

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CMC**



**CMC UNIVERSITY**  
Aspire to Inspire the Digital World

**BÀI TẬP CÔNG NGHỆ LẬP TRÌNH WEB**

Tên đề tài: **Kết nối Power BI REST API từ 1 Service**

**Nhóm sinh viên: Nguyễn Bá Minh Quang: BCS230073**

**Đỗ Phúc Hưng: BCS230042**

**Giảng viên: Lưu Thị Bích Hương**

Hà Nội, ngày 19 tháng 2 năm 2025

## MỤC LỤC

<b>BỘ GIÁO DỤC VÀ ĐÀO TẠO.....</b>	<b>1</b>
<b>TRƯỜNG ĐẠI HỌC CMC.....</b>	<b>1</b>
<b>Lời nói đầu.....</b>	<b>3</b>
1. Giới thiệu dự án.....	4
2. Cấu trúc dự án.....	5
3. Chức năng chính.....	6
Xác thực người dùng.....	6
Tích hợp Power BI.....	6
Quản lý tài khoản.....	6
Tạo và quản lý báo cáo.....	6
Hỗ trợ biểu đồ tương tác.....	7
4. Triển khai và cài đặt.....	8
4.1 Yêu cầu hệ thống.....	8
4.2 Cài đặt và triển khai.....	8
4.3 Mở rộng dự án trong tương lai.....	8
5. Công nghệ sử dụng.....	9
6. Thảo luận và kiến nghị.....	10
6.1 Đánh giá tổng quan bài làm.....	10
6.2 Những hạn chế và khó khăn trong quá trình làm.....	10
6.3 Đề xuất hướng cải tiến.....	11

# Lời nói đầu

Trong thời đại số hóa hiện nay, việc quản lý và phân tích dữ liệu đóng vai trò quan trọng trong việc ra quyết định của các tổ chức và doanh nghiệp. Power BI là một công cụ mạnh mẽ giúp trực quan hóa dữ liệu một cách hiệu quả. Tuy nhiên, để tích hợp và hiển thị dữ liệu từ Power BI vào một ứng dụng web vẫn còn là một thách thức đối với nhiều nhà phát triển

Dự án **PowerBILocalApp** ra đời nhằm giải quyết vấn đề này, giúp người dùng dễ dàng tích hợp báo cáo từ Power BI vào nền tảng web, tạo ra giao diện trực quan, thân thiện và dễ sử dụng. Báo cáo này sẽ trình bày về các công nghệ sử dụng, cấu trúc dự án, chức năng chính cũng như cách triển khai ứng dụng.

# 1. Giới thiệu dự án

Dự án **PowerBILocalApp** là một ứng dụng web được xây dựng bằng **ASP.NET Core MVC**, giúp quản lý và hiển thị dữ liệu từ Power BI trên nền tảng web. Mục tiêu của dự án là tích hợp dữ liệu từ Power BI để cung cấp báo cáo trực quan, hỗ trợ người dùng trong việc phân tích dữ liệu

Hiện nay, Power BI đã trở thành một công cụ phổ biến để trực quan hóa và phân tích dữ liệu, giúp các tổ chức dễ dàng theo dõi hiệu suất và ra quyết định dựa trên dữ liệu. Tuy nhiên, việc tích hợp Power BI vào hệ thống web vẫn còn gặp nhiều khó khăn, đặc biệt là khi làm việc với Power BI REST API và vấn đề bảo mật dữ liệu

Dự án **PowerBILocalApp** được phát triển nhằm khắc phục những khó khăn trên, cung cấp một giải pháp giúp doanh nghiệp hoặc cá nhân có thể truy cập và hiển thị báo cáo Power BI một cách dễ dàng mà không cần sử dụng giao diện mặc định của Power BI. Với kiến trúc linh hoạt, ứng dụng này có thể mở rộng để phù hợp với nhiều loại hình doanh nghiệp và nhu cầu cụ thể của từng tổ chức

Ngoài ra, **PowerBILocalApp** không chỉ giúp hiển thị dữ liệu mà còn cung cấp khả năng tương tác với báo cáo theo nhu cầu của người dùng, cho phép lọc dữ liệu, xem thông tin chi tiết, đồng thời đảm bảo tính bảo mật thông qua hệ thống xác thực người dùng

Trong báo cáo này, chúng ta sẽ đi sâu vào các công nghệ được sử dụng, kiến trúc của ứng dụng, các chức năng chính và cách triển khai ứng dụng để đảm bảo hoạt động hiệu quả.

## 2. Cấu trúc dự án

Dự án **PowerBILocalApp** được thiết kế theo mô hình MVC (Model-View-Controller), giúp tách biệt các thành phần để dễ dàng phát triển và bảo trì. Cấu trúc thư mục chính bao gồm:

- **Controllers/** – Chứa các bộ điều khiển xử lý logic ứng dụng, nhận yêu cầu từ người dùng và gọi các *service* để xử lý dữ liệu trước khi trả về *view*
- **Models/** – Định nghĩa các lớp dữ liệu, đại diện cho các đối tượng được sử dụng trong ứng dụng, giúp quản lý dữ liệu từ cơ sở dữ liệu một cách có hệ thống
- **Views/** – Chứa giao diện người dùng, hiển thị dữ liệu và giao tiếp với *controller* thông qua các mô hình (*Model*)
- **Services/** – Xử lý logic nghiệp vụ, giao tiếp với **Power BI REST API** để truy xuất và cập nhật dữ liệu từ Power BI
- **wwwroot/** – Chứa các tệp tĩnh như *CSS*, *JavaScript*, hình ảnh, giúp thiết kế và cải thiện giao diện người dùng
- **appsettings.json** – Chứa các cấu hình ứng dụng như kết nối cơ sở dữ liệu, thông tin xác thực Power BI và các thiết lập hệ thống khác
- **Program.cs** – Tập tin khởi động ứng dụng, nơi cấu hình các dịch vụ và middleware của ứng dụng

Nhờ kiến trúc này, dự án có thể dễ dàng mở rộng và bảo trì khi có yêu cầu phát triển thêm tính năng mới hoặc tích hợp các công nghệ khác

### 3. Chức năng chính

Dự án **PowerBILocalApp** cung cấp các chức năng quan trọng nhằm hỗ trợ người dùng trong việc tích hợp, hiển thị và quản lý dữ liệu từ Power BI một cách hiệu quả. Các chức năng chính bao gồm:

#### Xác thực người dùng

- Người dùng có thể **đăng ký** tài khoản, **đăng nhập** vào hệ thống bằng email hoặc tài khoản Microsoft và **đăng xuất** tài khoản
- Hỗ trợ **xác thực OAuth2.0**, giúp kết nối với tài khoản Power BI một cách bảo mật
- Cơ chế **quản lý phiên đăng nhập**, giúp duy trì trạng thái đăng nhập và ngăn chặn truy cập trái phép
- Chức năng **quên mật khẩu**, cho phép người dùng lấy lại mật khẩu thông qua email

#### Tích hợp Power BI

- Ứng dụng hỗ trợ hiển thị báo cáo từ **Power BI REST API**, giúp người dùng có thể truy cập dữ liệu theo thời gian thực
- Hỗ trợ **nhúng báo cáo Power BI vào trang web** mà không cần mở giao diện Power BI
- Cho phép **lựa chọn** và **thay đổi báo cáo** từ danh sách các báo cáo có sẵn trên tài khoản Power BI của người dùng
- Hỗ trợ **tải báo cáo nhanh**, đảm bảo hiệu suất tối ưu khi truy xuất dữ liệu lớn

#### Quản lý tài khoản

- Người dùng có thể xem và chỉnh sửa **thông tin cá nhân**, bao gồm tên, email và ảnh đại diện
- Hỗ trợ **thay đổi mật khẩu**, đảm bảo tính bảo mật cho tài khoản
- Cho phép người dùng **liên kết hoặc ngắt kết nối tài khoản Power BI**, giúp quản lý quyền truy cập dữ liệu linh hoạt hơn

#### Tạo và quản lý báo cáo

- Người dùng có thể xem danh sách tất cả các báo cáo hiện có trong hệ thống
- Hỗ trợ **lọc và tìm kiếm báo cáo**, giúp dễ dàng truy xuất thông tin cần thiết

- Tích hợp tính năng **truy vấn dữ liệu theo nhu cầu**, cho phép người dùng nhập điều kiện lọc để hiển thị dữ liệu theo mong muốn
- Hỗ trợ **xuất báo cáo** sang các định dạng như **PDF, Excel, CSV**, giúp người dùng lưu trữ và chia sẻ dữ liệu một cách thuận tiện

### **Hỗ trợ biểu đồ tương tác**

- Cung cấp **các biểu đồ trực quan** như cột, tròn, đường, heatmap, giúp phân tích dữ liệu dễ dàng hơn
- Hỗ trợ **zoom** và **drill-down** để xem chi tiết các điểm dữ liệu trên biểu đồ
- Tích hợp **chức năng kéo-thả (drag & drop)**, giúp người dùng có thể tùy chỉnh cách hiển thị dữ liệu theo nhu cầu
- Cho phép **tùy chỉnh bộ lọc dữ liệu** ngay trên giao diện mà không cần cập nhật lại toàn bộ trang

## 4. Triển khai và cài đặt

### 4.1 Yêu cầu hệ thống

**.NET SDK** (phiên bản tương thích với ASP.NET Core)

**MySQL Server** để lưu trữ dữ liệu

**Power BI Account** để tích hợp báo cáo

**Visual Studio 2022** hoặc **VS Code** để phát triển và chạy dự án

### 4.2 Cài đặt và triển khai

1. **Clone dự án:** `git clone <repository_url>`
2. **Cấu hình Power BI:**
  - Đăng ký ứng dụng trên *Microsoft Azure Portal* để lấy *Client ID* và *Secret*
  - Cập nhật thông tin xác thực vào *appsettings.json*
3. **Chạy ứng dụng:**
  - Mở terminal và điều hướng đến thư mục dự án
  - Chạy lệnh *dotnet run* để khởi chạy ứng dụng
  - Truy cập *http://localhost:5000* để sử dụng
4. **Triển khai trên server:**
  - Sử dụng **IIS**, **Azure App Service** hoặc **Docker** để triển khai ứng dụng trên môi trường production
  - Cấu hình domain và SSL nếu cần thiết (mở rộng)

### 4.3 Mở rộng dự án trong tương lai

Dự án có thể phát triển mở rộng hơn với nhiều tính năng trong tương lai như:

- **Hỗ trợ nhiều nguồn dữ liệu hơn:** Mở rộng khả năng kết nối với các nguồn dữ liệu khác như *Google Analytics*, *PostgreSQL*, hoặc *NoSQL*
- **Tích hợp Machine Learning:** Áp dụng AI để dự đoán xu hướng dữ liệu dựa trên báo cáo Power BI
- **Cải thiện giao diện người dùng:** Nâng cấp UI/UX để tương tác tốt hơn với các dashboard
- **Tối ưu hóa hiệu suất:** Cải tiến caching, load balancing để tăng tốc độ xử lý dữ liệu
- **Tích hợp Mobile App:** Cho phép người dùng truy cập báo cáo Power BI trên ứng dụng di động



## 5. Công nghệ sử dụng

Dự án sử dụng các công nghệ chính sau:

**ASP.NET Core MVC** – Framework phát triển ứng dụng web

**C#** – Ngôn ngữ lập trình chính

**Entity Framework Core** – ORM để làm việc với cơ sở dữ liệu

**MySQL** – Cơ sở dữ liệu chính

**Power BI REST API** – Để tích hợp và hiển thị dữ liệu từ Power BI

**Bootstrap, HTML, CSS, JavaScript** – Xây dựng giao diện người dùng

**ChatGPT** – Xây dựng code

## 6. Thảo luận và kiến nghị

### 6.1 Đánh giá tổng quan bài làm

Dự án **PowerBILocalApp** đã hoàn thành các mục tiêu đề ra, bao gồm việc tích hợp dữ liệu từ Power BI vào ứng dụng web thông qua Power BI REST API. Ứng dụng đã triển khai thành công các tính năng quan trọng như xác thực người dùng, quản lý báo cáo và trực quan hóa dữ liệu

Giao diện người dùng được thiết kế trực quan, giúp người dùng dễ dàng tương tác với báo cáo. Ngoài ra, hệ thống sử dụng kiến trúc MVC giúp tổ chức code một cách rõ ràng, dễ dàng bảo trì và mở rộng trong tương lai

### 6.2 Những hạn chế và khó khăn trong quá trình làm

Trong quá trình phát triển dự án, một số khó khăn đã được ghi nhận:

- **Tích hợp Power BI REST API:** Việc thiết lập xác thực và lấy dữ liệu từ Power BI yêu cầu cấu hình trên *Azure Portal*, điều này có thể gây khó khăn đối với những người chưa có kinh nghiệm làm việc với API của Microsoft
- **Quyền truy cập:** Việc kiểm soát quyền truy cập vào service *Azure Portal* khiến cho người mới không có group sẽ rất khó có được **App Registrations**. Hơn nữa báo cáo Power BI yêu cầu phải thiết lập đúng trên Azure và trong ứng dụng, nếu không có thể dẫn đến lỗi khi tải dữ liệu
- **Tối ưu hóa hiệu suất:** Khi xử lý dữ liệu lớn, việc hiển thị báo cáo có thể gặp độ trễ. Cần áp dụng caching hoặc tối ưu query để cải thiện hiệu suất
- **Cấu hình môi trường:** Việc triển khai trên môi trường thực tế đòi hỏi cấu hình phù hợp với server, cơ sở dữ liệu và domain, điều này đôi khi gây mất thời gian

## 6.3 Đề xuất hướng cải tiến

Dựa trên những hạn chế trên, một số cải tiến có thể được thực hiện trong tương lai:

- **Cải thiện hiệu suất tải báo cáo:** Áp dụng caching dữ liệu và tối ưu hóa truy vấn API để giảm thời gian tải báo cáo từ Power BI
- **Bổ sung tính năng tương tác:** Cho phép người dùng tùy chỉnh bộ lọc trên báo cáo ngay trong ứng dụng thay vì chỉ xem dữ liệu cố định, hỗ trợ những người mới vào nhóm cộng đồng để hướng dẫn và cấp quyền sử dụng *Azure*
- **Hỗ trợ nhiều nguồn dữ liệu hơn:** Tích hợp thêm các nguồn dữ liệu như *Google Analytics*, *PostgreSQL* để mở rộng phạm vi phân tích
- **Nâng cấp bảo mật:** Áp dụng *OAuth 2.0* và các cơ chế mã hóa mạnh hơn để bảo vệ dữ liệu và thông tin người dùng
- **Triển khai trên nền tảng đám mây:** Tích hợp với *Azure*, *AWS* hoặc *Google Cloud* để đảm bảo khả năng mở rộng và hiệu suất cao