ĐỒ ÁN CUỐI KÌ MÔN PHÁT TRIỂN ỨNG DỤNG WEB

*Food Order*

# **Thông tin nhóm thực hiện**

## Thông tin cơ bản

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STT | MSSV | Họ tên | | Email | | | Điện thoại | Điểm đề nghị |
| 1 | 1312009 | | Nguyễn Tuấn Anh | | tuananhpro2981995@gmail.com | 0981103589 | |  |
| 2 | 1312223 | | Nguyễn Thị Hòa | | nthoa9195@gmail.com | 0965205453 | |  |
| 3 | 1312494 | | Nguyễn Thái Sơn | | thaisonnguyen8295@gmail.com | 01693260701 | |  |

## Phân chia công việc

### *1312009 – Nguyễn Tuấn Anh*

* Cho phép người dùng đăng ký tài khoản hoặc kết nối qua các mạng xã hội Google+, Facebook bảm mật theo jwt, password được mã hóa…
* Viết các api kiểm tra các ràng buộc về tên đăng nhập, mật khẩu nhập lại, ... khi cho khách hàng đăng ký tài khoản mới.
* Viết các api đăng nhập hệ thống
* Viết api quản lí các role ngăn cấm người chưa đăng nhập sử dụng các chức năng bắt buộc đăng nhập theo quyền hạn
* Viết một số api liên quan đến việc mua hàng, đặt hàng, thanh toán của khách hàng cũng như các api liên quan đến quyền admin như thêm, xóa, sửa các món ăn, quản lí các đơn đặt hàng đồng thời quản lí các tài khoản có trên hệ phống. Phân quyền admin….

### *1312223 – Nguyễn Thị Hòa*

* Tổ chức tốt cấu trúc của CSDL
* Cập nhập CSDL
* Viết báo cáo
* Viết một số api liên quan đến việc load thông tin món ăn thể hiện món ăn, giá, rating, mô tả… lên trang web.
* Phân tách và tổ chức hợp lý, cụ thể theo từng cấp Module, Controller, Service

### *1312494 – Nguyễn Thái Sơn*

* Thiết kế giao diện.
* Angular: phân tách, xử lý Component; thực hiện Animations; kiểm tra các ràng buộc và thông báo ứng với các chức năng có yêu cầu nhập liệu.
* Quản lý các trạng thái khi sử dụng liên tiếp các tính năng trên trang web.
* Đăng tải website lên host thực tế.
* Tạo cấu trúc Project, hộ trợ viết một số api liên quan đến admin.
* Liên kết server với client.

# **Thông tin chung về đồ án**

* Mục tiêu đồ án cuối kỳ nhóm muốn xây dựng một hệ thống “FOOD ORDER” để có thể hỗ trợ việc gọi thức ăn tại bàn hoặc việc giao cơm văn phòng của bất kỳ cửa tiệm nào trong hệ thống. Việc này sẽ rất tiết kiệm về thời gian và sẽ không bị nhầm lẫn trong quá trình order thức ăn. Bạn không phải đứng chờ chen lấn hay xếp hàng chỉ để gọi món, không phải chờ đợi mãi mới có thể order được món , hay cảm giác không hài lòng khi phục vụ đem sai món, hay những rắc rối khi phải thối tiền,… “FOOD ORDER” sẽ giải quyết được mọi thứ khi bạn cho cúng tôi biết bạn là ai (yêu cầu một tài khoản uy tín) và bạn muốn ăn món gì ở đâu.

# **Thuyết minh nội dung**

## Phát biểu vấn đề

* Xuất phát từ nhu cầu gọi cơm văn phòng tại chỗ mà không cần tới cửa hàng. Nhóm xây dựng một hệ thống “Đặt cơm văn phòng” để có thể hỗ trợ việc gọi thức ăn tại bàn hoặc việc giao cơm văn phòng của một cửa hàng quán ăn. Với tiêu chí “Nhanh chóng – Tiện lợi-  Uy tín”, Đặt cơm văn phòng sẽ cung cấp cho người dùng một hệ thống gọi cơm tốt nhất và tiện lợi nhất.

## Mục tiêu đồ án

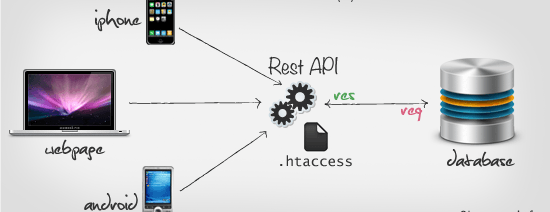
* Xây dựng thành công hệ thống đặt cơm văn phòng với các tiêu chí sau:
* Giao diện: Tiện dụng, dễ dàng theo tác, màu sắc hài hoà và có khả năng hỗ trợ đặt món ăn tốt.
* Xử lí: Chính xác, nhanh. Hỗ trợ đăng nhập từ các mạng xã hội với sự bảo mật thông tin tốt nhất được hỗ trợ từ Wep api của server
* Dữ liệu: Dùng SQL Server để lưu toàn bộ dữ liệu cho toàn bộ hệ thống.

## Ý nghĩa thực hiện

* Hệ thống này có ý nghĩa rất lớn đối với những nhân viên văn phòng không có nhiều thời gian rảnh. Điều này giúp họ tiết kiệm tối đa thời gian và không phải lo lắng việc chờ đợi gọi món tại cửa hàng. Chỉ cần vài cái click chuột, 30p sau sẽ có thức ăn được giao tận nơi.

# **Nội dung triển khai:**

* Kiến trúc đồ án:



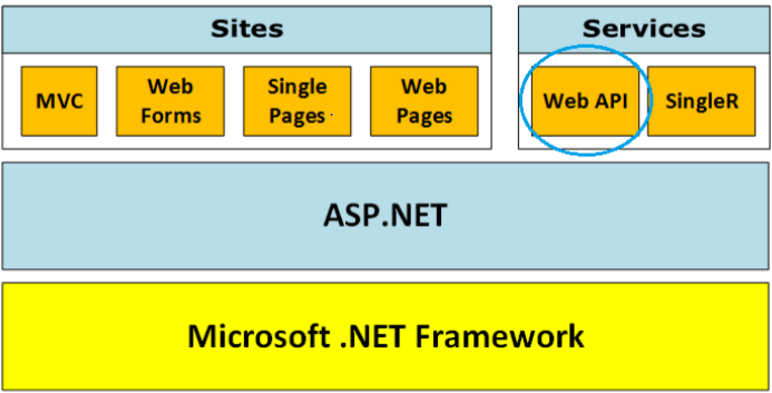
* **Đồ án sử dụng kiến trúc Database - Wep API – Client**

Trên nền tảng Net 4.0 trở lên có xuất hiện một công nghệ mới đó là tạo Webservices bằng Web Api. Web giúp xây dựng Rest full services trên nền tảng .net, nó tự động trả dữ liệu có kiều phù hợp: Json, XML. Web Api sẽ giúp bạn chuyển kiểu dữ liệu trả về theo thuộc tính Accept trong header của Request và bạn không cần quan tâm đến điều đó.

Web Api được map với các action dựa trên phương thức: GET, PUT, POST, DELETE.

Web Api là một kiến trúc gọn nhẹ ngoài việc sử dụng cho các ứng dụng Web, nó còn dược dùng cho các ứng dụng Mobile. Nó sẽ thao tác trên dữ liệu database và trả data cho client.

* Công nghệ sử dụng:



* Sử dụng WEB API:
* **Đặc điểm Web API (.NET 4.0 trở lên)**
* Giúp cho việc xây dựng các HTTP service rất đơn giản, nhanh chóng
* Mã nguồn mở (Open Source) và có thể được sử dụng bởi bất kì client nào hỗ trợ XML, JSON.
* Hỗ trợ đầy đủ các thành phần HTTP: URI, request/response headers, caching, versioning, content formats.
* Có thể host trong ứng dụng hoặc trên IIS.
* Kiến trúc lý tưởng cho các thiết bị có băng thông giới hạn như các thiết bị di động.
* Định dạng dữ liệu có thể là JSON, XML hoặc một kiểu dữ liệu bất kỳ.
* Làm mới và hiện đại hóa các mẫu dự án mặc định
* Mẫu dự án trên điện thoại di động
* Nhiều tính năng mới để hỗ trợ các ứng dụng di động
* Tùy chỉnh sinh mã(code).
* Tăng cường hỗ trợ cho các phương pháp bất đồng bộ
* Đọc danh sách đầy đủ tính năng trong các ghi chú phát hành
* **Ưu điểm của Web API**
* Cấu hình đơn giản hơn nhiều so với WCF.
* Hiệu suất(performance) cao.
* Hỗ trợ RESTfull đầy đủ.
* Hỗ trợ đầy đủ các thành phần MVC như: routing, controller, action result, filter, model binder, IoC container, dependency injection, unit test, …
* Mã nguồn mở (Open source).
* **Giới thiệu cơ bản về RESTful Service**
* Là một dịch vụ web đơn giản sử dụng HTTP và tính chất của REST.
* Nó tuân thủ theo 4 nguyên tắc thiết kế cơ bản sau:
* Sử dụng các phương thức HTTP một cách rõ ràng
* Phi trạng thái
* Hiển thị cấu trúc thư mục như URls
* Chuyển đổi linh hoạt JavaScript Object Notation (JSON) và XML hoặc cả hai.
* Sử dụng Angularjs
* AngularJS là một Framwork phát triển dựa trên Javascript để tạo các ứng dụng web phong phú
* AngularJS thường dùng để phát triển frontend (giao diện khách hàng) thông qua các API để gọi data, sử dụng mô hình MVC rất mạnh mẽ
* Mã nguồn AngularJS tự động fix với các trình duyệt khác nhau nên bạn không cần phải lo vấn đề tương thích trình duyệt
* Angular là mã nguồn mở, hoàn toàn miễn phí và được phát triển bởi hàng ngàn các lập trình viên trên thế giới.

Chung quy lại có thể hiểu khi làm việc với AngularJS giống như là đang làm việc với Ajax, sử dụng cớ chế bind data, hoạt động theo mô hình MVC và sử dụng service để tương tác với dữ liệu từ server.

* Sử dụng SQL Server
* Với SQL, chúng ta có thể truy vấn Database theo nhiều cách khác nhau, bởi sử dụng các lệnh.
* Với SQL, người dùng có thể truy cập dữ liệu từ RDBMS.
* SQL cho phép người dùng miêu tả dữ liệu.
* SQL cho phép người dùng định nghĩa dữ liệu trong một Database và thao tác nó khi cần thiết.
* Cho phép người dùng tạo, xóa Database và bảng.
* Cho phép người dùng tạo view, Procedure, hàm trong một Database.
* Cho phép người dùng thiết lập quyền truy cập vào bảng, thủ tục và view.

# **Dự kiến kết quả**

* Hệ thống sẽ đạt được những tiêu chí đưa ra về giao diện, xử lí và tính trải nghiệm người dùng.  Vấn đề bảo mật cũng sẽ được thực hiện tốt bằng cách sử dụng Microsoft Owin, JWT, Pass sercutrity kết hợp với các rule viết từ server cũng như client.