Mục lục

Group 1 Stage 3, 4 Assignment

Truong Do Vuong

Ngày 4 tháng 11 năm 2025

1 Cấu trúc Dự án và Tổ chức MVC

1.1 Cấu trúc Thư mục MVC

Sử dụng cấu trúc thư mục MVC để mã nguồn được tổ chức khoa học và dễ hiểu. Ví dụ:

```
src/
 models/        # lớp dữ liệu (vd: User, Payment, Announcement)
 views/         # mã giao diện GUI (các cửa sổ và hộp thoại Tkinter)
 controllers/   # logic ứng dụng, liên kết giữa views và models
 main.py        # điểm khởi đầu của ứng dụng (khởi tạo MVC và chạy app)
```

Các bộ phận là độc lập:

- **Models** chứa dữ liệu và logic MongoDB.
- **Views** tạo giao diện người dùng.
- **Controllers** xử lý thao tác người dùng và cập nhật models/views.

Trong MVC, view chỉ hiển thị dữ liệu, controller xử lý đầu vào và cập nhật model, còn khi dữ liệu model thay đổi thì view được cập nhật. Nói cách khác, *"model không biết gì về view hay controller, view cũng không biết gì về model hay controller, còn controller là biết cả hai"*. Cách tách biệt này giúp mô-đun hóa và dễ quản lý.

Ví dụ trong ứng dụng của nhóm:

- Các lớp giao diện CustomTkinter (form đăng nhập, đăng ký, bảng điều khiển, ...) đặt trong `views/`.
- Các Lớp đọc/ghi cơ sở dữ liệu MongoDB đặt trong `models/`.
- Các hàm kết nối sự kiện UI với cơ sở dữ liệu (ví dụ: `onRegisterClick`) đặt trong `controllers/`.

Một ví dụ thực tế có thể gồm các tệp như `models/auth.py`, `views/signup.py`, `controllers/signup.py`.

1.2 Phân công Vai trò và Nhiệm vụ Nhóm

Trương Đỗ Vương – Điều phối và phụ trách giao diện GUI. Thiết lập Git và cấu trúc thư mục ban đầu. Xây dựng khung ứng dụng chính và cửa sổ dashboard. Tích hợp các phần và hướng dẫn các thành viên khác. Lập trình giao diện Tkinter cho các trang chính (dashboard quản trị với bảng sinh viên/thanh toán) và đảm bảo ứng dụng tuân thủ MVC.

Hoàng Văn Hưng – Phụ trách Cơ sở dữ liệu / Models. Chuyển thiết kế CSDL sang các lớp Python model. Sử dụng pymongo để kết nối với MongoDB Tạo các hàm CRUD cho sinh viên, thanh toán, thông báo. Viết lớp `StudentModel` với các phương thức như `create_student()`, `find_student()`, v.v.

Lê Huỳnh Anh Khôi – Phụ trách Controller / Logic. Viết logic ứng dụng kết nối giao diện và cơ sở dữ liệu. Ví dụ: khi người dùng nhấn "Login", controller kiểm tra thông tin đăng nhập trong model; khi admin nhấn "Create Account", controller gọi hàm tạo tài khoản từ model. Cài đặt các luồng hoạt động: đăng nhập, đăng ký, cập nhật hồ sơ, thanh toán học phí, đăng thông báo. Phối hợp với Hưng về model API và với Vương về view methods.

Nguyễn Trần Việt Nhật – Hỗ trợ đa tuyến.

- Tạo form UI đơn giản (form quên mật khẩu, trang xem hồ sơ, trang xem thông báo).
- kiểm tra dữ liệu đầu vào, kiểm thử, viết tài liệu.
- Cài đặt mô phỏng gửi email phục hồi (sử dụng `smtplib`).
- Xử lý các tính năng nhỏ hoặc tĩnh như hiển thị hướng dẫn, viết README.

1.3 Phân chia Chức năng theo Module

- **Đăng ký & Tài khoản:** Hưng/Khôi (model/controller), Vương/Nhật (giao diện).
- **Đăng nhập:** Khôi (controller), Vương (UI).
- **Chỉnh sửa hồ sơ:** Khôi/Hưng (model/controller), Nhật (UI).
- **Quên mật khẩu:** Nhật/Khôi (email logic), Vương (UI).
- **Học phí:** Hưng (model), Khôi (logic), Vương (UI).
- **Thông báo:** Hưng (model), Khôi (logic), Vương/Nhật (UI).
- **Bảng điều khiển Admin:** Vương (UI), Khôi/Hưng (backend).

Mỗi thành viên nên đảm nhiệm 1–2 tính năng hoàn chỉnh (UI + controller + model), có sự hướng dẫn khi cần.

1.4 Thiết lập Môi trường Phát triển và Hợp tác

Khởi tạo dự án: Trưởng nhóm tạo kho Git và cấu trúc thư mục như trên. Sử dụng môi trường ảo Python và tệp `requirements.txt, pyproject.toml`.

Quy trình Branch: Sử dụng Git flow đơn giản: tạo branch mới cho từng tính năng (vd: `feature-login`, `feature-register`). Commit thường xuyên. Trước khi mở pull request, luôn kéo bản mới nhất từ `main hoặc dev`. Nguyên tắc: "Always be branching, and always be pulling" (*dev.to*).

Làm việc song song: Mỗi người có thể xây dựng phần khung của tính năng (stub) trước, sau đó dần hoàn thiện. Dùng Merge Request để review code. Ghi chú rõ ràng về giao diện giữa các phần (vd: `StudentModel.create_student()` nhận các trường nào).

Kiểm thử và Đồng bộ: Kiểm tra liên tục sau khi hoàn thành từng tính năng. Tổ chức họp ngắn định kỳ (hàng ngày hoặc hàng tuần) để đồng bộ tiến độ. Commit rõ ràng, merge sớm để tránh xung đột.