คู่มือการติดตั้งระบบ

โปรแกรมที่ต้องการ

* Visual Studio Community
* PostgreSQL

ไฟล์ที่ต้องการ

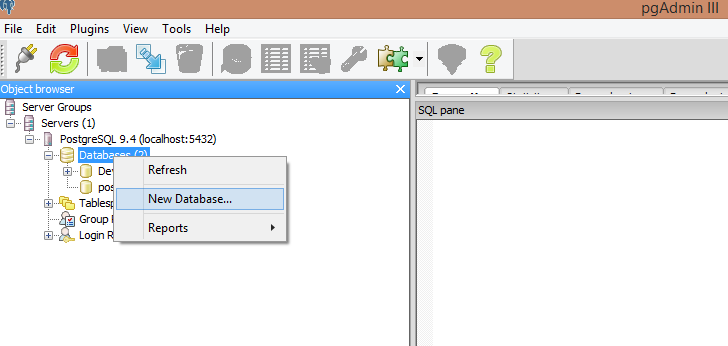
* ไฟล์ Database (Dev.backup)

ติดตั้งโปรแกรม

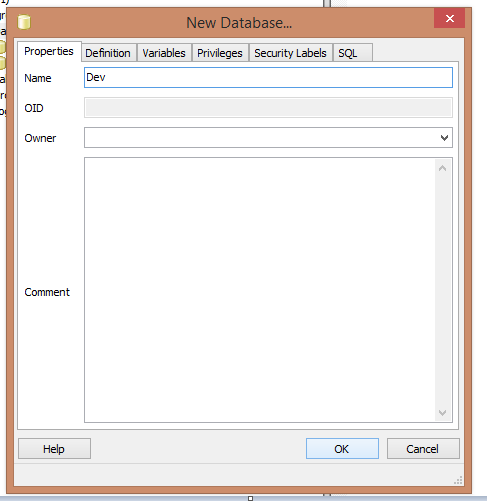
1. ติดตั้งโปรแกรม Visual Studio Community
2. ติดตั้งโปรแกรม PostgreSQL
3. Restore ไฟล์ Database ในโปรแกรม PostgreSQL

วิธีการ Restore Database

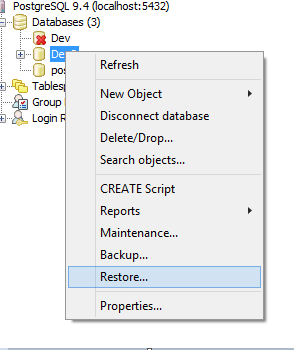
1.สร้าง Database



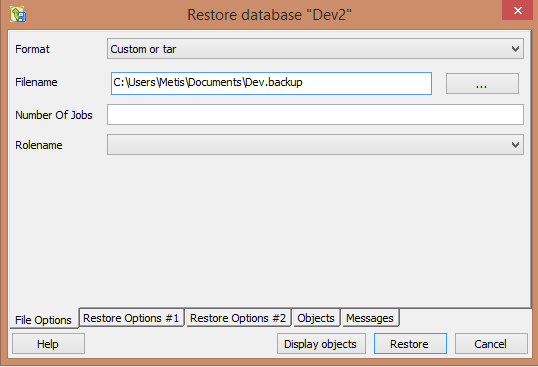
2.ตั้งชื่อ Database ว่า Dev



3.คลิกขวาที่ Database เลือก Restore

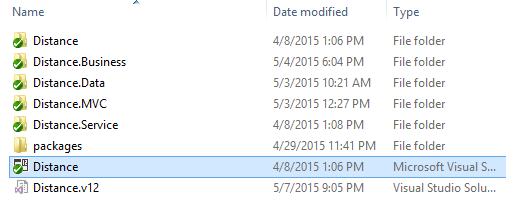


4.เลือกที่อยู่ไฟล์ Backup Database จากนั้นกด Restore

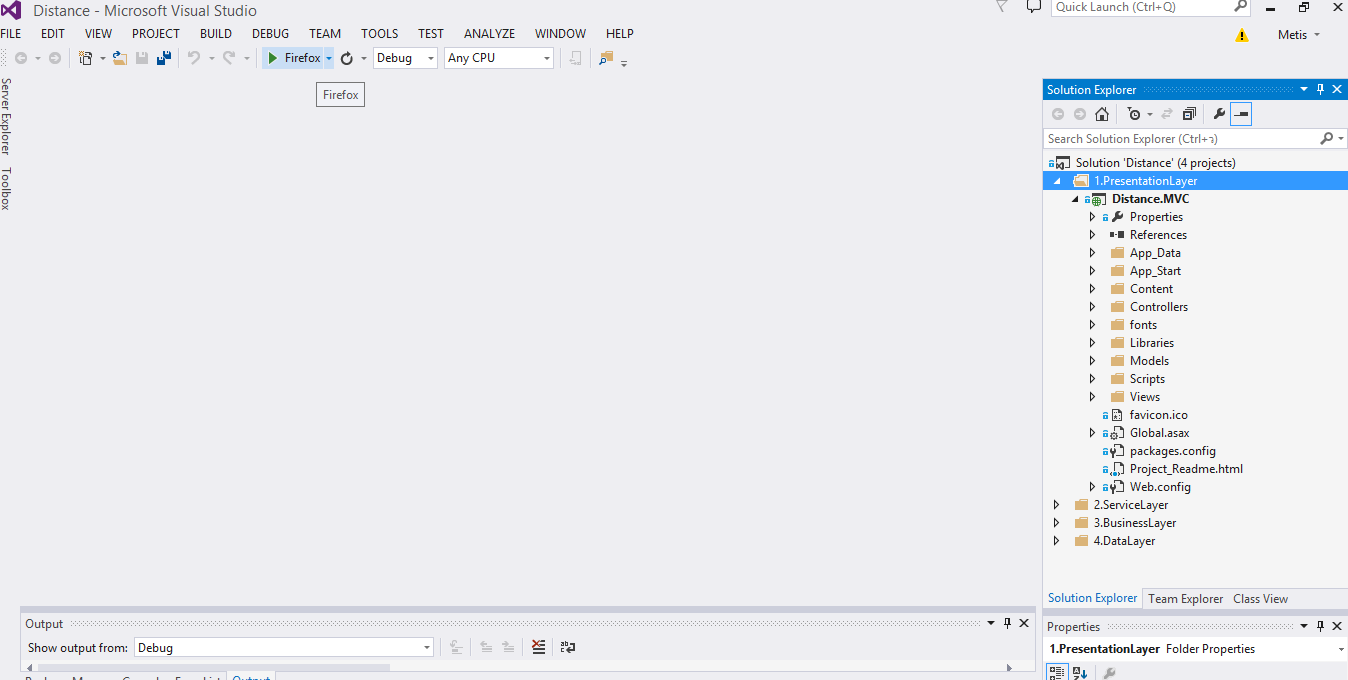


การเข้าใช้งานโปรแกรม

1.เปิดไฟล์ Distance



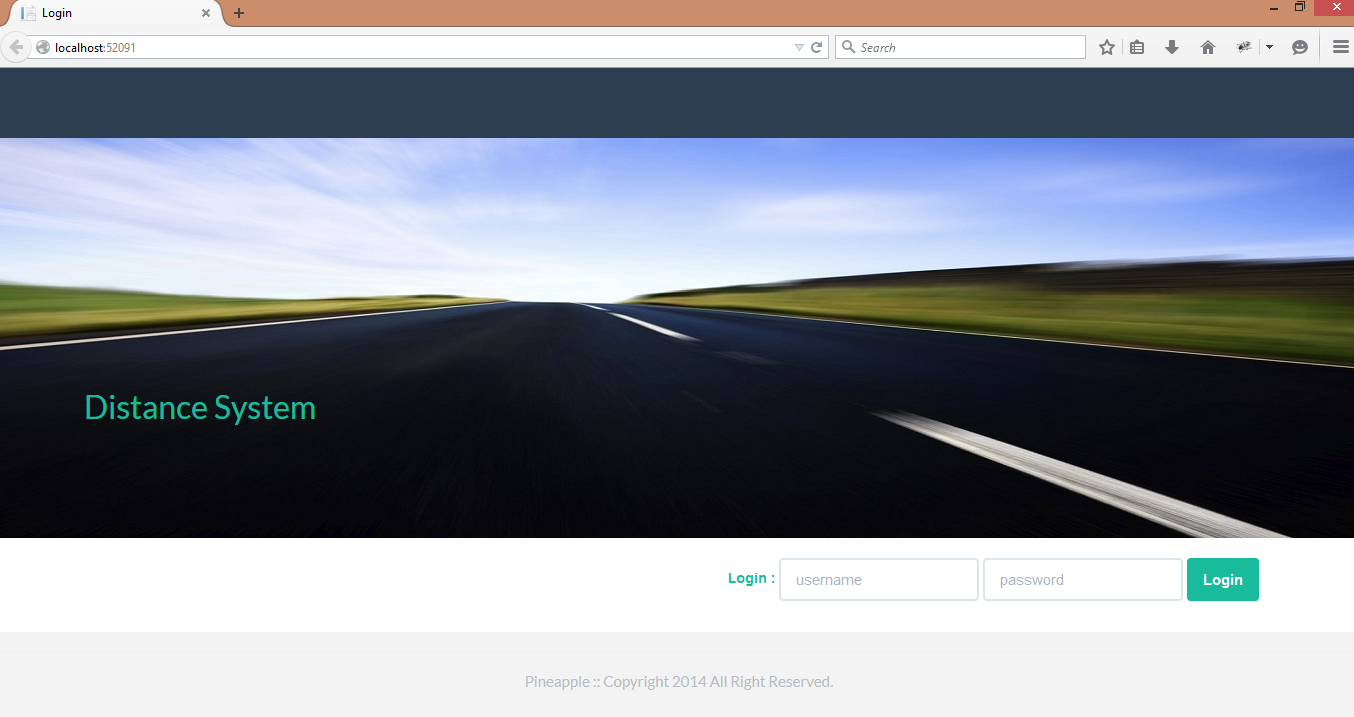
2.กดเริ่มโปรแกรม



3.ใส่ Username กับ Password

ผู้ดูแลระบบ  
- Username :admin  
- Password :123456

ผู้ใช้งานทั่วไป  
- Username : paspond  
- Password : 123456



ปล.ถ้าเกิดกด run แล้วมี error ให้ทำการลบ URL เหลือแค่ <http://localhost:xxxxx/>

โครงสร้างโปรแกรม

ได้ทำการเขียนโปรแกรมโดยใช้ Visual studio community โดยรูปแบบการเรียนเป็น C# .Net MVC

โดยแบ่งออกเป็น 3 บทบาทด้วยกันคือ

Model (M)

Model คือออบเจกต์ที่ทำหน้าเป็นตัวแทนของข้อมูล ไม่ว่าข้อมูลจะถูกจัดเก็บในรูปแบบใดในระบบฐานข้อมูลหรือในไฟล์ เมื่อข้อมูลนั้นถูกโหลดเข้ามาในแอพพลิเคชัน เราจะเปลี่ยนมันให้อยู่ในรูปของออบเจกต์ และเราเรียกบทบาทของออบเจกต์นี้ว่า "Model" ยกตัวอย่างเช่นออบเจกต์ Customer, Employee, Product เป็นต้น

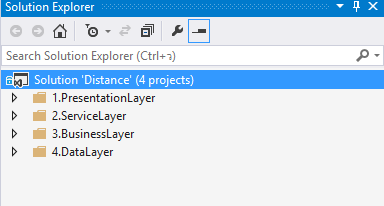
View (V)

View คือออบเจกต์ที่ทำหน้าที่ในการแสดงผล เช่น แสดงหน้าฟอร์มกรอกข้อมูล, แสดงผลลัพธ์ของการค้นหา เป็นต้น ซึ่งมันจะประกอบไปด้วย Presentation Logic เป็นหลัก อย่าลืมว่าการแสดงผลก็มีโลจิกของมันอยู่ อาทิเช่น การเรียงลำดับสินค้าตามราคา เป็นต้น อันที่จริงแล้ว View ก็คือ User Interface (UI) นั่นเอง

Controller (C)

Controller คือออบเจกต์ที่ทำหน้าที่รับคำสั่ง เช่น คำสั่งค้นหา, คำสั่งลบข้อมูล เป็นต้น เมื่อมีคำสั่ง (หรือ Request) เข้ามาจากผู้ใช้ (User) ออบเจกต์ตัวนี้จะทำหน้าที่รับคำสั่งและเรียกใช้ออบเจกต์ตัวอื่นๆ (M และ V) ให้ทำงานร่วมกัน ดังนั้นมันจึงประกอบไปด้วย Control Logic ยกตัวอย่างหน้าที่ของ Controller เช่น  
- ดึงข้อมูลที่ถูกส่งมาจากผู้ใช้ เช่น มาจากการกรอกฟอร์ม  
- ทำการตรวจสอบ (Server-side Validation) ข้อมูลว่าครบถ้วนถูกต้องก่อนนำไปประมวลผลหรือไม่  
- เรียกใช้ Service เพื่อทำการประมวลผลคำสั่งพร้อมข้อมูล  
- ส่งต่อ (Forward) ผลลัพธ์จากการประมวลผลไปยัง View เพื่อแสดงต่อผู้ใช้ (เพราะการแสดงผลไม่ใช่หน้าที่ของ Controller)

ในส่วนของโปรแกรมได้แบ่งออกเป็น 4 เลเยอร์



Presentation

คือส่วนที่ติดต่อกับผู้ใช้งาน

Service

คือส่วนที่คอยทำหน้าที่ติดต่อ Presentation กับ Data

Business

คือส่วนที่คอยกำหนดทิศทางของข้อมูล

Data

คือส่วนให้การเข้าถึงข้อมูล โดยให้ Business มารับรองข้อมูลส่งไปยัง Service เพื่อไปติดต่อกับส่วน Presentation