

Module/môn: Lập trình C# 6	Số hiệu assignment:	% điểm: 60%
Người điều phối của FPT Polytechnic: Phan Viết Thế	Ngày ban hành:	
Bài assignment này đòi hỏi sinh viên phải thực hiện trong suốt khóa học		
Tương ứng với mục tiêu môn học: - Tạo và hoàn thiện tài liệu ApiRest - Ứng dụng WebAssembly - Ứng dụng Blazor framework - Các kỹ thuật xử lý tập tin với ApiRestful - QRcode-BARCODE - Unit testing asp core		

Gian lận là hình thức lấy bài làm của người khác và sử dụng như là mình làm ra. Hình thức đó bao gồm những hành động như: copy thông tin trực tiếp từ trang web hay sách mà không ghi rõ nguồn tham khảo trong tài liệu; gửi bài assignment làm chung như là thành quả cá nhân; copy bài assignment của các sinh viên khác cùng khóa hay khác khóa; ăn trộm hay mua bài assignment của ai đó và gửi lên như là sản phẩm mình làm ra. Những sinh viên bị nghi ngờ gian lận sẽ bị điều tra và nếu phát hiện là có gian lận thì sẽ phải chịu các mức phạt theo quy định của Nhà trường.

Mọi tài nguyên copy hay điều chế từ bất cứ nguồn nào (VD: Internet, sách) phải được đặt trong cặp dấu nháy kép và in nghiêng, với thông tin tham khảo đầy đủ về nguồn tài liệu.

Bài làm của bạn sẽ được đưa vào phần mềm kiểm tra gian lận. Mọi hình thức cố tình đánh lừa hệ thống phát hiện gian lận sẽ bị coi là Vi phạm quy định thi cử.

Quy định nộp bài assignment

- Một bản mềm kết quả bài làm assignment của bạn phải được upload trước nửa đêm (giờ địa phương) vào ngày hạn nộp. **Quá hạn nộp hệ thống sẽ khóa lại và sinh viên không còn quyền nộp bài.**
- Phiên bản upload lên cuối cùng sẽ được chấm điểm. Sinh viên có quyền upload đè file nhiều lần trước khi hết hạn nộp.
- Tất cả những file tài liệu văn bản phải để ở dạng file gốc chứ không file dạng file được xuất ra từ định dạng khác (ví dụ pdf được xuất từ doc). Không được gửi tài liệu văn bản dưới dạng ảnh chụp.
- Đối với bài assignment này bạn cũng phải đưa các bằng chứng hay sản phẩm khác vào trong file nén dạng zip.
- Kích thước file cần tuân thủ theo giới hạn trên hệ thống nộp bài (thông thường là <50M).
- Hãy đảm bảo các file được upload lên không bị nhiễm virus (**điều này có thể dẫn đến file**

bị hệ thống xóa mất) và không đặt mật khẩu mở file. Nếu vi phạm những điều này, bài coi như chưa được nộp.

- Hãy chú ý xem thông báo sau khi upload để chắc chắn bài của bạn đã được nộp lên hệ thống chưa.
- Bạn không phải gửi lại file đề bài của assignment (file này).

Quy định đánh giá bài assignment

1. Sinh viên không có bài assignment trên hệ thống sẽ bị 0 điểm bài assignment.
2. Sau hạn nộp bài một tuần, sinh viên nộp muộn có quyền nộp đơn kiến nghị xin được chấp nhận gia hạn nộp. Hội đồng Nhà trường sẽ xét duyệt từng trường hợp. Nếu kiến nghị không được chấp nhận, bài giữ nguyên điểm 0. Nếu quá một tuần không có kiến nghị thì bài cũng sinh viên không nộp mặc nhiên nhận điểm 0.
3. Ngay cả trường hợp bài của sinh viên bị phát hiện gian lận sau khi có điểm, sinh viên sẽ không được công nhận bài đó và chịu mức kỷ luật như quy định của Nhà trường.

=====

Assignment

Lập trình C# 6 – Thiết kế web bán thức ăn nhanh trực tuyến

Mục tiêu cụ thể	Sau bài assignment này , sinh viên cần đạt được: <ul style="list-style-type: none"> - Thiết kế web asp.net core RESTful Api với Swager - Sử dụng .Net Core Blazor, WebAssembly - Xử lý tập tin trong asp.net core RESTful Api - Tạo ứng dụng với QRcode-BARCODE và Unit testing asp.net core
Các công cụ cần có	<ul style="list-style-type: none"> - .Net Core frame work - Visual Studio 2017 trở lên - MS SQL server 2012 hoặc MySQL - Postman - Swagger
Tài nguyên	Thư mục tài nguyên đi kèm

ĐẶT VẤN ĐỀ

Hiện nay các mô hình kinh doanh qua hệ thống e-commerce ngày càng đa dạng. Từ thực tế đó, ứng dụng web tự động hóa quá trình quản lý và bán hàng trực tuyến được thiết kế để phù hợp với mọi cửa hàng nhỏ và vừa ra đời. Một cửa hàng bán thức ăn nhanh trực tuyến giúp mọi người có thể mua được các món ăn, thức uống yêu thích ngay cả khi ngồi ở nhà hoặc nơi làm việc, điều này mang lại sự tiện nghi và dễ dàng tiếp cận cho tất cả khách hàng.

Về phía khách hàng thì Một cửa hàng bán thức ăn nhanh trực tuyến cho phép khách hàng tự kiểm soát thời gian, ghi rõ chi tiết đơn hàng và chủ động mỗi khi có nhu cầu. Khách hàng sẽ không phải mang một cái bụng đói xẹp hàng dài để đặt đồ ăn, thay vào đó có thể thoải mái khám phá các món trên menu và đặt đồ mà không cần thông qua nhân viên.

Nhà hàng cũng có thể đưa ra nhiều hình thức thanh toán qua hệ thống ví điện tử: MoMo, VN Pay, Zalo Pay, v.v... giúp tăng lợi nhuận và giảm chi phí vận hành. Khách hàng vẫn có cơ hội trò chuyện với nhân viên khi mua đồ ăn hoặc yêu cầu vận chuyển. Hơn nữa việc khách hàng đặt đồ ăn online sẽ giúp nhân viên có thêm thời gian để tập trung tạo ra những món ăn ngon, sáng tạo, và thu hút thị giác. Tốc độ phục vụ nhanh hơn và khách hàng không phải mất thời gian chờ đợi. Chỉ với vài thao tác đơn giản trên thiết bị di động, đơn hàng sẽ được chuẩn bị và giao đi bất kỳ lúc nào, bất kỳ nơi đâu.

Về mặt công nghệ, .NET 6 sẽ là nền tảng hợp nhất để xây dựng các ứng dụng chạy trên tất cả các nền tảng (như Windows, Linux) và các thiết bị (như IoT, Mobile). Là một lập trình viên .NET, nhìn về tương lai của công nghệ .NET và hướng đi của nó thì việc trở thành một fullstack là hoàn toàn có thể hiện thực, bạn có thể không còn cần phải học JavaScript để viết ứng dụng web SPA vì có thể dùng Blazor. bạn không còn cần phải học Python cho Machine Learning vì có thể sử dụng ML.NET. Bạn không còn cần phải học Android hay Swift vì Xamarin sẽ làm điều đó. Nếu bạn biết lập trình C#, bạn có thể phát triển ứng dụng có thể chạy mọi nơi từ IoT đến Cloud!

Với các lợi thế như mô tả bên trên, sinh viên cần xây dựng ứng dụng web đáp ứng một các yêu cầu bên dưới

YÊU CẦU

Asm thực hiện cá nhân mỗi sinh viên.

Ứng dụng cần đáp ứng được những chức năng cơ bản sau:

- **Đối với nhóm người dùng Khách viếng thăm (Guest):**
 - o Đăng ký tài khoản trên hệ thống (thông tin đăng ký ít nhất là 6 thông tin).
 - o Duyệt các loại thức ăn nhanh + thực đơn riêng theo combo.
 - o Xem thông tin chi tiết thức ăn nhanh + thực đơn riêng theo combo.
 - o Tìm kiếm cơ bản: Tìm món ăn theo tên
 - o Tìm kiếm nâng cao: Tìm món ăn theo tên, giá cả, chủng loại, thông tin, chủ đề.
- **Đối với nhóm người dùng khách hàng (Customer):**
 - o Đăng nhập hệ thống Site khách hàng + lựa chọn đăng nhập qua Google
 - o Cập nhật thông tin cá nhân của tài khoản
 - o Đặt món ăn và thanh toán
 - o Xem thông tin lịch sử quá trình đặt món ăn qua các hóa đơn
 - o Theo dõi tình trạng các hóa đơn mới mua
- **Đối với nhóm người dùng quản trị (Admin):**
 - o Đăng nhập hệ thống Site quản lý
 - o Cập nhật thông tin cá nhân của tài khoản
 - o Quản lý các tài khoản của người dùng (Danh sách, Thêm, Xóa, Cập nhật). Lưu ý không xóa tài khoản Admin hiện đang sử dụng.
 - o Quản lý các loại món ăn nhanh (,Danh sách, Thêm, Xóa, Cập nhật)
 - o **Mỗi món ăn có một mã QR code**
 - o Quản lý các combo (Danh sách, Thêm, Xóa, Cập nhật)
 - o Quản lý đơn đặt hàng (đã giao, chưa giao, đang giao)

Y1. Thiết kế cơ sở dữ liệu

- Tạo CSDL
- Thiết kế mô hình ERD và các bảng phù hợp đáp ứng các chức năng bên trên
- Nhập dữ liệu sẵn cho các bảng dữ liệu

Y2. Những yêu cầu về hình thức sản phẩm

- Phát triển giao diện client và admin thích hợp, có thể sử dụng các mẫu template có sẵn nhưng phải layout lại phù hợp với yêu cầu Asm
- Template và layout cần đồng nhất cho ứng dụng

Y3. Đáp ứng được các yêu cầu chức năng Đối với nhóm người dùng Khách viếng thăm (Guest)

Y4. Viết RestApi đáp ứng được các yêu cầu chức năng Đối với nhóm người dùng khách hàng (Customer) và Sử dụng Blazor WebAssembly tạo các Single Page Application gọi các RestApi này

Y5. Đáp ứng được các yêu cầu chức năng Đối với nhóm người dùng quản trị (Admin)

- Viết RestApi cho chức năng Quản lý các loại món ăn nhanh (,Danh sách, Thêm, Xóa, Cập nhật), Sử dụng Blazor server cho webpage admin thực hiện các thao tác liên quan chức năng này thông qua RestApi đó

Y6. . Các yêu cầu bảo mật và quản lý

- Sử dụng Dependency Injection
- Sử dụng các kỹ thuật Model Validation, Data Validation, Data Annotation, Fluent API ràng buộc csdl và validation các form
- Bảo mật RestApi với kỹ thuật JWT
- Thực hiện Unit Test cho các chức năng
- Tài liệu RESTful API (Swagger)
- Upload website lên hosting

MÔ TẢ SẢN PHẨM PHẢI NỘP

1. Thư mục BaoCao chứa file BaoCao.doc
2. Thư mục Source chứa toàn bộ ứng dụng
3. Thư mục Database chứa file database.sql chứa mã script tạo CSDL hoặc file .mdf và .ldf của CSDL

Tên của file nén được quy định là <Mã sinh viên>_Mã Môn_Assignment.zip (ví dụ: TeoNVPS10005_SOF...._Assignment.zip). Nộp bài đúng nơi yêu cầu của giảng viên hướng dẫn.

THANG ĐÁNH GIÁ

<div data-bbox="295 1606 386 1703" data-label="Text"> <p>A</p> </div> <div data-bbox="261 1749 422 1782" data-label="Text"> <p>70%-100%)</p> </div>	<ul style="list-style-type: none"> - Đáp ứng tất cả các yêu cầu - Đạt điểm tối đa 100% nếu có thêm chức năng
---	--

<p>B</p> <p>60%- 69%</p>	<ul style="list-style-type: none"> - Thiếu Y6
<p>C</p> <p>50%- 59%</p>	<ul style="list-style-type: none"> - Thiếu Y5, Y6
<p>TRƯỢT</p> <p>< 50% (không đáp ứng yêu cầu nộp bài)</p>	<ul style="list-style-type: none"> - Không nộp đủ các sản phẩm được mô tả trong mục Sản phẩm phải nộp - Ứng dụng không chạy