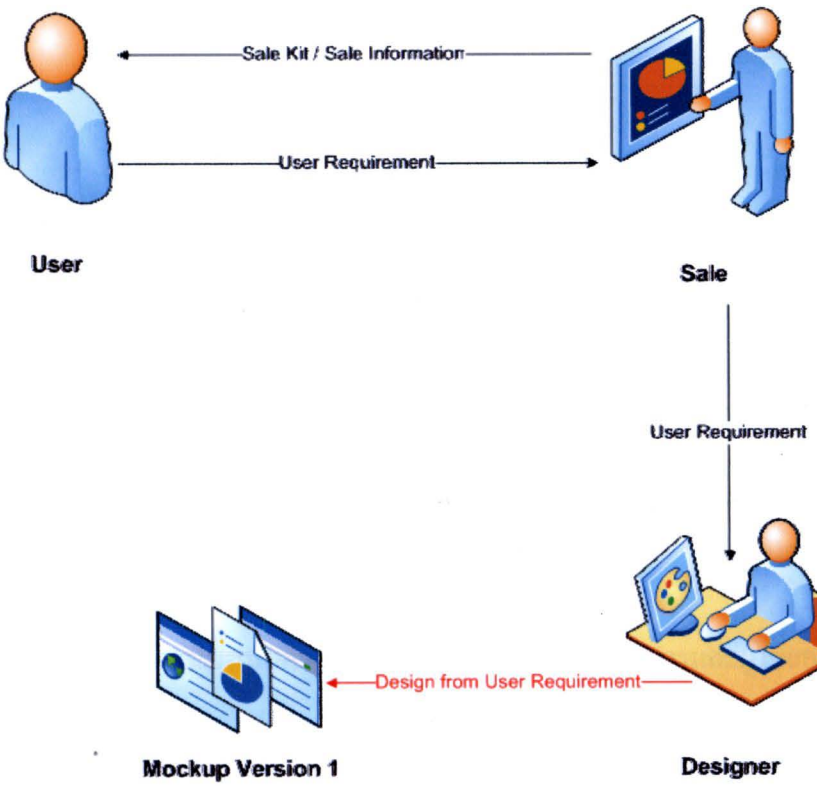
บทที่ 3

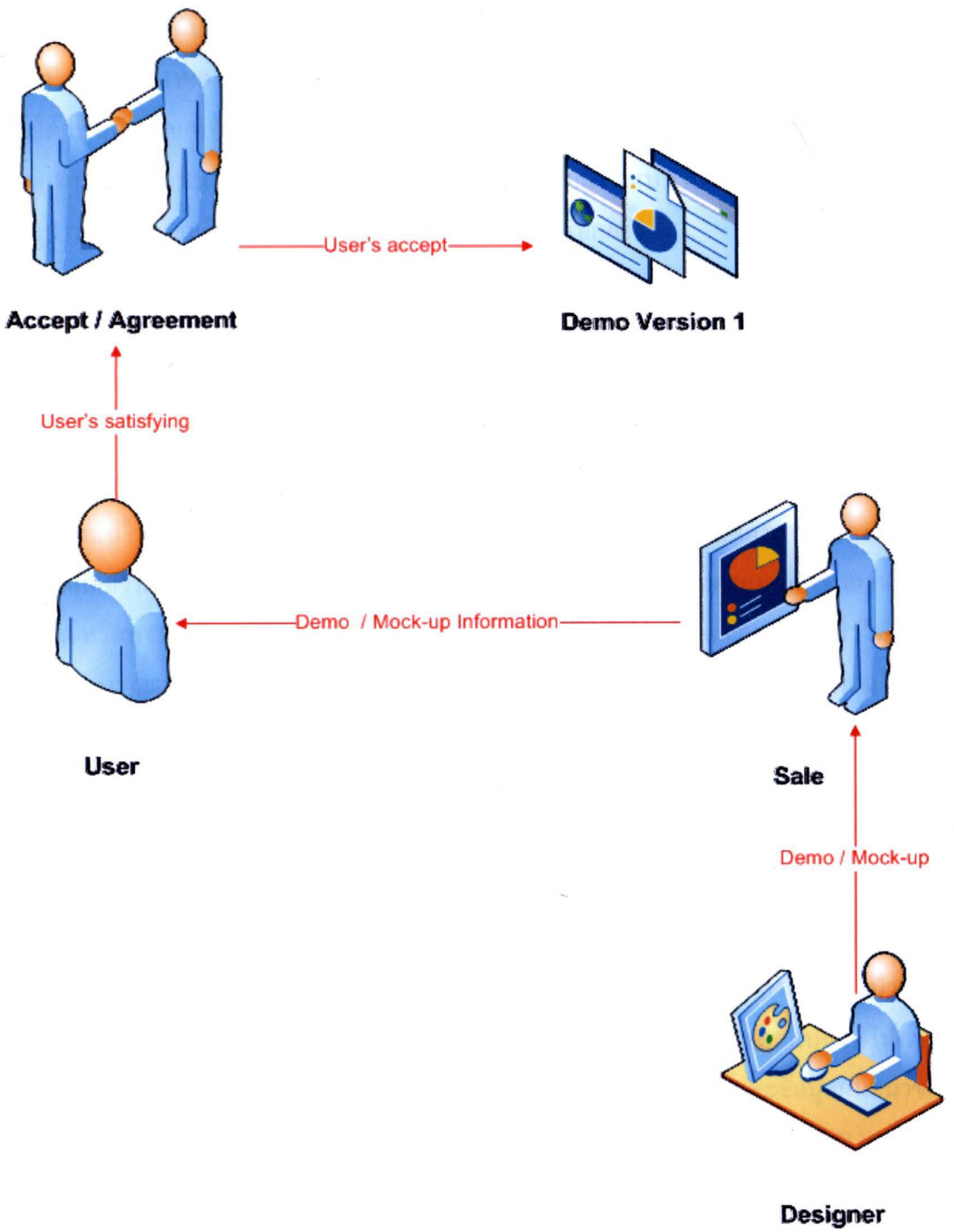
โครงสร้างและขั้นตอนการทำงานของระบบ

ในบทนี้จะเป็นการกล่าวถึงข้อมูลของกระบวนการและขั้นตอนการทำงานของระบบจีคิว ทั้งหมดอย่างละเอียด โดยเริ่มจากการเปรียบเทียบกับระบบเดิมที่นำไปใช้ก่อนที่จะนำระบบจีคิวเข้ามาปรับปรุง ระบบและลักษณะโครงสร้างในแต่ละขั้นตอนตลอดจนรายละเอียดโดยสรุปของโครงสร้างดังกล่าว และรวมไปถึงผังงานของข้อมูลต่างๆที่ใช้และเกิดขึ้นภายในระบบจีคิว โดยจากที่ได้กล่าวมาในข้างต้นสามารถแบ่งออกเป็นหัวข้อต่างๆได้ดังนี้

3.1 ระบบและกระบวนการในรูปแบบเดิม



ภาพ 3-1 กระบวนการเก็บความต้องการและออกแบบส่วนประสานงานจากผู้ใช้งานในรูปแบบเดิม



ภาพ 3-2 กระบวนการนำเสนอตัวอย่างตัวต้นแบบจากความต้องการและผู้ใช้ยอมรับได้

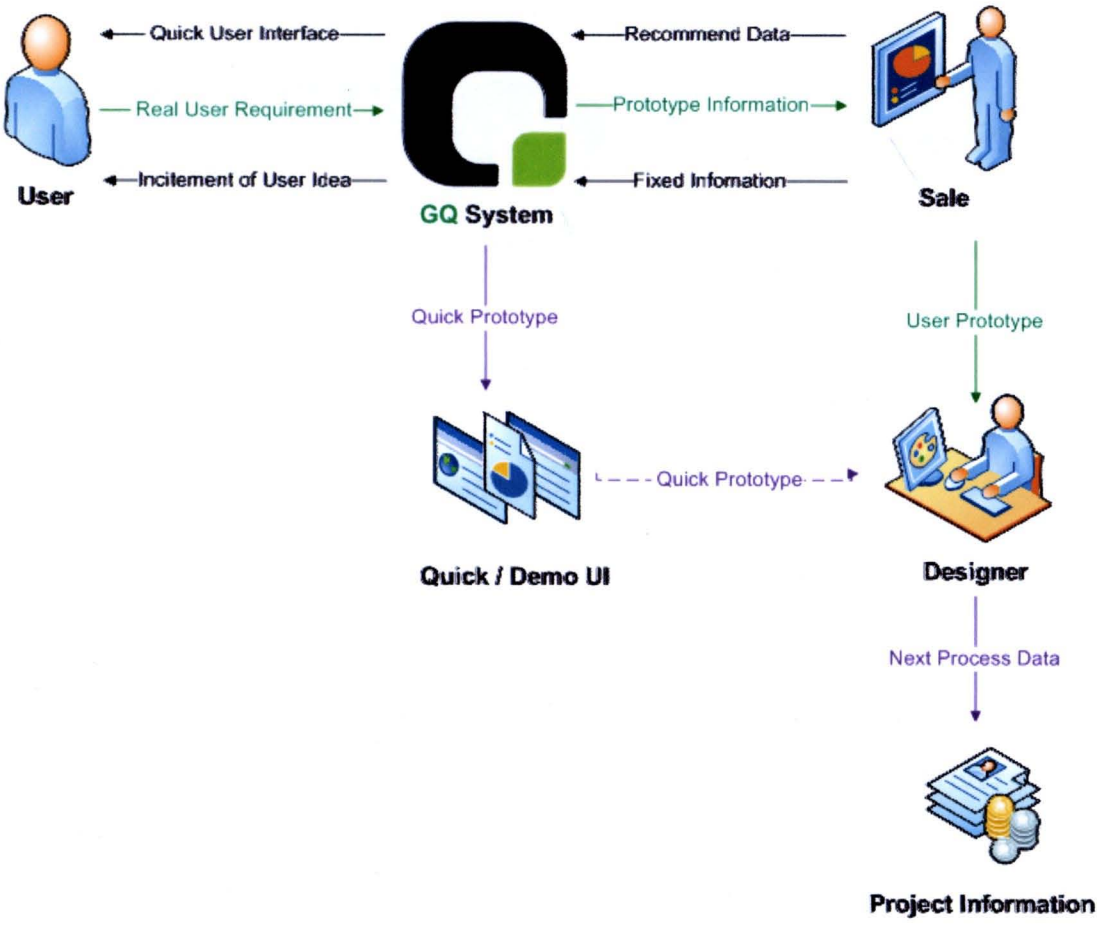
จากภาพ 1-3 เริ่มกระบวนการจากผู้ขายได้เสนอขายสินค้าหรือผลิตภัณฑ์ IRN กับผู้ใช้ และเริ่มมีการสอบถามว่าผู้ใช้ต้องการให้รูปแบบของสินค้าหรือผลิตภัณฑ์ออกมาอยู่ในรูปแบบไหน

ความต้องการเบื้องต้นเป็นเช่นไร โดยผู้ขายจะทำหน้าที่เสนอความต้องการที่สามารถทำได้หรือแนวคิด เช่น ตัวอย่างของผลงานเก่าๆที่เคยพัฒนาผ่านมา หรือ แนวความคิดตามความต้องการที่ผู้ใช้อยากได้ เป็นต้น โดยหลังจากนั้นผู้ขายจะนำความต้องการที่ได้จากผู้ใช้กลับไปพิจารณาและส่งให้ผู้ออกแบบได้ออกแบบรูปแบบลักษณะส่วนประสานงานกับผู้ใช้ แบบโครงร่างโดยสังเขปหรือตัวต้นแบบ เพื่อให้ผู้ขายหรือเซลล์ ได้นำกลับไปนำเสนอกับผู้ใช้อีกครั้ง

ซึ่งในส่วนของการเก็บข้อมูลความต้องการของผู้ใช้งานในขั้นตอนนี้ถือว่าเป็นขั้นตอนที่มีความสำคัญมากเป็นอย่างยิ่ง เนื่องจากถ้าเซลล์เก็บความต้องการของผู้ใช้มาไม่หมดหรือว่าไม่ครบถ้วน เมื่อผู้ออกแบบ ได้ออกแบบตัวต้นแบบกลับไปให้ผู้ใช้งาน อาจจะทำให้เกิดข้อผิดพลาดที่ว่าตัวต้นแบบไม่ตรงกับความต้องการของผู้ใช้งานที่ได้คุยกับเซลล์เอาไว้ และอาจจะส่งผลให้เซลล์ต้องกลับไปเก็บความต้องการเพิ่มและปรับตัวต้นแบบให้ตรงตามความต้องการนั้นๆมากที่สุด ส่งผลให้เกิดความล่าช้าในงานหรือโครงการนั้นๆ หรือ เกิดลูป (Loop) ของกระบวนการออกแบบที่ไม่สิ้นสุด หรือบางกรณีอาจจะส่งผลให้โครงการล้มเหลวไม่ประสบผลอย่างที่คาดหมายไว้ได้ แต่ถ้าผู้ใช้งานสามารถยอมรับได้กับสิ่งที่ผู้ออกแบบและเซลล์ได้นำเสนอไป จะทำให้เกิดขั้นตอนในภาพ 3-2 ซึ่งสามารถนำตัวต้นแบบดังกล่าวไปออกแบบและใช้ในขั้นต่อไปของกระบวนการออกแบบและพัฒนาซอฟแวร์ได้

3.2 ระบบและกระบวนการในรูปแบบจีคิว (GQ)

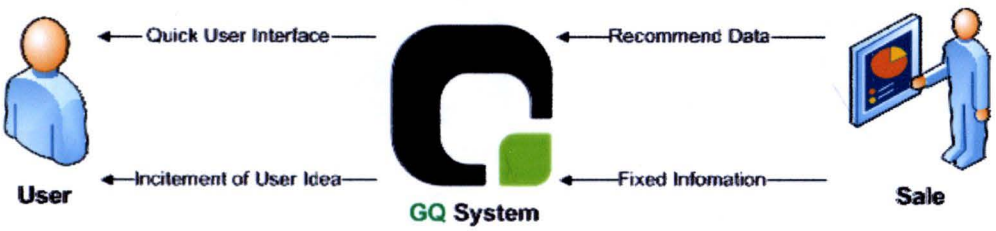
ภายในระบบของจีคิวจะแตกต่างจากระบบเดิมที่เซลล์ต้องเก็บความต้องการและนำเสนอข้อมูลเพียงฝ่ายเดียว แต่ภายในระบบจีคิวจะทำให้ผู้ใช้งานมีส่วนร่วมในการออกแบบให้ให้ความต้องการได้อย่างเต็มที่ โดยตัวของระบบจีคิว เมื่อได้รับความต้องการของผู้ใช้และข้อมูลแนะนำของเซลล์แล้วจะสามารถสร้างตัวต้นแบบของส่วนประงานกับผู้ใช้ออกมาได้ทันที และถ้าหากผู้ใช้ไม่พอใจกับตัวต้นแบบ (ที่เป็นส่วนประสานงานกับผู้ใช้) ผู้ใช้สามารถเพิ่มเติมและแก้ไขได้ทันที จนได้ตัวต้นแบบที่คาดว่าจะตรงต่อความต้องการกับตนเองและตนเองพอใจมากที่สุดดังภาพ 3-3



ภาพ 3-3 กระบวนการทั้งหมดของระบบจีคิว

3.2.1 การรับความต้องการเข้าสู่ระบบจีคิว

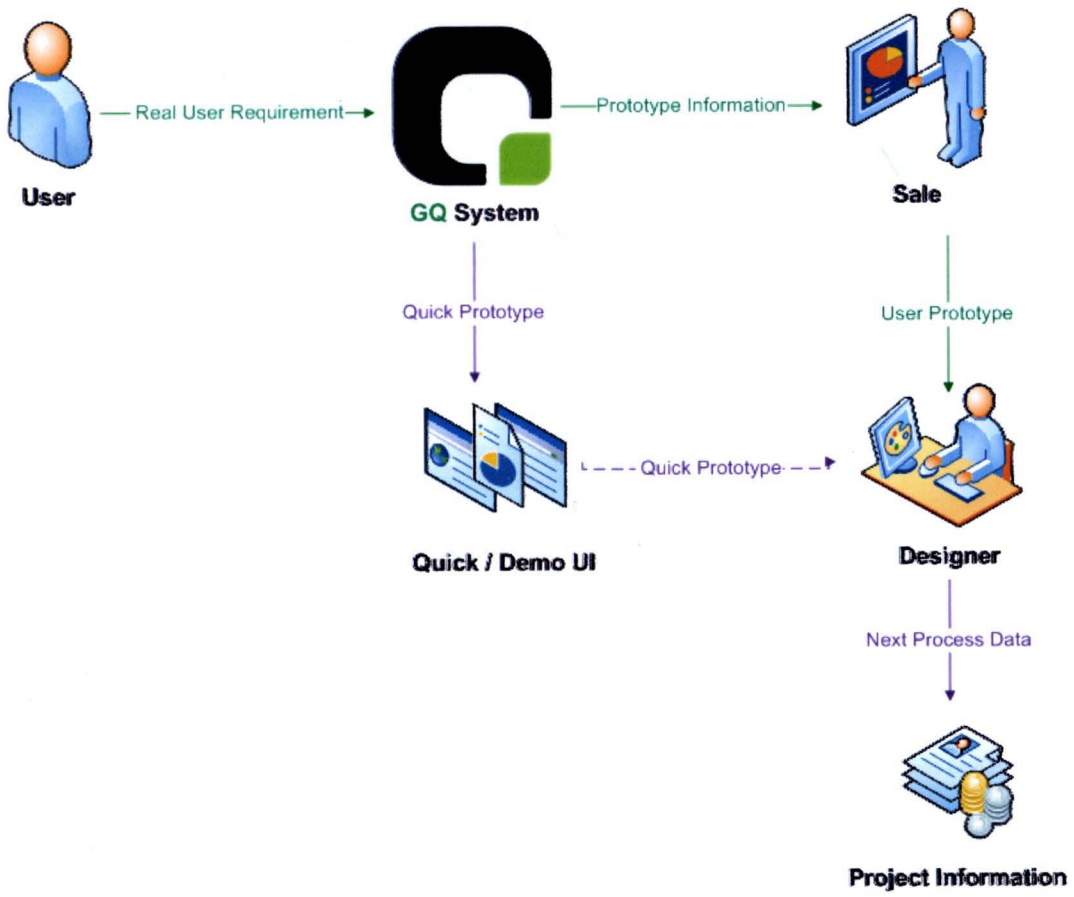
ก่อนที่ระบบจะทำการประมวลผลออกมาเป็นตัวต้นแบบอย่างเร็วนั้น ระบบจะต้องรับความต้องการเบื้องต้นจากผู้ใช้งานก่อน ดังภาพ 3-4 โดยในที่นี้จะถูกแบบออกเป็นสองส่วนหลักๆ คือ ส่วนแรกคือข้อมูลพื้นฐานและความรู้เกี่ยวกับเว็บไซต์เบื้องต้น ตลอดจนตัวอย่างของตัวต้นแบบที่ถูกต้องตามมาตรฐาน หรือง่ายต่อการออกแบบในขั้นตอนการพัฒนา ที่เซลล์จะต้องแนะนำให้กับผู้ใช้งาน (Fixed informationandrecommend data) ซึ่งจะรวมเข้ากับความต้องการในส่วนที่สองคือความต้องการของผู้ใช้งาน (Main user interface) ที่ต้องการจะสร้างให้ออกมาอยู่ในลักษะอย่างไร โดยผู้ใช้งานจะต้องร่วมทำการออกแบบส่วนประสานงานผู้ใช้กับเซลล์ ซึ่งในส่วนนี้ระบบจะเป็นตัวช่วยดึงเอาความต้องการของผู้ใช้งานที่ซ่อนอยู่ออกมาให้ได้มากที่สุด และสุดท้ายก็จะได้ตัวต้นแบบออกมาให้ผู้ใช้ได้ตัดสินใจเลือกและนำไปสู่ขั้นตอนต่อไป



ภาพ 3-4 กระบวนการแนะนำผู้ใช้งานผ่านระบบจีคิวและข้อมูลจากเซลล์

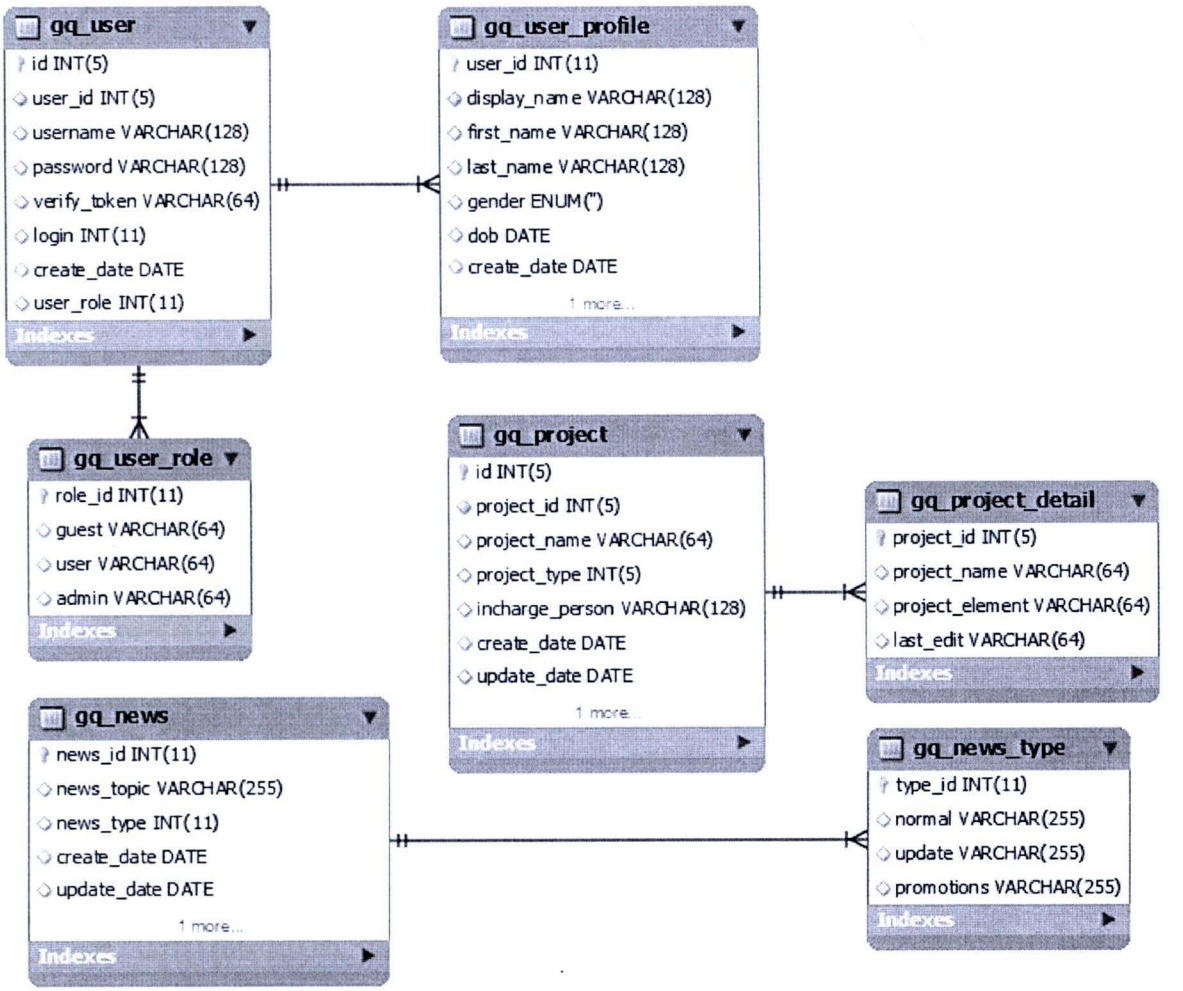
3.2.2 การออกแบบตัวต้นแบบจากความต้องการของผู้ใช้งาน

จากหัวข้อที่ 3.2.1 เมื่อได้ความต้องการที่แท้จริงจากผู้ใช้งานแล้ว ระบบจะทำการรวบรวมและประมวลผลออกมาเป็นตัวต้นแบบของส่วนประสานงานกับผู้ใช้ (ชนิดเร่งด่วน) ตลอดจนข้อมูลความต้องการที่เป็นข้อมูลเสริมที่เป็นประโยชน์กับเซลล์ โดยตัวต้นแบบของส่วนประสานงานกับผู้ใช้ที่ได้นั้นจะถูกส่งไปให้กับเซลล์หรือผู้ออกแบบได้ออกแบบ ตัวต้นแบบ (ชนิดจริง) หรือตัวอย่างส่วนประสานงานกับผู้ใช้ ที่จะนำไปใช้จริงกับโครงการนั้นๆ ตามภาพ 3-5 ซึ่งอาจจะสามารถนำไปใช้งานได้ทันที หรือ ว่าอาจจะนำไปให้ผู้ใช้ได้พิจารณาอีกรอบ ซึ่งต่างจากขั้นตอนเดิมที่ผู้ใช้ไม่ได้มีส่วนร่วมในการออกแบบส่วนประสานงานกับผู้ใช้ หรือตัวต้นแบบอย่างสิ้นเชิง ตลอดจนช่วยลดเวลาในการติดต่อประสานงานระหว่างเซลล์กับผู้ใช้ ที่จะต้องมีการติดต่อกันทุกครั้งเมื่อเกิดการเปลี่ยนเปลี่ยนความต้องการ (Change) ให้ตรงกับความต้องการจริงให้มากที่สุด และช่วยลดความเสี่ยงในการการทำงานที่ช้าลงให้เสร็จตามกำหนดที่ได้วางแผนไว้ได้ ตลอดจนสามารถนำข้อมูลที่ได้ไปใช้ในขั้นตอนต่อไปของระบบการพัฒนาและออกแบบซอฟแวร์ของโครงการได้



ภาพ 3-5 ข้อมูลที่ได้จากระบบจีคิว

3.3 โครงสร้างฐานข้อมูล (ER Diagram)

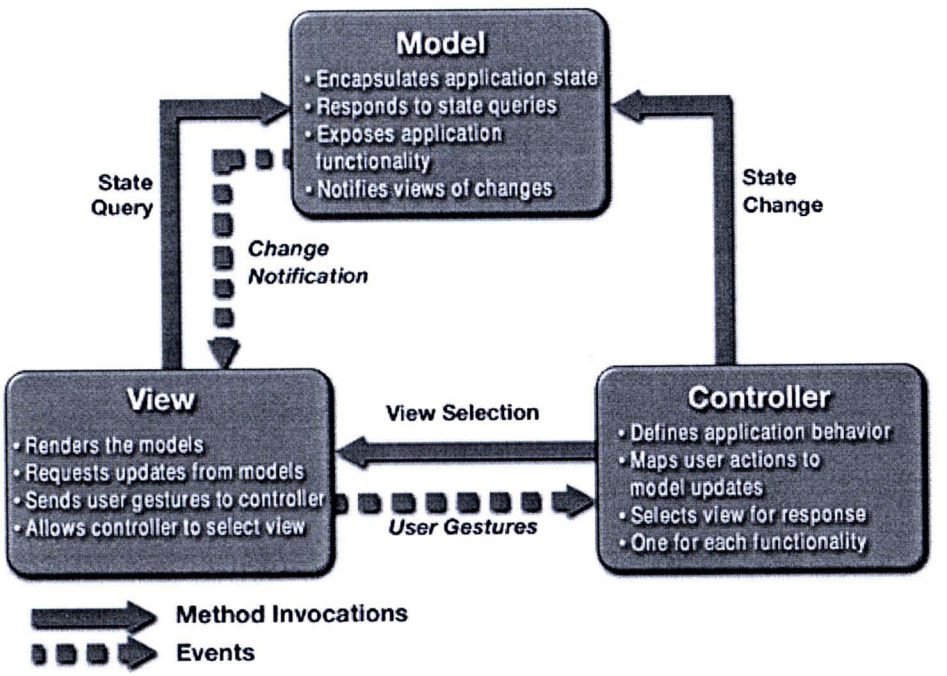


ภาพ 3-6 GQ Entity Relationship Diagram (ER-Diagram)

3.4 Kohana : PHP Framework Model

โคฮานา เฟรมเวริค์ (Kohana Framework) เป็นอีกหนึ่งชื่อของกลุ่มโค้ดของภาษาพีเฮชพี (PHP language) ที่ถูกออกแบบและพัฒนาขึ้นมาสนับสนุนการทำงานตามหลักการของการแยกส่วนการทำงานของระบบให้เป็นไปตามโครงสร้างเอ็มวีซี (MVC) เอ็มวีซี หรือ Model-View-Controller (MVC) คือ สถาปัตยกรรมซอฟต์แวร์ชนิดหนึ่ง ซึ่งในขณะนี้ถือว่าเป็นแบบแผนสถาปัตยกรรม (Architectural pattern) ที่ใช้ในสาขาวิศวกรรมซอร์ฟแวร์ รูปแบบ เอ็มวีซี ใช้เพื่อแยกส่วนซอฟต์แวร์ในส่วน การตัดสินใจของระบบ (Domain logic)ได้แก่ความเข้าใจในระบบของผู้ใช้ และส่วนการ

ป้อนข้อมูลและแสดงผล (GUI) ซึ่งช่วยให้การพัฒนา การทดสอบ และการดูแลรักษาซอฟต์แวร์ ให้แยกออกจากกันเพื่อให้ง่ายต่อการนำไปใช้และพัฒนา



ภาพ 3-7 โครงสร้างเอ็มวีซี

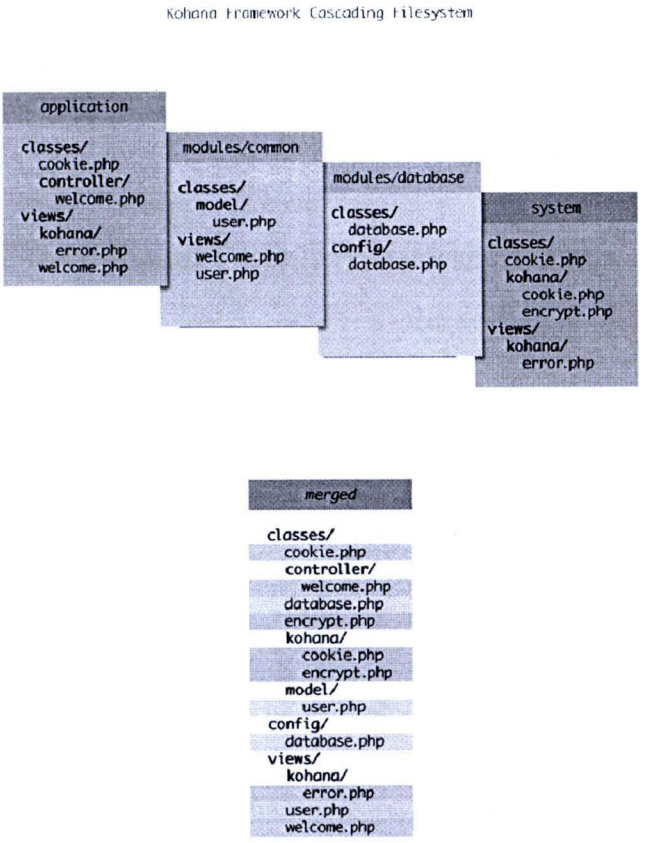
- โมเดล (Model) หมายถึง ส่วนของซอฟต์แวร์ที่ใช้ในการแปลการทำงานของระบบ ไปสู่สิ่งที่ระบบซอฟต์แวร์ได้ถูกออกแบบเอาไว้ การตัดสินใจของระบบ ใช้เพื่อให้ความหมายแก่ข้อมูลดิบ ยกตัวอย่างเช่น การคำนวณว่าวันนี้เป็นวันเกิดของผู้ใช้หรือไม่, หรือจำนวนเงินรวม ภาษี และค่าส่งสินค้า ในตะกร้าสินค้า ฯลฯ เมื่อโมเดลมีการเปลี่ยนแปลง จะมีการส่งคำเตือนให้แก่ วิว ที่เกี่ยวข้องเพื่อปรับค่า ระบบซอฟต์แวร์หลายระบบใช้การเก็บข้อมูลถาวร เช่น ฐานข้อมูล เพื่อเก็บข้อมูลเหล่านี้ เอ็มวีซีไม่ได้กำหนดถึงระดับการเข้าถึงข้อมูล เพราะเป็นที่เข้าใจกันว่าส่วนนี้จะอยู่ภายใต้ หรือถูกครอบคลุมด้วยโมเดล โมเดลไม่ได้เป็นเพียงอ็อบเจกต์ (Object) ที่ใช้เข้าถึงข้อมูล แต่ในระบบซอฟต์แวร์เล็กๆ ซึ่งมีความซับซ้อนน้อยจะไม่เห็นความแตกต่างมากนัก
- วิว (View) แสดงผลค่าในโมเดลในรูปแบบที่เหมาะสมต่อการปฏิสัมพันธ์กับผู้ใช้ ในแต่ละโมเดลสามารถมีวิวได้หลายแบบ เพื่อใช้ในจุดประสงค์ที่ต่างกัน

- คอนโทรลเลอร์ (Controller) รับค่าเข้ามา และทำการตอบสนองโดยเรียกใช้ออบเจคในโมเดลแอปพลิเคชันที่ใช้ เอ็มวีซีอาจจะเป็นกลุ่มของ โมเดล/วิว/คอนโทรลเลอร์ โดยแต่ละกลุ่มใช้ในงานต่างกันไป

เอ็มวีซีมักจะพบได้ในเว็บแอปพลิเคชันโดย วิว จะเป็น HTML หรือ XHTML ที่สร้างโดยแอปพลิเคชันนั้น ส่วนคอนโทรลเลอร์รับค่า GET หรือ POST เข้ามา แล้วเลือกติดต่อกับโมเดลในส่วนที่เกี่ยวข้องเพื่อตอบสนอง โมเดลซึ่งมีกฎหรือเงื่อนไขต่างๆจะทำการจัดการตามคำร้องขอนั้นๆ

โคฮาน่า เฟรมเวริค์ (Kohana PHP framework) คือ เฟรมเวริค์ที่ใช้สำหรับออกแบบและพัฒนาเว็บแอพลิเคชั่นด้วยภาษาพีเฮชพี ที่มีการพัฒนาและออกแบบตามหลักการและโครงสร้างตามรูปแบบของเอ็มวีซี ซึ่งการทำงานของโคฮาน่านั้นสามารถแบ่งการทำงานออกเป็น 3 ส่วน ตามภาพ 3-8ดังนี้

- ตัวควบคุม (Controller) จะทำหน้าที่ควบคุมและจัดการระบบการทำงานของ โค้ดพีเฮชพี ให้เป็นไปตามรูปแบบและมาตรฐานที่กำหนด โดยผู้พัฒนาสามารถเพิ่มการทำงาน เงื่อนไข หรือออกแบบกระบวนการต่างๆของเว็บแอพลิเคชั่นได้ในส่วนนี้ และถ้าพิจารณาตามภาพ3-8ส่วนการจัดการและการควบคุมจะอยู่ในส่วนของแอพลิเคชั้นและระบบ (application and system)
- โมเดล (Model) เปรียบเสมือนเครื่องมือหรือตัวช่วยที่ถูกออกแบบมาเพื่อเรียกใช้งานในรูปแบบและความต้องการต่างๆ ซึ่งการทำงานในแต่ละโมเดลก็จะแตกต่างกันออกไป แต่การเรียกใช้จะอยู่บนมาตรฐานการทำงานของตัวควบคุม
- มุมมอง (Views) เป็นส่วนที่ผู้ใช้สามารถเรียกใช้งานได้ โดยการทำงานต่างๆจะถูกเรียกใช้จากตัวควบคุม หรืออาจจะรวมไปถึงโมเดล ที่มีการออกแบบโค้ดมาแล้วจากตัวควบคุมทำให้ระบบเกิดความปลอดภัยสูง ผู้ใช้งานไม่สามารถทราบได้ว่า ไฟล์ที่กำลังเรียกใช้อยู่นั้นถูกจัดการจากส่วนไหนของระบบ



ภาพ 3-8 โครงสร้างโคฮาน่า

โคฮาน่า จึงเป็นเฟรมเวริค์ทีมีมาตรฐานของการทำงาน การออกแบบโค้ด ต่างๆให้อยู่ในรูปแบบเดียวกัน ง่ายต่อการเรียกใช้และพัฒนากันเป็นทีม ตลอดจนมีความปลดภัยเรื่องระบบความปลอดภัยที่สูง มีฟังก์ชันการทำงานสำเร็จรูปหลากหลายพร้อมที่จะถูกเรียกใช้งานได้ทันทีโดยไม่ต้องกำหนดหรือออกแบบโค้ดใหม่ มีประสิทธิภาพที่รวดเร็วและไม่กินทรัพยากรต่างๆในการทำงานของระบบ ซึ่งในระบบจีคิว ได้มีการนำเอาโคฮาน่ามาเป็นระบบกลางในการเรียกใช้งานส่วนต่างๆของเว็บแอพลิเคชั่น