\*Customer/ student Page

**1. Tùng:**

Trang Home: /home

+ hiện list các khóa học, list category,

+ bấm khóa học thì ra thông tin khóa học, bấm categry thì ra trang courses theo category

**2. Thương:**

trang Courses: /courses/list

+ hiện full list các khóa học

+ lọc theo category, bấm khóa học thì ra thông tin khóa học, bấm categry thì ra trang courses theo category  
+ tìm kiếm theo tên course

**3. thương:**

Course Detail: /courses/detail?id=1

+ hiện thông tin khóa học

+ nút bấm enroll (nếu course mà có phí thì phải hiện form thanh toán, thanh toán xong thì mới được nhảy sang trang lesson)

**4. Vũ:**

Lession: /courses/lesson?id=1

hiện video bài giảng + list các bài học + mỗi bài giảng có 1 đề quiz để kiểm tra

**5. Tùng:**

Enroll: /student/enroll

+ hiện list các course đang/ đã tham gia học

+ in kết quả học tập kèm theo các khóa đó (tiến độ, avg score)

**6. Thương:**

profile: /account/studentprofile

+hiện thông tin người đăng nhập

+CRUD thông tin người dùng, mật khẩu, ....

\*admin page:

-role admin:

**7. Vũ:**

Trang Dashboard: /admin/dashboard

hiện thông tin người đăng ký khóa học mới, ....

**8. Thương:**

User: /admin/UserMana

quản lí user, phân quyền (có thể lock tk người dùng này, thay đổi role của người dùng khác )

**9. Vũ:**

StudentFee: /admin/studentfee

quản lí hóa đơn thanh toán

-role lecturer:

**10. Tùng**

Course: /lecturer/coursemana

+quản lí CRUD khóa học, CRUD lession,

+CRUD đề kiểm tra(upload xử lí file excel các đề: cột 1 là câu hỏi, cột 2345 là đáp án,

cột 6 là đáp án đúng, chuyển nó sang DTO rồi chuyển về json lưu vào db )

-role staff:

**11. Vũ:**

StudentFee: /staff/studentfee

+ quản lí hóa đơn thanh toán

-common:

**12. Tùng**

profile: account/userprofile

hiện thông tin đăng nhập, CRUD thông tin người dùng, mật khẩu, ....

Gợi ý:

Khi pull về thì chạy terminal:

//dotnet tool install --global dotnet-ef

// Cài đặt công cụ dotnet ef

//dotnet ef migrations add NameMigration

// Tạo một Migration có tên NameMigration

//dotnet ef database update

// Cập nhật Database đến Migration cuối

- tài khoản test: admin – 12345678

-Dòng màu vàng là cấu trúc views, ví dụ account/userprofile => Views/Account/userprofile.cshtml

/home => Views/Home/index.cshtml

-Controller để gọi View Controllers/..........

-Phân quyền API/VIEWS: tham khảo Models/RoleEnum.cs để xem tên role và xem file API/AdminController.cs ,nếu API/VIEW public thì không thêm [authorize]

-Tham khảo wwwroot/fetchdata.js để biết gọi API

-Template viết API: TemplateController.cs

-Muốn lấy thông tin người đang đăng nhập, tham khảo authorizeAttribute.cs dòng 31

-Một số chức năng(modal, filter, search) nên tham khảo trong file billing.js và view billing.cshtml và các page khác

