKi='" + GetMaHocKi(DieuKienSua) + "'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Xóa học kì

public bool DeleteHocKi(string TenHocKi)

{

string cautruyvan = "delete from SoHocKi where SoHocKi.MaHocKi='" + GetMaHocKi(TenHocKi) + "'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Lấy ra mã học kì từ tên học kì

public string GetMaHocKi(string HocKi)

{

string cautruyvan = "select SoHocKi.MaHocKi from SoHocKi where SoHocKi.HocKi='"+HocKi+"'";

return DataProvider.Instance.ExcuteScalar(cautruyvan).ToString();

}

}

}

## **Lớp Môn học**

using ProjectQLHocSinh.DAO;

using ProjectQUANLyHS.Entities;

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ProjectQUANLyHS.Business

{

class MonHocBLL

{

private static volatile MonHocBLL instance; //tranh xung dot giua cac thread

public static MonHocBLL Instance

{

get

{

if (instance == null)

{

instance = new MonHocBLL();

}

return MonHocBLL.instance;

}

}

private MonHocBLL()

{ }

//thêm tên môn học

public bool AddMonHoc(MonHocEntities MonHoc)

{

return DataProvider.Instance.exedata("exec USP\_ThemMonHoc @TenMonHoc=N'"+MonHoc.TenMonHoc1+"',@MaMonHoc='"+ MonHoc.MaMonHoc1+"'");

}

//Load bảng danh sách môn học

public DataTable LoadMonHoc()

{

string cautruyvan = "select \* from MonHoc";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Sửa môn học

public bool UpdateMonHoc(string MonHocSua,string DieuKienSua)

{

string cautruyvan = "UPDATE MonHoc SET MonHoc.TenMonHoc = '"+ MonHocSua + "' WHERE MonHoc.MaMonHoc=N'"+GetMaMonHoc(DieuKienSua)+"'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Xóa môn học

public bool DeleteMonHoc(string TenMonHoc)

{

string cautruyvan = "delete from MonHoc where MonHoc.MaMonHoc=N'"+GetMaMonHoc(TenMonHoc)+"'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Lấy mã môn học từ tên môn học

public string GetMaMonHoc(string TenMonHoc)

{

string cautruyvan = "select MonHoc.MaMonHoc from MonHoc where MonHoc.TenMonHoc=N'"+TenMonHoc+"'";

return DataProvider.Instance.ExcuteScalar(cautruyvan).ToString();

}

//Lay bao cao tong ket mon

}

}

## **Lớp các lớp**

using ProjectQLHocSinh.DAO;

using ProjectQUANLyHS.Entities;

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ProjectQUANLyHS.Business

{

class LopBLL

{

private static volatile LopBLL instance; //tranh xung dot giua cac thread

public static LopBLL Instance

{

get

{

if (instance == null)

{

instance = new LopBLL();

}

return LopBLL.instance;

}

}

private LopBLL()

{ }

//thêm lớp

public bool AddLop(LopEntities Lop)

{

return DataProvider.Instance.exedata("insert into Lop values('"+Lop.MaLop1+"','"+Lop.TenLop1+"',"+Lop.SiSo1+")");

}

//Load bảng danh sách lớp

public DataTable LoadLop()

{

string cautruyvan = "select Lop.TenLop from Lop";

return DataProvider.Instance.ExcuteQuery(cautruyvan);

}

//Sửa lớp

public bool UpdateLop(string LopSua, string DieuKienSua)

{

string cautruyvan = "UPDATE Lop SET Lop.TenLop = '" + LopSua + "' WHERE Lop.MaLop='" + GetMaLop(DieuKienSua) + "'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Sửa sỉ số

public bool UpdateSiSo(int SiSoSua,string DieuKien)

{

string cautruyvan = "UPDATE Lop SET Lop.SiSo = " + SiSoSua + " WHERE Lop.MaLop='" + GetMaLop(DieuKien) + "'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Xóa lớp

public bool DeleteLop(string Lop)

{

string cautruyvan = "delete from Lop where MaLop='" + GetMaLop(Lop) + "'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Xóa sỉ số

public bool DeleteSiSo(string MaLop)

{

string cautruyvan = "update Lop set Lop.SiSo=0 where Lop.MaLop='" + GetMaLop(MaLop) + "'";

return DataProvider.Instance.exedata(cautruyvan);

}

//Lấy mã lớp từ tên

public string GetMaLop(string TenLop)

{

string cautruyvan = "select Lop.MaLop from Lop where Lop.TenLop='"+TenLop+"'";

return DataProvider.Instance.ExcuteScalar(cautruyvan).ToString();

}

//Lay sI So toi da

public int GetSiSoToiDa( string TenLop)

{

string cautruyvan = "select Lop.SiSo from Lop where Lop.MaLop='" + GetMaLop(TenLop) + "'";

return Convert.ToInt32(DataProvider.Instance.ExcuteScalar(cautruyvan));

}

public int GetSiSoThuc(string MaLop)

{

string cautruyvan="select count(HocSinh.MaHS) from HocSinh where HocSinh.TenLop='"+MaLop+"'";

return Convert.ToInt32(DataProvider.Instance.ExcuteScalar(cautruyvan));

}

public string GetTenLop(string Ten)

{

return DataProvider.Instance.ExcuteQuery("select Lop.TenLop from Lop where Lop.MaLop='" + Ten + "'").ToString();

}

}

}

## **Lớp tài khoản**

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ProjectQLHocSinh.DAO

{

class Account

{

private static volatile Account instance;

static object key = new object();

public static Account Instance

{

get

{

if (instance == null)

{

lock (key)

{

instance = new Account();

}

}

return Account.instance;

}

private set => instance = value;

}

private Account()

{ }

public bool Login(string UserName, string PassWord)

{

string query = "USP\_Login @userName , @passWord";

DataTable result = DataProvider.Instance.ExcuteQuery(query, new object[] { UserName, PassWord });

return result.Rows.Count > 0;

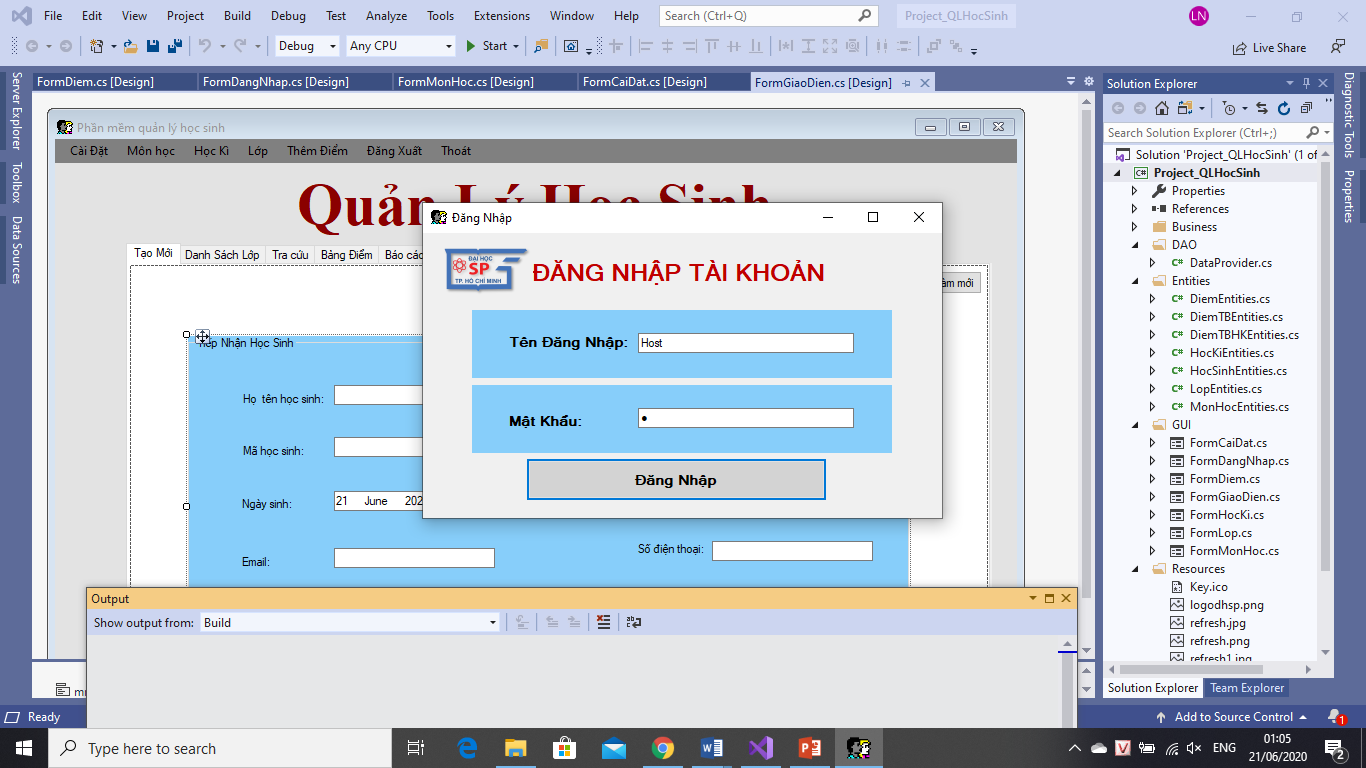
}

}

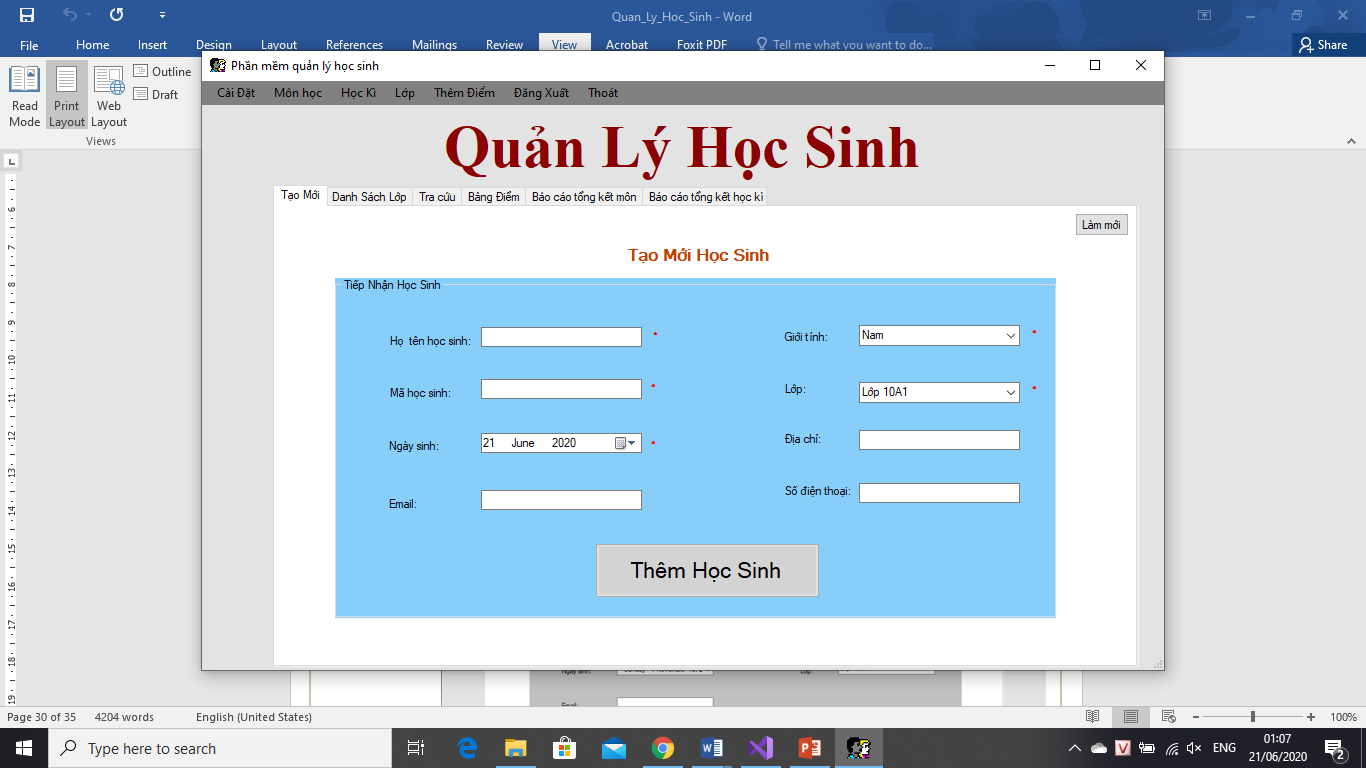
}

# CHƯƠNG 4: HƯỚNG DẪN SỬ DỤNG PHẦN MỀM

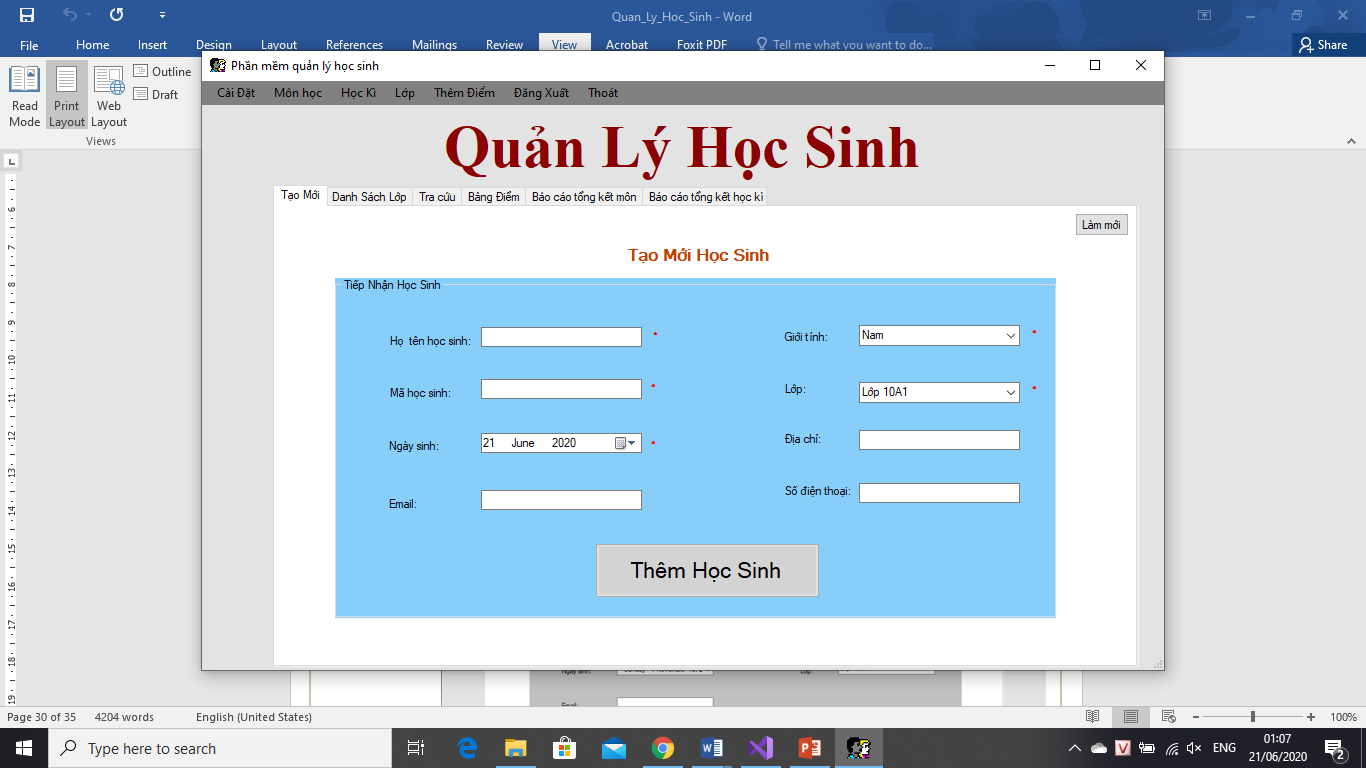
Sau đây nhóm sẽ hướng dẫn sử dụng phần mềm Quản lý điểm học sinh như sau:

Đầu tiên là màn hình đăng nhập. Cần nhập đúng tên và tài khoản để sử dụng. Đã mặc định tài khoản là “host” là mật khẩu là “0”

Sau khi đăng nhập thành công, giao diện chính hệ thống sẽ xuất hiện:

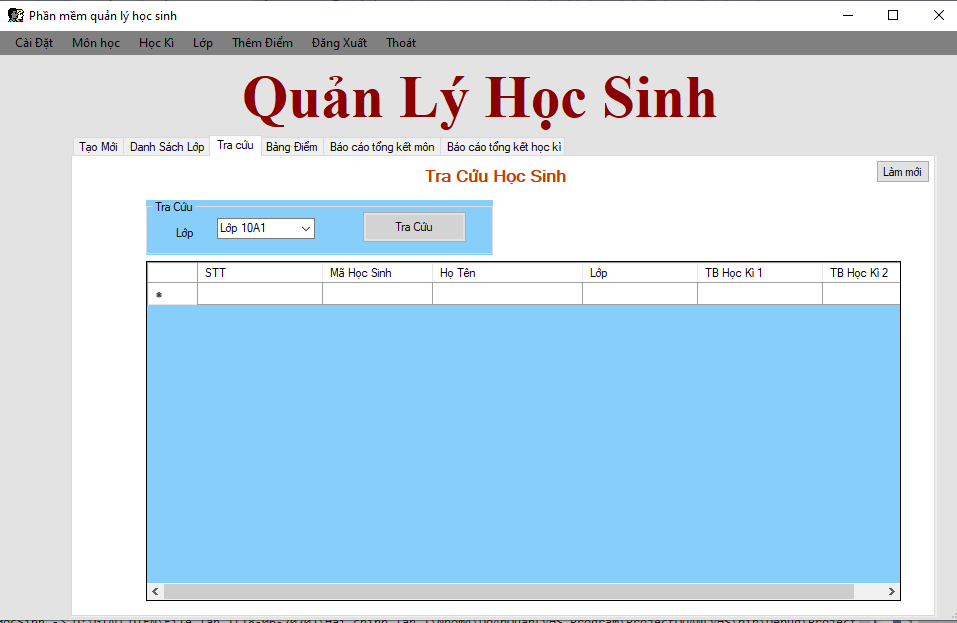


Nhập thông tin học sinh

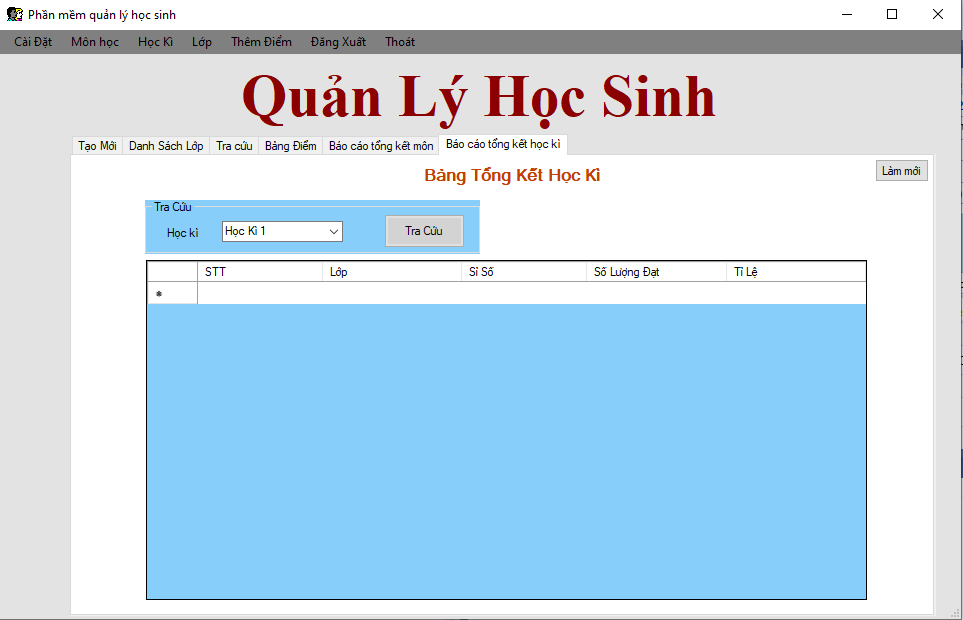


Sau đó có thể truy xuất thông tin học sinh

 Danh sách lớp

Điểm trung bình

Trung bình học kì theo môn

Trung bình học kì theo học kì

# CHƯƠNG 5: ĐÁNH GIÁ PHẦN MỀM QUẢN LÝ HỌC SINH

# Tổng kết

# 1. **Khả thi về kỹ thuật**

-Thành viên tham gia dự án: 4 thành viên

- Thời gian thực hiện: 4 tuần

- Kích thước hệ thống: trung bình

# 2. **Công cụ phát triển**

- Hệ điều hành Windows 10

- Công cụ lập trình: Microsoft Visual Studio 2019 (ngôn ngữ C sharp)

- Cơ sở dữ liệu: Microsoft SQL Server 2012

- Công cụ khác: Word, PPT…

# 3. **Bảng tổng kết**

| **STT** | **Rủi ro** | **Mức độ** | **Mô tả đánh giá** | **Chú thích** |
| --- | --- | --- | --- | --- |
| 1 | Tâm lý người dùng chưa quen với hệ thống mới | Trung bình | Nhân viên nơi làm việc đều có khả năng sử dụng máy vi tính | Tổ chức đào tạo huấn luyện cho người sử dụng hệ thống |
| Hỗ trợ trực tiếp người dùng trong thời gian bắt đầu.  Tiến hành triển khai thử từng bước. |
| 2 | Kích thước hệ thống | Trung bình | Thời gian triển khai hệ thống là 1 tuần |  |
| 3 | Chuyển từ cách hoạt động từ thủ công sang sử dụng phần mềm | Bình thường | Nhân viên sẽ nhanh chóng thích nghi với việc sử dụng hệ thống mới | Đào tạo một lớp sử dụng phần mềm trước khi sử dụng |
| 4 | Dữ liệu không thể phục hồi khi xóa | Cao | Chọn chức năng xóa | Có cảnh báo |
| 5 | Nhập dữ liệu sai | Khá cao | Trong quá trình nhập không tránh khỏi khả năng nhập sai | Có cảnh báo |
| 6 | Mở thêm tính năng cho phần mềm | Thấp | Khi muốn mở tính năng mới cho phần mềm | Nhóm phát triển phải chỉ cho phép người dùng hoạt động trên những model có sẵn |

1. Yêu cầu bảo mật

Chỉ người có user + password của hệ thống mới có quyền truy cập quản trị hệ thống

# **Ngôn ngữ viết**

Chương trình sử dụng ngôn ngữ C #:

* Phát triển tại Microsoft, đội nghiên cứu do Anders Hejlsberg và Scott Wiltamuth lãnh đạo
* Điều khiển sự kiện, hướng đối tượng, ngôn ngữ lập trình trực quan
* Dựa trên nền tảng C, C++ và Java
* Hợp nhất trong nền .NET platform
* Có thể phân phối các ứng dụng Web
* Các thiết bị và máy tính để bàn
* Các chương trình cho phép mọi người truy cập qua bất cứ thiết bị nào
* Cho phép giao tiếp với các ngôn ngữ máy khác.
* Integrated Design Environment (IDE)-Môi trường tích hợp phát triển phần mềm
* Dễ dàng hoá việc lập trình và gỡ lỗi.
* Phát triển ứng dụng nhanh- Rapid Application Development (RAD)

# KẾT LUẬN

* Ưu điểm:
* Các chức năng cơ bản của chương trình đã hoàn thành.
* Giao diện thân thiện, dễ sử dụng.
* Cài đặt trên nhiều hệ điều hành
* Có trợ giúp hướng dẫn sử dụng
* Nhược điểm
* Dữ liệu cập nhật hơi chậm.
* Độ bảo mật không cao.
* Thời gian hạn chế và nhóm chỉ có 4 thành viên cho nên trong quá trình xây dựng phần mềm còn gặp nhiều khó khăn.
* Chưa có kinh nghiệm nhiều trong việc xây dựng một chương trình quản lý với quy mô lớn.
* Hầu hết các chức năng hệ thống đã hoàn chỉnh.

* Kinh nghiệm thu được:
* Hiểu được nhiều kinh nghiệm trong giao tiếp cũng như công việc thực tế
* Có cơ hội để vận dụng kiến thức lý thuyết trong nhà trường áp dụng vào thực tiễn.
* Hướng giải quyết:

Phần mềm được viết với các chức năng cơ bản và có thể dựa vào nó để nâng cấp lên quản trị thêm nhiều chức năng hơn, như chức năng quản lý hạnh kiểm học sinh (số buổi vắng, các vi phạm), quản lý điểm tổng kết 3 năm và điểm thi tốt nghiệp để xét loại tốt nghiệp cho học sinh , quản lý học sinh diện con nhà nghèo,...

Link GitHub: <https://github.com/PTUDGD-TEAM-4/DoAnQuanlyHS>

Link Youtube: https://youtu.be/\_TyHMp6Amqs

# BẢNG PHÂN CÔNG

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ tên** | **Nhiệm vụ** |
| 44.01.103.001 | Trần Thị Xuân Diễm | Lên ý tưởng, bìa báo cáo, chỉnh word và thiết kế giao diện. |
| 44.01.103.004 | Đặng Sơn Hải | Phân công chia công việc cho các thành viên, thiết kế giao diện, điều chỉnh PPT, lên ý tưởng. |
| 44.01.103.025 | Ngô Thị Lượm | Thiết kế giao diện, soạn word, lên ý tưởng |
| 44.01.103.041 | Nguyễn Ngọc Bảo Trân | Lên ý tưởng bài, thiết kế giao diện, soạn PPT, tạo cơ sở dữ liệu Database. |

# NGUỒN THAM KHẢO:

#### **[1] API là gì? Giới thiệu về Giao diện lập trình ứng dụng API- Freelancervn (2019). Available at: https://freelancervietnam.vn/gioi-thieu-ve-giao-dien-lap-trinh-ung-dung-api/#:~:text=API%20%E2%80%93%20Application%20Programming%20Interface%20%E2%80%93%20giao,%C4%91%E1%BB%95i%20th%C3%B4ng%20tin%20v%E1%BB%9Bi%20nhau. (Accessed: 22 June 2020).**

#### **[2] Lập trình Winform cơ bản | Website hướng dẫn lập trình miễn phí | How Kteam | Free Education (2020). Available at: https://www.howkteam.vn/course/lap-trinh-winform-co-ban-27 (Accessed: 22 June 2020).**

#### **[3]Hồng, T., Tiến, P. and Tiến, P. (2015) Code Export dữ liệu từ DataGrid tới PDF với C# | Tìm ở đây, Timoday.edu.vn. Available at: https://timoday.edu.vn/code-export-du-lieu-tu-datagrid-toi-pdf-voi-c/ (Accessed: 22 June 2020).**

[4] link youtube: <https://www.youtube.com/watch?v=NNXNYZWdyrg>

<https://www.youtube.com/watch?v=h7OYMHCa2WE&t=1235s>

#### **[5] Báo cáo thực tập cuối khóa Quản lý học sinh trường cấp 3 (2019). Available at: https://hotroontap.com/bao-cao-thuc-tap-cuoi-khoa-quan-ly-hoc-sinh-truong-cap-3/ (Accessed: 22 June 2020).**