

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

การปฏิบัติงานด้านสารสนเทศภายในบริษัท ฟรีวิลล์โซลูชั่นส์ จำกัด ทำให้เข้าใจถึงหลักการการทำงานของเว็บแอปพลิเคชันและสามารถนำมาใช้ประโยชน์ในการพัฒนาโครงการ โดยในโครงการได้นำทฤษฎีเอ็มวีซีมาใช้งาน ซึ่งเป็นรูปแบบการพัฒนาเว็บแอปพลิเคชันที่ได้รับความนิยมในปัจจุบัน เนื่องจากภาษาที่ใช้ในการเขียนคือ ซีชาร์ปซึ่งเป็นภาษาที่มีเครื่องมือในการพัฒนาเช่น วิวอลสตูดิโอ ซึ่งสามารถทำขั้นตอน ดิบค เพื่อให้สามารถสร้างลำดับการทำงานและระบุตำแหน่งที่เกิดปัญหาขึ้นในระหว่างการพัฒนาได้ รวมถึงการทำงานร่วมกับภาษาที่ถูกใช้ในการพัฒนาเว็บ ไซต้อย่าง เอชทีเอ็มแอล จาวาสคริป เรเซอร์ซินแทก ได้อย่างดีเยี่ยมและสุดท้ายการใช้งาน ลินคิว เอนดีตีเฟรมเวิร์ค เอสคิวแอลเซิร์ฟเวอร์ เพื่อจัดการในส่วนของฐานข้อมูลที่ใช้ในโครงการ

2.1 เว็บแอปพลิเคชัน (Web Application)

เว็บแอปพลิเคชัน (Web Application) [1] หรือ โปรแกรมประยุกต์บนเว็บ เป็น โปรแกรมประยุกต์ที่เข้าถึงด้วย โปรแกรมค้นหาเว็บผ่านเครือข่ายคอมพิวเตอร์อย่างอินเทอร์เน็ตหรือ อินทราเน็ต เว็บแอปพลิเคชันเป็นที่นิยมเนื่องจากความสามารถในการอัปเดตและดูแล ผู้ใช้งานสามารถเข้าถึงโปรแกรมได้โดยตรงจากเครื่องผู้ใช้ ตัวอย่างเว็บแอปพลิเคชัน ได้แก่ เว็บเมล การพาณิชย์อิเล็กทรอนิกส์ การประมูลออนไลน์ กระดานสนทนา บล็อก วิกี เป็นต้น

ลักษณะเว็บแอปพลิเคชันแบบเบื้องต้น ฟังก์ชันเซิร์ฟเวอร์จะประกอบไปด้วยเว็บเซิร์ฟเวอร์ ทำหน้าที่เชื่อมต่อกับไคลเอนต์ตาม โพรโตคอลเอชทีทีพี/เอชทีทีพีเอส (HTTP/HTTPS) โดยนอกจากเว็บเซิร์ฟเวอร์จะทำหน้าที่ส่งไฟล์ที่เกี่ยวข้องกับการแสดงผลตามมาตรฐานเอชทีทีพีตามปกติทั่วไป เว็บเซิร์ฟเวอร์มีส่วนประมวลผลอาจจะเป็นตัวแปลภาษา เช่น Script Engine ของภาษาพีเอชพีหรือ อาจจะมีการติดตั้ง .NET Framework ซึ่งมีตัวแปลภาษาซีแอลอาร์(Common Language Runtime) ที่ใช้แปลภาษา Intermediate จากโค้ดที่เขียนด้วย VB.NET หรือ C#.NET หรืออาจจะเป็น J2EE ที่มีตัวแปลไบต์โค้ดของคลาสที่ได้จากโปรแกรมภาษาจาวา เป็นต้น

ภาษาที่สามารถนำมาสร้างเว็บแอปพลิเคชัน ได้แก่

- HTML (Hypertext Markup Language), XHTML (Extended HTML)
- XML (Extensible Markup Language)
- JavaScript/Jscript
- ASP.NET
- PHP
- LINQ

ภาษาที่เหมาะสมนำมาเขียนเว็บแอปพลิเคชันมากที่สุด คือ เอชทีเอ็มแอล พีเอชพี จาวา เพราะภาษาเหล่านี้สามารถเรียนรู้ได้ง่าย สามารถหาแหล่งข้อมูลของการใช้งานได้ง่าย และในปัจจุบันปี 2558 ภาษาเหล่านี้เป็นที่นิยมในการนำมาสร้างเว็บแอปพลิเคชันด้วย

2.2 เอ็มวีซี (Model-View-Control : MVC)

เอ็มวีซี (MVC) [2] เป็นสถาปัตยกรรม (Architectural Pattern) ที่ใช้ในสาขาวิศวกรรมซอฟต์แวร์รูปแบบ เอ็มวีซีใช้เพื่อแยกส่วนซอฟต์แวร์ในส่วนตรรกะเนื้อหา (Domain Logic) ได้แก่ ความเข้าใจในระบบของผู้ใช้ ส่วนการป้อนข้อมูลและแสดงผล (GUI) ช่วยให้การพัฒนาการทดสอบและการดูแลรักษาซอฟต์แวร์แยกออกจากกันเอ็มวีซีประกอบไปด้วย3ส่วนคือ

2.2.1 โมเดล (Model)

เป็นส่วนของ Business Logic และส่วนของเอนิตี้ส่วนนี้แตกต่างจากแนวคิดแบบ 3-tier แยกส่วนล่างสุดเป็น DataAccessLayer (DAL) ทำหน้าที่ติดต่อกับดาต้าเบส (Database) เท่านั้น โมเดลติดต่อกับดาต้าเบสเท่านั้น และสามารถรวมเรื่องของ BusinessLogic ด้วย เช่น การคำนวณ VAT7% หรือการคำนวณค่าต่างให้อยู่ในชั้นนี้และให้การทำงานดีเยี่ยมชั้นนี้ควรทำการสร้าง ServiceLayer ขึ้นมา ตรงส่วนนี้เพื่อจัดการในส่วนของการนำข้อมูลไปใช้งานภายในโปรแกรมได้สะดวกมากขึ้น

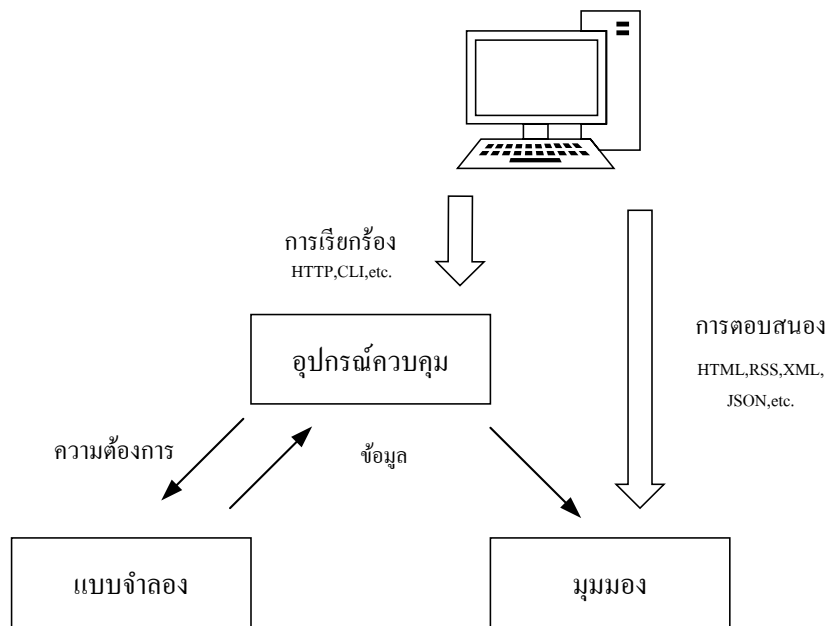
2.2.2 วิว (View)

เป็นส่วนของการแสดงผลของเอชทีเอ็มแอลซีเอสเอสและจาวาสคริปต์ (JavaScript) ในการทำเว็บแอปพลิเคชัน โดยวินโดวส์แอปพลิเคชัน (Windows Application) คือ รูปแบบฟอร์ม (Forms) ต่างๆในการใช้งาน PHP Framework บางตัวรวมเข้ากับ Template Engine เพื่อนำมาใช้งานด้วย เช่น Symfony จะมี Twig เป็น Template Engine ข้อดีของการใช้งาน Template Engine คือ การแยกส่วนของการแสดงผลกับลอจิก (Logic) ได้ชัดเจนมากขึ้นซึ่งทำให้ง่ายต่อผู้ที่พัฒนาด้วยเอชทีเอ็มแอลซีเอสเอสและจาวาสคริปต์

2.2.3 คอนโทรลเลอร์ (Controller)

ส่วนนี้เป็นส่วนประมวลผลของระบบทำหน้าที่คอยควบคุมการดึงข้อมูลจากโมเดล แล้วนำไปแสดงผลที่วิว ซึ่งโดยทั่วไปจะมีการเข้าใจผิดกันในหมู่นักพัฒนาเว็บแอปพลิเคชันเริ่มต้นในเรื่องของรูปแบบการใช้งาน MVCFramework เพราะในส่วนของคอนโทรลเลอร์จะเป็น Workflow หรือขั้นตอนการทำงานต่างๆ มีเฉพาะ Flow ของการทำงานเท่านั้นหน้าที่ของคอนโทรลเลอร์มีเพียงรับค่าและเริ่มทำการยืนยันการใช้งาน (Validate) แล้วส่งค่ากลับไปให้โมเดลเท่านั้น

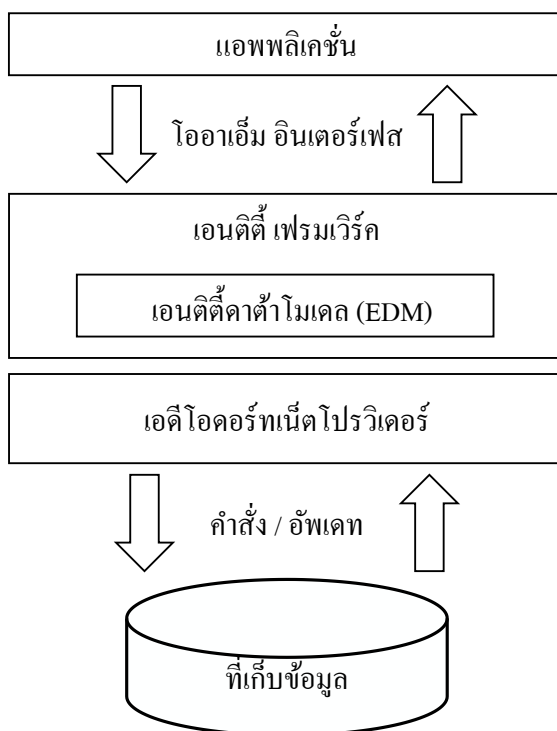
ประโยชน์ของการใช้เอ็มวีซี คือ การเพิ่มประสิทธิภาพในการบำรุงรักษา (Maintainability) หรือทำให้ระบบแก้ไขได้ง่ายขึ้นเพราะระบบมีการแยกส่วนออกจากกันอย่างชัดเจน อีกปัจจัยหนึ่งที่มีความสำคัญคือสามารถแบ่งหน้าที่การทำงานตามความถนัดของบุคลากรเช่น ผู้พัฒนาในส่วนของวิวจะมีหน้าที่ทำส่วนของการแสดงผลของซอฟต์แวร์ที่จะรับค่ามาจากโมเดลหรือคอนโทรลเลอร์ ผู้พัฒนาในส่วนโมเดลมีหน้าที่ทำซอฟต์แวร์ที่เกี่ยวข้องในส่วนฐานข้อมูล โดยเรียกข้อมูลจากดาต้าเบสเพื่อส่งให้กับวิว



รูปที่ 2.1 แนวคิดเอ็มวีซี

2.3 เอนติตี้เฟรมเวิร์ค (Entity Framework)

เอนติตี้เฟรมเวิร์ค (Entity Framework) [3] เป็นไลบรารี (Library) ที่ทำหน้าที่จัดการดาต้าเบส ร่วมกับการเขียนคลาสบนโปรแกรมที่จะเขียนขึ้นแนวคิดของเอนติตี้เฟรมเวิร์คอยู่ในรูปแบบของ Object/Relational Mapping(O/RM) คือ เอนติตี้เฟรมเวิร์คทำการสร้างเลเยอร์ที่ทำหน้าเป็นแบบจำลองข้อมูล(Database Model) ขึ้นมาเป็นคลาส (Class) ภายในโปรเจกที่กำลังถูกพัฒนา โดยทำการวางรูปแบบการเข้าถึงข้อมูลภายในตัวคลาสที่สร้างขึ้นใหม่นี้กับตารางแสดงผล (Table View) และการสร้างขึ้นตอนการบันทึก (Stored Procedure) จากดาต้าเบสเข้าสู่โปรเจกทำให้สามารถเรียกใช้ผ่านคลาสที่อยู่ในโปรเจกได้ โดยไม่จำเป็นต้องใช้คำสั่ง SQLStatement เพื่อทำการบันทึก อัปเดตหรือลบ แล้วจึงส่งไปดำเนินการที่ ตาราง โดยคิดได้ว่าต่อไปคำสั่งเอสคิวแอลบนโปรเจกสามารถจัดการกระทำที่เกิดขึ้นก่อนหน้าได้ และชนิดคำสั่ง ข้อมูลที่กำหนด (Dataset) ตารางข้อมูล (Data Table) การอ่านข้อมูล (Data Reader) การปฏิบัติที่ไม่มีการจัดลำดับ (ExecuteNonQuery) ที่ใช้สำหรับการจัดการกับดาต้าเพราะตัวเอนติตี้เฟรมเวิร์คจะทำหน้าที่แทนทั้งหมดและให้เรียกใช้งานตารางต่างๆผ่าน EntitySet ที่มีการสร้างขึ้น ส่วนรูปแบบภาษาที่ใช้ในการพัฒนาเอนติตี้เฟรมเวิร์คให้ทำงานแทนคือ ลินควินแทนทำงานร่วมกันกับGeneric ของ List ซึ่ง ลินควินมีสัญลักษณ์คำสั่งที่แตกต่างจาก SQLStatement แต่สามารถเข้าใจง่ายตามรูปแบบของภาษาเช่น วิบีคอตเน็ตหรือซีชาร์ป สามารถแสดงแนวคิดการทำงานของ เอนติตี้เฟรมเวิร์ค ได้ดังรูปที่ 2.2



รูปที่ 2.2 เอนติตี้เฟรมเวิร์ค

2.4 ลินคว (LINQ)

ลินคว (LINQ) [4] โดยทั่วไปซอฟต์แวร์คือการผสมผสานระหว่างคำสั่งและข้อมูล การพัฒนาซอฟต์แวร์คือการพัฒนาคำสั่งเพื่อดำเนินการกับข้อมูล ซึ่งการพัฒนาคำสั่งมีภาษาโปรแกรมให้เลือกจำนวนมากสามารถเลือกใช้ภาษาโปรแกรมขึ้นอยู่กับปัจจัยต่างๆหลายประการเช่น ลักษณะของงานที่จะนำไปใช้ ความถนัดของบุคคลากรภายในทีมพัฒนา ระบบปฏิบัติการที่นำมาใช้ หรือนโยบายที่บริษัทหรือลูกค้ากำหนด เมื่อดำเนินการด้วยภาษาใดภาษาหนึ่งและเสร็จสิ้นการทำงานควรมีการระบุข้อมูล เวอร์ชันการใช้งานของภาษานั้นๆ โดยข้อมูลอาจอยู่ในแหล่งต่างๆหลากหลายแบบเช่น ไฟล์ในดิสก์ ตารางในฐานข้อมูล เอกสารรูปแบบ เอกซ์เอ็มแอล (XML) ที่สืบทอดบนเว็บไซต์ หรือบ่อยครั้งที่งานเดียวต้องทำการอ้างอิงแหล่งข้อมูลทุกรูปแบบ โดยที่สุดท้ายแล้วการพัฒนาซอฟต์แวร์โครงการต้องเกี่ยวข้องกับการจัดการข้อมูลในรูปแบบใดรูปแบบหนึ่งเสมอ ซึ่งลินควถูกนำมาใช้เพื่อเป็นคำสั่งในการเรียกข้อมูลจากเอนติตี้เฟรมเวิร์คและเชื่อมโยงเข้ากับ โครงสร้างโปรแกรมภายในต่างๆ

2.5 เอชทีเอ็มแอลเรเซอร์ (HTML Razor)

การใช้ @ และตามด้วยคำสั่งภาษาโปรแกรมซีชาร์ป เพื่อกำหนดให้สามารถนำคำสั่งที่ถูกพัฒนาด้วยภาษาซีชาร์ปเข้ามาใช้งานบนหน้าเว็บที่ถูกพัฒนาด้วยภาษาเอชทีเอ็มแอลเนื่องจากไม่สามารถนำคำสั่งมาใช้ได้โดยตรง เรียกโดยทั่วไปว่า เรเซอร์ซินแทก

2.6 เอสคิวแอลเซิร์ฟเวอร์ (SQL Server)

เอสคิวแอลเซิร์ฟเวอร์ หรือ ไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์ (Microsoft SQL Server) [5] คือ ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System หรือ RDBMS) พัฒนาโดยบริษัท ไมโครซอฟต์(Microsoft) เป็นระบบฐานข้อมูลที่ถูกใช้ในรูปแบบเครื่องลูกและเซิร์ฟเวอร์ (Client/Server) และทำงานบนระบบ วินโดวเอนที (Window NT) ภาษาที่ใช้พัฒนาคำสั่งในการเรียกข้อมูลคือ ที-เอสคิวแอล (T-SQL) ด้วยเหตุที่ข้อมูลส่วนใหญ่เก็บไว้ในเครื่องที่ทำงานบนระบบปฏิบัติการ ไมโครซอฟต์วินโดวเป็นหลักทำให้ไมโครซอฟต์เอสคิวแอลสามารถนำข้อมูลในรูปแบบ วินโดวเบส (Windows base) มาเก็บและนำมาประมวลผล ประกอบกับการที่มีการลงทุนที่ต่ำและสามารถจัดหาได้ง่ายจึงเป็นปัจจัยหลักที่ทำให้ไมโครซอฟต์เอสคิวแอลเป็นระบบฐานข้อมูลที่ถูกเลือกใช้เป็นจำนวนมาก

2.7 ภาษาซีชาร์ป (C# Programming Language)

ภาษาซีชาร์ป (C# Programming Language) [6] คือ ภาษาที่ใช้ในการพัฒนาโปรแกรมแบบหลายโมเดล ที่ใช้ระบบชนิดข้อมูลแบบรัดกุม (Strong typing) และสนับสนุนการเขียนโปรแกรมเชิงคำสั่ง การเขียนโปรแกรมเชิงประกาศ การเขียนโปรแกรมเชิงฟังก์ชัน การเขียนโปรแกรมเชิงกระบวนการ การเขียนโปรแกรมเชิงวัตถุแบบคลาส และการเขียนโปรแกรมเชิงส่วนประกอบ ได้รับการพัฒนาเริ่มแรกโดยบริษัทไมโครซอฟท์เพื่อทำงานบนคอตเน็ตเฟรมเวิร์ค โดยมีแอนเดอร์ เฮลส์เบิร์ก (Anders Hejlsberg) เป็นหัวหน้าโครงการและมีรากฐานมาจากภาษาซีพลัสพลัส (C++) และภาษาอื่นๆ โดยเฉพาะภาษาเคแอลไฟและจาวา โดยมีจุดมุ่งหมายให้เป็นภาษาสมัยใหม่ไม่ซับซ้อนใช้งานได้ทั่วไป (General-Purpose) และจัดการข้อมูลให้เป็นเชิงวัตถุ ซึ่งง่ายต่อการพัฒนา

2.8 การวางแผนทรัพยากรทางธุรกิจ (Enterprise Resource Planning : ERP)

อีอาร์พี คือ การวางแผนทรัพยากรทางธุรกิจขององค์กรโดยรวม เพื่อให้เกิดการใช้ประโยชน์ได้อย่างสูงสุดของทรัพยากรทางธุรกิจขององค์กรคือ ระบบที่ใช้ในการจัดการและวางแผนการใช้ทรัพยากรต่างๆขององค์กร โดยเป็นระบบที่เชื่อมโยงระบบงานต่างๆขององค์กรเข้าด้วยกัน ยกตัวอย่างเช่น อีอาร์พีของบริษัทจะรวมตั้งแต่ระบบงานทางด้านบัญชีการเงิน ระบบงานทรัพยากรบุคคล ระบบบริหารการผลิต รวมถึงระบบการกระจายสินค้า เพื่อช่วยให้การวางแผนและบริหารทรัพยากรของบริษัทนั้นเป็นไปอย่างมีประสิทธิภาพ ทั้งยังช่วยลดเวลาและขั้นตอนการทำงานได้อีกด้วย

2.9 แอปพลิเคชันระบบจัดการสินค้า (System Application Products : SAP)

แซป คือ อีอาร์พี ตัวหนึ่งที่มีการทำงานครอบคลุมในทุกๆส่วนขององค์กรอย่างทั่วไไว้ใน อีอาร์พี และมีการเชื่อมต่อซึ่งกันและกันได้อย่างดีของทุกสายงานในองค์กรหรือบริษัท ได้ข้อมูลที่ต้องการแม่นยำสมบูรณ์อีกทั้งระบบยังมีความสามารถมากมายเพื่อช่วยในการวิเคราะห์การดำเนินงานของธุรกิจ โปรแกรมแซปเป็นโปรแกรมที่ค่อนข้างเปิดต้องมีการปรับเปลี่ยนปรับแต่ง (Custom) ก่อนนำไปใช้งานเพื่อให้เข้ากับธุรกิจของบริษัททำให้มีอาชีพทางด้านแซปเกิดขึ้นมากมาย

2.10 ริสปอนซีฟดีไซน์เนอร์ (Responsive Design)

ริสปอนซีฟดีไซน์เนอร์ (Responsive Design) [7] คือ แนวคิดการออกแบบที่เรียกว่า One Size Fit All คือ ออกแบบเพียงครั้งเดียว แต่สามารถใช้ได้กับทุกขนาดของหน้าจอ โดยเว็บไซต์จะสามารถตรวจจับขนาดของหน้าจอ และปรับขนาด และเลย์เอาต์ (Layout) ให้เหมาะสมตามขนาดของหน้าจอโดยอัตโนมัติ โดยอาศัยการทำงานร่วมกันระหว่างเทคโนโลยีต่างๆ ดังต่อไปนี้ คือ เลย์เอาต์ของเว็บแบบ Flexible Grid รูปภาพแบบ Flexible Image และ CSS3 Media Query

ประโยชน์ของการออกแบบเว็บไซต์แบบريسปอนซีฟ

- แสดงผลได้สวยงามบนขนาดหน้าจอที่แตกต่าง
- ออกแบบเพียงครั้งเดียวแต่สามารถใช้ได้กับขนาดหน้าจอที่หลากหลาย
- มี Experience ในการทำงานที่ดีกว่า แสดงข้อมูลได้ง่ายโดยไม่ต้องขยายภาพ
- ประหยัดเวลา และ ค่าใช้จ่ายได้มากกว่าการทำ 2 เว็บไซต์
- ช่วยในเรื่องของการทำ SEO (Search Engine Optimization)